

CENTRO UNIVERSITÁRIO DA FEI

GUILHERME ALBERTO WACHS LOPES

**RECONHECIMENTO DE OBJETOS UTILIZANDO
PERCEPÇÃO MULTISENSORIAL COMPETITIVA
BASEADA EM REDES COMPLEXAS**

São Bernardo do Campo
2014

RESUMO

A área de Visão Computacional, sobretudo para aplicações envolvendo reconhecimento de objetos, utiliza geralmente modelos matemáticos inspirados em estatística ou geometria, muitas vezes com foco apenas nos resultados, sem se preocupar com aspectos biológicos, como cognição. Embora alguns desses modelos sejam espelhados em mecanismos advindos da Psicologia e Neurociência já conhecidos, como visão top-down/bottom-up, outros aspectos, como competição e colaboração entre sinais inspirados em percepção, foram parcialmente incorporados aos estudos, ou por não serem considerados totalmente ou por serem considerados de maneira isolada. Um modelo único que agregue esses aspectos ainda não foi proposto pela literatura. Este trabalho apresenta um Meta-Modelo para reconhecimento de objetos em cena baseado em alguns aspectos da Neurociência. A hipótese central dessa tese é que as informações contextuais de uma cena podem aumentar a qualidade do reconhecimento de objetos. No Meta-Modelo proposto, uma rede complexa é implementada como um elemento central para agregar informações de co-ocorrências entre objetos que aparecem em uma mesma cena. A implementação proposta permite o estudo da organização topológica entre diversos sinais que competem ou colaboram entre si. Resultados preliminares sugerem que algumas dessas informações parecem seguir uma organização baseada no contexto.

Palavras-chave: Redes Complexas, Reconhecimento de Objetos, Grafos, Aprendizagem Infinita

Keywords: Complex Networks, Object Recognition, Graphs, Never-Ending Learning

SUMÁRIO

1 INTRODUÇÃO	1
1.1 Objetivo	3
1.2 Principais Contribuições da Tese	3
2 CONCEITOS FUNDAMENTAIS	5
2.1 Visão Computacional e Modelos de Reconhecimento de Objetos	5
2.1.1 Modelos de Representação de Imagens e Características	10
2.2 Aprendizagem de Máquina	18
2.2.1 Aprendizagem Supervisionada	20
2.2.2 Aprendizagem Não Supervisionada	27
2.2.3 Aprendizagem por Reforço	35
2.3 Influência da Neurociência nos Modelos de Visão Computacional	40
2.4 Redes Complexas	55
2.4.1 Modelos de Redes	57
2.4.2 Exemplos e Aplicações de Redes Complexas	60
2.4.3 Características Físicas de Redes Complexas	62
3 PROPOSTA	71
3.1 Base de Dados	71
3.2 Metodologia	73
3.2.1 Descrição Geral do Modelo	75

3.2.2 Sub-Modelo de Treinamento	77
3.2.3 Sub-Modelo Central	79
3.2.4 Modelo Intermediário de Inferência	83
4 PROPOSTA EXPERIMENTAL	88
4.1 Sub-Modelo de Treinamento	88
4.1.1 Definição das Características a Serem Extraídas das Regiões	88
4.1.2 Grau de Discretização das Instâncias de Características	89
4.2 Sub-Modelo Central	89
4.2.1 Estudo Sobre a Equação de Co-Ocorrência	90
4.3 Sub-Modelo de Inferência	91
4.3.1 Experimentos na Etapa de Segmentação	91
4.3.2 Estudo Sobre a Função do Maximizador	91
5 EXPERIMENTOS E RESULTADOS PRELIMINARES	93
6 Cronograma	97
REFERÊNCIAS	98

1 INTRODUÇÃO

A área de Visão Computacional tem sido uma das mais estratégicas dentro da área de Ciência da Compucação, com aplicações em diversas outras áreas, tais como: engenharia, física, biologia, medicina, astronomia, entre muitas outras.

Por outro lado, o avanço da multimídia aliado à popularização da internet tem contribuído para ampliar o potencial dessa área, haja vista a quantidade de vídeos e imagens que todos os dias são despejados às “toneladas” por usuários comuns em todo o mundo, tornado nossas vidas mais fáceis, contudo dependentes do gerenciamento adequado desse volume de informação gerado. Um exemplo disso são os dados médicos que, ao mesmo tempo que facilitam a interpretação de diagnósticos e gerenciamento de exames, também dificultam o tráfego e armazenamento dessas informações.

Mas não é somente na área médica onde a quantidade de informação multimídia vem crescendo. Mesmo para o entretenimento ou dados de informações governamentais de uma nação, os avanços tecnológicos relacionados a vídeos e imagens podem ter benefícios imediatos muito vantajosos.

Devido a esses tipos de aplicações, o volume de informação multimídia demanda por sistemas de gerenciamento cada vez mais elaborados. Tradicionalmente, a área de visão computacional lida com esses problemas de duas maneiras. A primeira delas trata de problemas específicos sem a preocupação de que o método seja inspirado em modelos biológicos. Geralmente, chamados de sistemas especialistas, esses modelos tendem a focar na resolução do problema sem se preocupar com o mecanismo de solução. Um exemplo clássico desse tipo de problema é a análise do controle de qualidade em linhas de produção, detecção de placas de veículos, reconhecimento de digitais, íris ou faces.

A segunda maneira, mais utilizada em interpretação de objetos em cenas, procura solucionar problemas através de modelos inspirados em mecanismos biológicos a medida que esses

também vão sendo compreendidos e propostos.

Um exemplo clássico bem conhecido foi a proposição das Redes Neurais Artificiais (R.N.A.) para reconhecimento de padrões. Quando surgiram, desde a década de 50, criou-se a expectativa de que tratava-se de um modelo representativo da própria estrutura cerebral humana. Sabemos hoje, no entanto, que as R.N.A. são um ferramental limitado considerando este ponto de vista.

Ao longo das últimas décadas, muitos modelos inspirados na biologia humana surgiram dentro da área da neurociência e psicologia para explicar os mecanismos de visão humana. Muitos desses mecanismos foram trazidos, totalmente ou em parte, para a área de Visão Computacional com o objetivo de interpretar cenas ou reconhecer objetos, mas também interessados em validar os modelos propostos. Nesse sentido, a área de Visão Computacional obteve grande sucesso ao analisar características individuais, como cor, forma ou textura. No entanto, muito trabalho ainda deve ser feito na integração dessas características individuais. Por exemplo, sabe-se que cor, contraste e relacionamento espacial são informações fundamentais para interpretação de objetos em cena, mas não há um consenso geral sobre como realmente essas características cooperam entre si em um único mecanismo de interpretação de objetos ou cenas.

Existem muitos aspectos cognitivos relacionados ao processo de interpretação de objetos ou cenas que ainda são ensejo de profundo debate na área da neurociência e psicologia. Um exemplo são informações de contexto, que consideram objetos distribuídos na cena para inferência de outros objetos. Apesar de, a informação contextual ser um fator determinante na interpretação da cena, não há na área de Visão Computacional muitos trabalhos que explorem esses aspectos, nem como característica, nem como modelo matemático. Outros mecanismos ainda mal compreendidos são os mecanismos de visão Top-Down, Bottom-Up, movimento e foco.

Embora, hoje em dia, na área da Neurociência já haja uma melhor compreensão desses mecanismos, poucos deles foram incorporados ou, pelo menos, considerados parcialmente na área de Visão Computacional para criação de modelos matemáticos visando a interpretação de cena. O que se encontra na literatura científica da área de Visão Computacional geralmente é o estudo e a introdução individual desses mecanismos. Por exemplo, sistemas de *tracking* costumam se basear

apenas no movimento dos objetos para rastrear o alvo. Propostas mais recentes, como o SIFT ou Bag-Of-Words, apostam na mistura de características, mas sem considerar nenhum mecanismo neural.

Para atender a demanda tanto do volume de informações multimídias quanto de suas aplicações, a área de Visão Computacional deve seguir um modelo que incorpore ao máximo tanto as características da área da Neurociência quanto os mecanismos que unificam essas características nas tarefas de análise e reconhecimento de objetos em cena. Entretanto, não há na literatura um modelo único, nem com relação ao mecanismo, nem com relação às características. A proposta dessa tese é avançar a área nesse sentido, propondo um mecanismo que permita discutir descobertas recentes da Neurociência aplicados à Visão Computacional.

1.1 Objetivo

Apresentar e validar um Meta-Modelo para reconhecimento de objetos em cenas naturais, que incorpore aspectos cognitivos atualmente estudados na Neurociência, mas ainda não plenamente incorporados como um único modelo matemático em Visão Computacional.

1.2 Principais Contribuições da Tese

Esta tese apresenta um Meta-Modelo para reconhecimento de objetos utilizando o modelo de Redes Complexas para gerenciamento de informações contextuais para reconhecimento de objetos em cenas. Dentre as contribuições desta tese, destacam-se:

- a) Incorporação de alguns aspectos da Neurociência que influenciam no reconhecimento de objetos em cena, e que foram pouco abordados na área de Visão Computacional.
- b) Modelagem contextual de objetos utilizando a co-ocorrência entre diferentes características extraídas a partir de imagens.
- c) Um modelo único que incorpora os modelos de visão top-down/bottom-up, e competição/collaboração entre características extraídas a partir de imagens.

- d) Estudo de novas características extraídas de imagens e suas influências sobre o reconhecimento de objetos.
- e) Um modelo adaptativo que utiliza os conceitos de Aprendizagem Infinita.
- f) Utilização dos conceitos de Redes Complexas para estudar e analisar a organização topológica da estrutura aprendida pelo modelo.

2 CONCEITOS FUNDAMENTAIS

2.1 Visão Computacional e Modelos de Reconhecimento de Objetos

Visão Computacional é um ramo da inteligência artificial que surgiu em meados dos anos 60 com o problema de reconhecimento de textos a partir de folhas impressas. Sua crescente utilização em ambientes industriais novos desafios e chamou a atenção de muitos cientistas. Com o passar do tempo, novas aplicações foram desenvolvidas utilizando esta técnica. Exemplos dessas aplicações são: reconhecimento de chips eletrônicos, sistemas de controle de qualidade, sistemas para análise de imagens médicas, sistemas de segurança automotivos, sistemas de rastreamento de objetos, entre muitos outros.

Um sistema de visão computacional deve extrair de uma imagem, ou sequência de imagens, informações capazes de descrever seu conteúdo, propriedades, forma, iluminação, distribuição de cores entre outras características (Szeliski, 2010) relevantes para o problema a ser tratado. Essa definição mostra o quanto genérico, complexo e abrangente é esse campo de pesquisa.

Há décadas, cientistas, tanto da área da psicologia quanto da área da computação, tentam entender como o sistema visual humano trabalha com o objetivo de criar um modelo fidedigno. Porém, sua complexidade vai além da *percepção* do mundo como o vemos, envolvendo também a *cognição* humana. Por isso, muitos pesquisadores acabam subestimando os problemas de visão computacional e acabam adotando heurísticas ou até mesmo diminuem o escopo do projeto a fim de ter um resultado aceitável sem tanta complexidade no modelo computacional.

Dos diversos tipos de problemas tratados pela visão computacional, podemos generalizá-los em duas categorias: problemas em ambientes controlados e problemas genéricos (Andreopoulos; Tsotsos, 2013). O primeiro tipo é mais abordado em casos onde é definida uma tarefa restrita, tais como: controle de qualidade de produtos industriais, identificação de chips eletrônicos, leitura de digitais. Os sistemas de visão para tais tarefas são superiores a humanos, uma vez que

são mais rápidos e construídos especificamente para atingir um objetivo. Por outro lado, os sistemas de visão computacional genéricos envolvem meta-modelos e vão ao encontro das questões de análise contextual e semântica. Neste caso, ainda não se conhece um modelo matemático que consiga armazenar e processar informações tal como os humanos.

Os sistemas de visão computacional genéricos são mais difíceis de serem modeladas porque lidam com o *problema inverso* (Szeliski, 2010); isto é, devemos procurar algo desconhecido considerando informações insuficientes para especificar totalmente a solução.

Apesar dos avanços da área nos últimos 50 anos, ainda não há um modelo definitivo capaz de interpretar e extrair informações básicas de uma imagem, tal como faz uma criança (Szeliski, 2010; Palmer, 1999; Marr, 1982). Uma tarefa simples como contar o número de pessoas em um ambiente não controlado ainda é uma realidade complexa para os pesquisadores.

São diversas as aplicações reais em que metodologias de visão computacional são utilizadas, como exemplo temos: reconhecimento óptico de caracteres, inspeção industrial, fotometria para geração de objetos 3D, análise de imagens médicas, segurança automotiva, vigilância, reconhecimento digital e biometria, entre outros. Contudo, muitas dessas aplicações são especialistas e específicas para reconhecer ou analisar características bem definidas.

Dentre os problemas encontrados na área de visão computacional, mais especificamente em ambientes não controlados, destaca-se o de reconhecimento de objetos. Segundo (Prasad, 2012; Dickinson, 1999), reconhecimento de objetos lida com a identificação da presença de diversos objetos em uma imagem e é uma das principais tarefas do sistema visual humano. Porém, para ambientes não controlados o problema ainda é um desafio, com muitos pontos em aberto.

O reconhecimento automático de objetos é uma tarefa que envolve um modelo meta-heurístico em que muitas vezes deve ser executado em tempo real. Em um modelo meta-heurístico, erros são toleráveis até uma especificação, mas o mais importante é que ele seja genérico o bastante para ajudar a resolver problemas muitas vezes intratáveis¹. Geralmente, esses problemas estão relacionados à busca de uma solução em um espaço n-dimensional. No domínio de imagens

¹Entenda como “intratável” um problema que não pode ser solucionado em tempo polinomial.

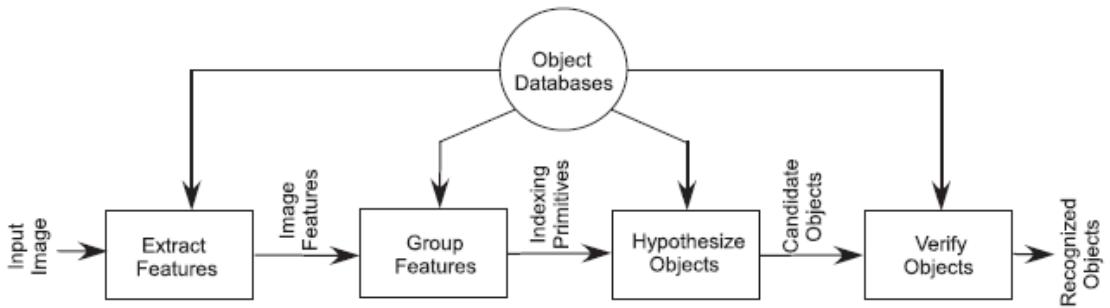


Figura 2.1: Componentes básicos de um típico sistema de reconhecimento de objetos (ANDREOPoulos; TSOTSOS, 2013)

e vídeos, esses espaços de buscas são grandes o bastante para tornar algoritmos de força bruta inviáveis. A escolha por uma meta-heurística será necessária para guiar os algoritmos na busca da solução ideal.

O trabalho de (PRASAD, 2012) também revela que a acuracidade dos modelos atuais não são satisfatórios, alcançando somente cerca de 16% de acerto sobre 22.000 categorias de objetos no caso do *Google Glass*.

O trabalho de (ANDREOPoulos; TSOTSOS, 2013) faz um estudo sobre a área de reconhecimento de objetos nos últimos 50 anos e as técnicas e modelos tradicionais. Além disso, o trabalho identifica as principais tarefas envolvidas no processo.

Como mostrado na Figura 2.1, um sistema de reconhecimento de objetos tem como entrada um imagem digital. Uma vez adquirida a imagem, o próximo passo é extrair uma ou mais características ou *features*. Esse processo transforma a imagem de entrada em estruturas que capturam informações de cor, forma e textura, tais como: arestas (descontinuidades de brilho e/ou cor), cantos, regiões (áreas com cores homogêneas). Além disso, o objetivo é extrair características que serão suficientes para possibilitar a localização e identificação dos objetos de interesse.

As características extraídas da imagem são então processadas e agrupadas como primitivas de indexação na etapa “Group Features”. Essas primitivas têm a mesma função de uma linguagem de consulta em um banco de dados. Tome como exemplo um sistema de reconhecimento de aeroportos a partir de imagens de satélite. Neste caso, as primitivas de indexação são as linhas que delimitam as pistas.

De posse das primitivas, o sistema de reconhecimento deverá buscar em uma base de conhecimento por algum objeto que contém a mesma primitiva. Esse processo é nomeado como *matching algorithm* ou algoritmo de casamento. O resultado desse processo é um conjunto de objetos candidatos.

Contudo, se o conjunto de objetos candidatos for maior que um, o sistema deverá escolher o objeto com maior semelhança. A esse processo damos o nome de *evaluation* ou avaliação. Nesse processo, deve-se escolher um candidato por métodos de ranqueamento baseado em medidas heurísticas. Essas medidas heurísticas dão uma nota para cada objeto candidato. O candidato que tiver a maior nota será o objeto escolhido.

Do ponto de vista da Neurociência, o processo de reconhecimento de uma cena, ou objetos em uma cena, pelo sistema visual humano, pode ser visto de duas maneiras: *top-down* ou *bottom-up*. Ainda não há um consenso geral sobre qual a estratégia deve ser utilizada, ou mesmo se ambas deve ser utilizadas. No processo *top-down*, considera-se o reconhecimento de uma cena de forma abstrata antes de serem identificados os componentes principais. Por outro lado, o processo *bottom-up* considera-se o inverso. Inicialmente, pequenas características isoladas são identificadas, juntando-as para formar regiões e, finalmente, criar uma cena abstrata.

Independente do tipo de abordagem que realmente é seguida pelo sistema visual de primatas, é de consenso quase universal dentro da área de visão computacional que o processo de reconhecimento de objetos ou cenas pode ser dividido, pelo menos para efeitos didáticos, em níveis *top-down* ou *bottom-up*. É comum na literatura sobre o assunto, considerar pelo menos três níveis:

- Baixo nível: onde se trabalha com os *pixels*, elementos individuais de textura ou pequenas regiões. Algoritmos típicos ou métodos para lidar com esse nível são: filtros espaciais ou filtros de frequência, conversores de formatos, restauradores de artefatos como ruídos, efeitos indesejados, compressão de informações e esterografia. De forma geral, algoritmos de baixo nível normalmente recebem como entrada uma imagem e retornam como saída uma outra imagem.

- Médio nível: onde trabalha-se com regiões ou agrupamento de regiões. Algoritmos típicos para lidar com problemas nesse nível pode ser: segmentadores de imagens, métodos morfológicos, clusterizadores de regiões, restauradores de imagens baseados em regiões, algoritmos de inter-relacionamento de regiões para descrição da estrutura da cena, entre outros. O processamento de médio nível normalmente recebe uma imagem como entrada e devolve atributos tais como: bordas, contornos e regiões com objetos de interesse.
- Alto nível: neste nível, trabalha-se com a cena como um todo, considerando as regiões e objetos como sendo seus elementos cognitivos. Algoritmos típicos para lidar com problemas de médio nível são: classificadores, localizadores de objetos em cenas, rastreamento de objetos ou classificadores de cena.

Uma referência didática, que descreve a área em três níveis, pode ser encontrada em (GONZALEZ; WOODS, 2002).

Dentro dessa ideia de divisão do processo de visão computacional, vários autores têm proposto diferentes trabalhos e modelos de reconhecimento, tanto em sentido *top-down*, quanto em sentido *bottom-up*.

Um exemplo de arquitetura *bottom-up* é encontrado em (LOWE, 1987, 1999; HUTTENLOCHER; ULLMAN, 1990). Em (LOWE, 1987), o autor cria um modelo probabilístico para encontrar características locais, como linhas, que pertencem a um mesmo objeto. Após essa localização há um processo de união desses elementos em uma estrutura de dados. Essa transformação relaxa o objeto que se quer encontrar na imagem e torna o processo de busca dos objetos mais eficiente. O autor ainda comenta que a identificação de grupos estáveis (sem ruído, ou elementos acidentais) na imagem é uma tarefa típica do sistema visual humano. Contudo, o modelo proposto pelo autor é restrito a objetos primitivos e mais simples de detectar. Assim, para identificar uma face humana, por exemplo, seria necessário acrescentar novos módulos ao reconhecedor.

Em (KHAN; WEIJER; VANRELL, 2009), os autores exploram a modelagem *top-down* a partir de informações que relacionam cor com objetos; isto é, a coloração das regiões de uma imagem ajuda o sistema a encontrar regiões onde há objetos. Após esse processo, novas características

são extraídas das regiões delimitadas para serem classificadas. Contudo, o modelo sugerido não contém parâmetros de ajuste, fazendo com que, para algumas classes de objetos, o reconhecimento não tenha sido satisfatório. A falta de parâmetros pode impedir o uso de classificadores supervisionados (veja Seção 2.2).

Quando se fala em reconhecimento de objetos, estamos nos referindo a modelos matemáticos capazes de *classificar* um conjunto de dados obtidos a partir de uma imagem digital. Contudo, pesquisadores sempre buscam modelos mais representativos e capazes de reconhecer diversas classes de objetos. A compreensão desses modelos é profundamente estudada na área de aprendizagem de máquina. Portanto, o estudo de aprendizagem de máquina em conjunto com os classificadores é necessário quando se deseja criar um modelo de reconhecimento de objetos.

2.1.1 Modelos de Representação de Imagens e Características

Do ponto de vista do sistema visual humano, existem muitas características que podem ser utilizadas para a interpretação de um objeto ou cena. No entanto, somente um sub-conjunto delas é utilizado para aplicações computacionais. Entre elas, as mais populares envolvem a representação de luminância, cor, forma, textura e movimento.

Para imagens digitais binárias, duas formas de representação de luminância são essenciais. A primeira delas é através de uma função $f : (x, y) \rightarrow \{0, 1\}$, onde x, y são as coordenadas para um ponto da imagem. A imagem dessa função é formada pelos valores de luminância 0 para preto e 1 para branco. Para imagens em tons de cinza, essa função é dada por $f : (x, y) \rightarrow \mathbb{R}$, sendo capaz de representar valores entre preto e branco. Esta é a maneira tradicional de representação de imagens sempre que é necessária sua exibição ou leitura de valores individuais de luminância.

Por outro lado, uma segunda maneira de representar imagens digitais em tons de cinza é através da representação como histograma. A função correspondente a essa representação é dada por $f : k \rightarrow \mathbb{R}^+$, onde k é um valor de luminância e a imagem da função é a quantidade de *pixels* que têm k como valor de luminância. Uma conclusão imediata desse modelo de representação é que as informações de localização espaciais dos valores de luminância são perdidas. Contudo,

essa técnica é utilizada em aplicações cujas informações espaciais não de interesse, tais como segmentação, ajuste de contraste, ajuste de brilho, entre outras.

Para representação de imagens coloridas, é necessário primeiramente definir um sistema de cores. Um sistema de cores é um espaço multidimensional que descreve cores. Existem diversos sistemas de cores, contudo dois deles se destacam na área de Visão Computacional: RGB e HSV. O sistema RGB (Red, Green, Blue) é composto por três componentes $(r, g, b) \in \mathbb{R}^3$, no qual cada componente descreve a intensidade de cada canal de cor primária. Por outro lado, o sistema de cores HSV é uma transformação não-linear do sistema RGB e é normalmente utilizado em aplicativos de edição de imagens por serem mais intuitivos e perceptivos ao sistema visual humano. O sistema HSV é formado por três componentes (h, s, v) , matiz, saturação e valor, respectivamente.

Uma função de representação de imagens digitais coloridas f é feita utilizando um sistema de cores de tal forma que $f : (x, y) \rightarrow \mathbb{R}^d$, onde d é o número de componentes do sistema de cores. De maneira semelhante às imagens em tons de cinza, também é possível representar uma imagem digital colorida através do histograma de cores. Em (Wachs-Lopes; FUKUMA; RODRIGUES, 2012), os autores criam um sistema de detecção de tomadas em vídeos de futebol. O objetivo do trabalho é classificar a cena como principal ou secundária. Foi utilizado histogramas gerados a partir do sistema HSV e a teoria da informação como uma maneira de comparar as semelhanças entre esses histogramas. Para isso, os autores discretizaram o sistema HSV possibilitando 18 valores para matiz, 3 valores para saturação e 3 valores para valor, gerando um histograma de 162 posições. Para cada imagem, os autores as descreveram através dos histogramas. Em seguida, foi gerado um histograma médio sobre o conjunto de histogramas gerados a partir de tomadas principais supervisionadas. De posse do histograma médio das cenas de tomada principal, os autores utilizaram a divergência de Kullback-Leibler com uma cena de tomada desconhecida. Um limiar foi definido para determinar se a tomada desconhecida era principal ou secundária. Os resultados mostraram que o histograma HSV foi capaz de classificar com 97% de acerto. Esse trabalho mostra que a informação de frequência foi suficiente para classificar as tomadas.

A característica de textura em imagens é outro domínio estudado na área de visão computacional para representação de imagens. Ao contrário da característica de cor, a textura de uma

imagem é dada a partir de uma região e não somente sobre um ponto. A textura é uma característica que geralmente descreve a disposição espacial de cores e padrões de intensidades em uma imagem inteira ou em uma região específica. Três modelos de representação de imagem a partir da característica de textura serão apresentados nesta tese: baseado em estatística, baseado em fatores psicológicos e utilizando processamento de sinais.

A representação de imagens utilizando características estatísticas de textura pode ser feita através de uma matriz co-ocorrência P gerada a partir do relacionamento entre as vizinhanças dos *pixels* (HOWARTH; RÜGER, 2004a). Cada elemento $p_{i,j}$ dessa matriz contém a frequência relativa entre dois *pixels* vizinhos, um com luminância i e outro com luminância j , de acordo com uma regra de vizinhança estabelecida previamente. Contudo, a determinação da vizinhança de um pixel envolve duas constantes: a distância d e o ângulo θ . Assim, uma matriz de co-ocorrência gerada com $d = 1$ e $\theta = 0^\circ$ será capaz de representar as relações de luminância entre *pixels* esquerdos/direitos. O trabalho de (HARALICK, 1979), propõem a utilização de algumas funções aplicadas sobre a matriz P com o objetivo de extrair mais informações de textura. Algumas dessas equações são mostradas abaixo:

Energia	$\sum_{i,j} p_{i,j}^2$
Entropia	$\sum_{i,j} p_{i,j} \log p_{i,j}$
Probabilidade Máxima	$\max_{i,j} p_{i,j}$
Homogeneidade	$\sum_i \sum_j \frac{p_{i,j}}{1+ i-j }$

Uma segunda abordagem para representação de imagens utilizando características de texturas foi proposta em (TAMURA; MORI; YAMAWAKI, 1978). Na área de Visão Computacional, essa abordagem é conhecida como *Tamura*. Para essa representação, os autores se basearam no sistema visual humano para definir algumas características extraídas de imagens, na qual três delas obtiveram resultados semelhantes quando comparadas a humanos. A primeira característica é denominada *coarseness* (rispidez), que é responsável pela quantificação do tamanho do menor elemento da textura. Essa característica é obtida primeiramente extraiendo as cores média de cada

pixel sobre uma vizinhança de tamanho variável, segundo a Equação (2.1)

$$A_k(x,y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} \frac{f(i,j)}{2^{2k}} \quad (2.1)$$

onde k é o tamanho da vizinhança e f é a função da imagem. Os valores de A_k para cada pixel da imagem são então comparados com outros *pixels* que estão na fronteira vertical e horizontal. A Equação (2.2) mostra a comparação das médias na projeção horizontal.

$$E_{k,hor}(x,y) = |A_k(x+2^{k-1},y) - A_k(x-2^{k-1},y)| \quad (2.2)$$

A escolha de k que maximiza $E_{k,hor}$ e $E_{h,ver}$ para um determinado pixel determinará o valor de sua rispidez. Para obter a rispidez de uma região da imagem, extrai-se a média da rispidez sobre todos os *pixels* dessa região.

A segunda característica extraída por *Tamura* comparada com o sistema visual humano é o contraste. Essa característica é calculada em termos do desvio padrão e a curtoise do histograma de cores da imagem segundo a Equação (2.3),

$$contraste = \frac{\sigma}{(\alpha_4)^n} \quad (2.3)$$

onde σ^2 é a variância, n é uma constante empiricamente definida como $1/4$ e α_4 é a curtoise, calculada através da Equação (2.4).

$$\alpha_4 = \frac{\frac{1}{XY} \sum_x \sum_y (f(x,y) - \mu)^4}{\sigma^4} \quad (2.4)$$

onde σ^2 é a variância das luminâncias dos *pixels*.

A terceira característica extraída por *Tamura* é a direção. Essa característica utiliza o operador *Prewitt* para detectar a direção das arestas da imagem. Um histograma contendo as frequências de cada ângulo é gerado e utilizado como descritor.

Outro método empregado na representação de imagens utilizando características de textura é feito através dos filtros de *Gabor*, que permite a filtragem de imagens no domínio espacial e da frequência através da transformada *Wavelet* (HOWARTH; RÜGER, 2004b). A transformada *Wavelet* é

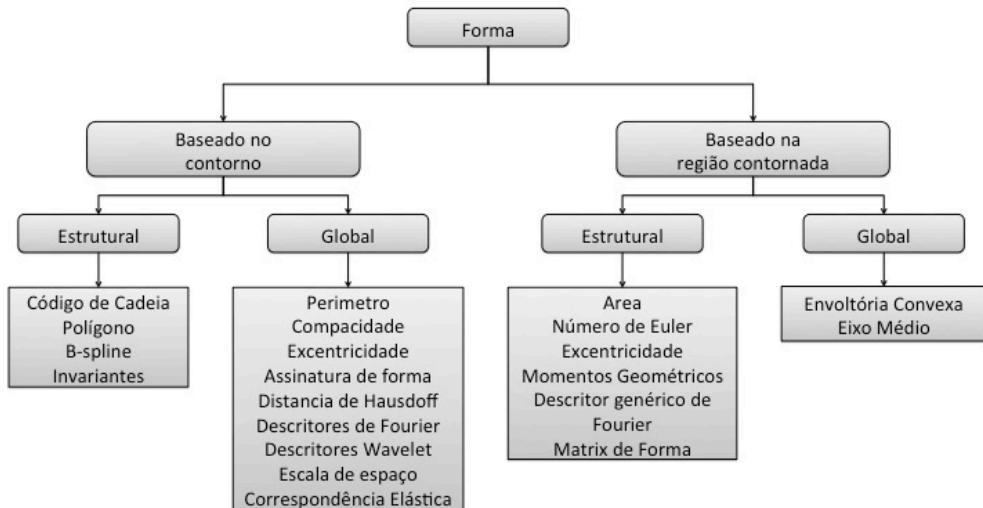


Figura 2.2: Classificação e algumas características de forma (ZHANG; LU, 2004)

semelhante a transformada de *Fourier*, porém com um *kernel* que está localizado tanto no espaço de *Fourier* quanto no espaço Real. O filtro de *Gabor* permite compor um vetor de características que armazena informações de orientação e frequência através da utilização de diversas configurações de escala e angulação. Os valores obtidos nesse vetor podem ser utilizados como um modelo de representação de imagens.

Outra característica utilizada para a representação de imagens é a forma. Para a utilização dessa característica é necessário primeiramente definir uma área de interesse sobre a imagem original utilizando, por exemplo, um segmentador. As características de forma podem ser classificadas em dois grupos: baseada no contorno e baseada na região delimitada. A diferença entre essas características está em determinar se as informações serão extraídas a partir da linha de contorno ou sobre toda a área delimitada pelo contorno (ZHANG; LU, 2004). As características de forma também podem ser classificadas em globais ou estruturais. Características estruturais descrevem a forma a partir do próprio domínio espacial e as características globais descrevem a forma através de uma transformação. A Figura 2.2 classifica algumas características de forma que são descritas em (ZHANG; LU, 2004). Dentre as características apresentadas nessa figura, três se destacam na área de visão computacional: descritores de Fourier, assinatura de forma e código de cadeia.

Os descritores de Fourier utilizam a representação de pontos de um contorno discretizado através de um plano complexo. Mais formalmente, seja $C = \{(x_0, y_0), (x_1, y_1), \dots, (x_{N-1}, y_{N-1})\}$

o conjunto de pontos de um contorno discretizado. Essas coordenadas podem ser representadas através de duas funções: $x(k) = x_k$ e $y(k) = y_k$. Para fazer a transformação do domínio cartesiano para o domínio imaginário utiliza-se a Equação (2.5).

$$s(k) = x(k) + jy(k) \quad (2.5)$$

Nessa equação, o eixo x é representado pela parte real do espaço e o eixo y pela parte imaginária. Através da transformada discreta de *Fourier* é possível encontrar os $N - 1$ descritores de C segundo a Equação (2.6).

$$a(u) = \frac{1}{N} \sum_{i=0}^{N-1} s(i) e^{-j2\pi ui/N} \quad (2.6)$$

Uma vez que essa equação é apenas uma transformação espacial, é possível fazer a transformada inversa de *Fourier* e obter as coordenadas originais de C . Embora a Equação (2.6) revele $N - 1$ descritores, é possível desprezar alguns desses descritores de tal forma que somente as tendências principais da forma sejam capturadas. Para isso, é necessário zerar os componentes de $a(u)$ que representam as altas frequências; isto é, para todo $u > P$, onde P é o número de componentes que se deseja considerar. A Figura 2.3 ilustra a transformada inversa dos descritores de *Fourier* sobre um quadrado para diferentes valores de P . Uma propriedade interessante dos descritores de *Fourier* é que as baixas frequências da série estão associadas às características globais da forma, enquanto as altas frequências estão associadas aos detalhes da forma. Isso explica o comportamento da Figura 2.3, uma vez que quanto maior o valor de P , maior é a quantidade de detalhes da forma.

Outro método para descrição de imagens utilizando informações de forma é através do método de assinatura de forma. Esse método baseia-se na criação de um vetor gerado a partir do contorno da forma. O método tradicional de assinatura de forma utiliza as distâncias dos pontos do contorno igualmente espaçados até seu centróide, conforme demonstrado na Equação (2.7).

$$s(u) = \sqrt{(x(u) - c_x)^2 + (y(u) - c_y)^2} \quad (2.7)$$

Nesta equação, $x(u)$ e $y(u)$ são as coordenadas do ponto u do contorno, e c_x e c_y são as coorde-

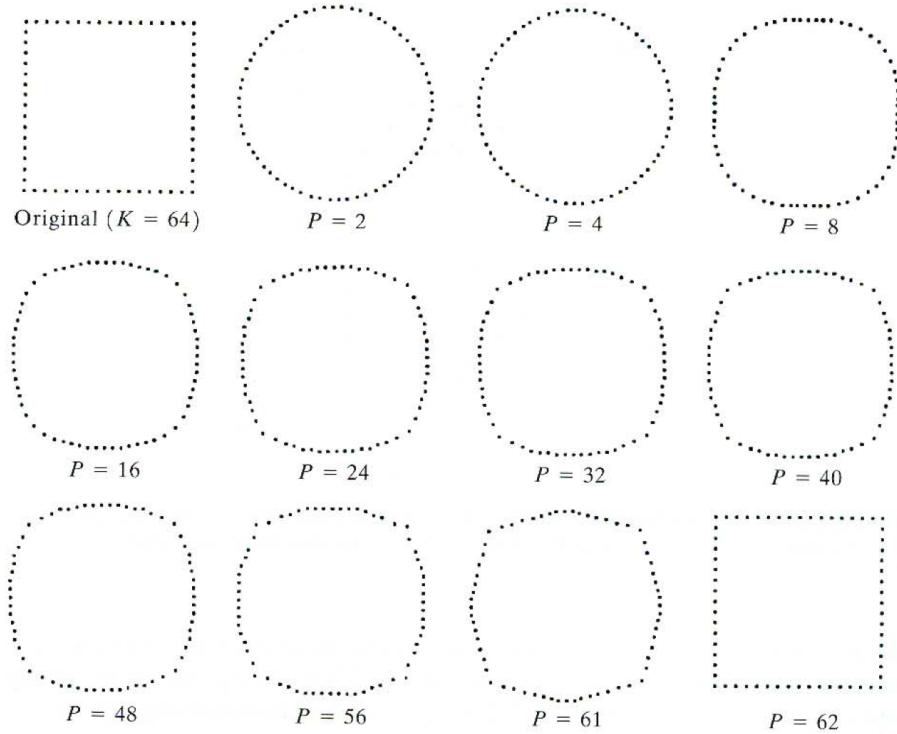


Figura 2.3: Descritores de Fourier aplicados a um quadrado com diferentes valores de P (GONZALEZ; WOODS, 2002)

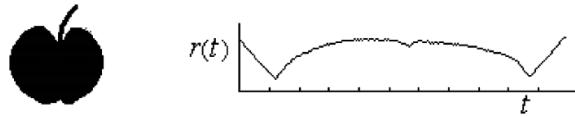


Figura 2.4: Uma forma descrita através das distâncias entre o centróide e os pontos discretizados de seu contorno (ZHANG; LU, 2004)

nadas do centróide da forma. A Figura 2.4 mostra um exemplo de assinatura de forma. Na área de recuperação de informação, esta representação é frequentemente normalizada, tornando-a invariante à escala. Contudo, a invariância de rotação é feita através de *shift matching*; isto é, ao comparar duas assinaturas, mantém-se uma assinatura fixa e desloca-se outra assinatura sobre seu eixo das ordenadas até que se encontre a menor distância entre elas. Essa técnica deve ser analisada com cautela, uma vez que a complexidade para encontrar a forma mais semelhante a partir de outra forma dada em um banco de tamanho F toma tempo computacional $O(F.n^2)$, visto que são feitos $n - 1$ deslocamentos sobre um vetor e a medida de distância é $\theta(n)$ para cada forma do banco.

As representações vistas até aqui são chamadas de Modelos de Representação de Imagens,

uma vez que o objetivo é descrever uma imagem utilizando algumas características extraídas a partir das imagens. O histograma, template ou domínio de frequência e domínio espacial são os domínios que são popularmente utilizados.

Recentemente, foram propostos modelos mistos; isto é, modelos que agregam mais do que uma característica para a representação de imagens. O *SIFT* é um dos mais conhecidos dentre eles (CHANG et al., 2010). A palavra *SIFT* é um acrônimo para *Scale-Invariant Feature Transform*. Trata-se de um método para representação de imagens através da identificação de pontos discriminantes que são menos afetados por fatores de rotação e escala. Este método é utilizado nas áreas de Reconhecimento de Objetos, Reconhecimento de Gestos, Monitoramento de Vídeo, Identificação de Indivíduos entre outras áreas correlatas a Visão Computacional.

O método *SIFT* utiliza a distribuição de gradiente em pontos específicos e em diferentes escalas para determinar pontos chaves. De forma mais específica, o *SIFT* é composto por quatro etapas. A primeira etapa tem por objetivo localizar um conjunto P de pontos que não mudam de características quando a imagem sofre uma operação de escala. Para isso, o método de diferença de gaussianas é utilizado, onde cada gaussiana é aplicada a imagens de diferentes escalas. A segunda etapa filtra os P pontos utilizando a matriz Lapaciana da imagem. Com essa transformação, é possível localizar os pontos que apresentam maiores contrastes que fazem parte do novo conjunto P filtrado. A terceira etapa, insere a informação de orientação para cada ponto $p \in P$. Essa informação poderá ser então utilizada mais tarde quando uma imagem semelhante, porém com uma orientação diferente, for empregada em uma comparação. A quarta etapa é responsável por atribuir características locais aos pontos fiduciais obtidas a partir do gradiente ao seu redor. Essas características são representadas de forma a não permitir que valores de luminância e forma não interfiram de forma acentuada na representação do gradiente.

O último método, abordado nesta tese, capaz de representar imagens através das características de forma é o código de cadeia (GONZALEZ; WOODS, 2002). O principal objetivo da representação através do código de cadeia é descrever o contorno de uma forma utilizando segmentos de reta imaginários que conectam pontos vizinhos do contorno. Para cada segmento de reta, estabelece um número, ou código, que representa a direção desse segmento.

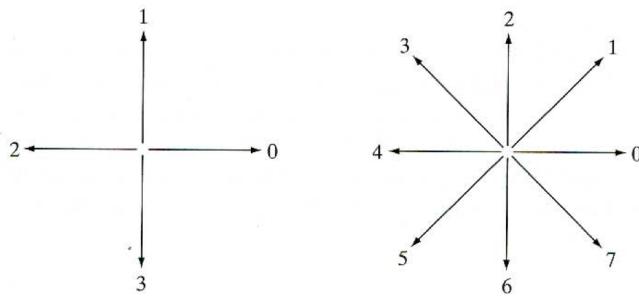


Figura 2.5: Discretizações para representação de forma através de código de cadeia. **Esquerda:** representação de direção utilizando 4 valores distintos. **Direita:** representação de direção utilizando 8 valores distintos. (GONZALEZ; WOODS, 2002)

De maneira mais prática, para se obter o código de cadeira, deve-se inicialmente estabelecer três propriedades essenciais do algoritmo: a discretização dos pontos do contorno, o sentido de varredura, a discretização das direções. O primeiro parâmetro estabelece a discretização dos pontos do contorno representado para geração dos segmentos de reta imaginários. Esta propriedade estabelece o quanto a imagem será sub-amostrada para que o algoritmo inicie a detecção dos pontos do contorno. Para que isso, o sentido de varredura, é uma propriedade que estabelece a ordem com que os segmentos de retas serão obtidos: horário ou anti-horário. O terceiro parâmetro estabelece o grau de discretização para a codificação da direção de cada segmento de reta. A Figura 2.5 ilustra duas possíveis discretizações. Na Figura 2.6, todo o processo de representação de forma através do código de cadeia é ilustrado.

Os métodos abordados nessa seção são estudados a partir de duas linhas de pesquisas diferentes. A primeira explora as características e a segunda explora o modelo de representação. Os modelos de representação podem ser ajustados para atividades específicas. Na proposta dessa tese, apresentamos um modelo de representação misto para reconhecimento de objetos em cenas naturais, utilizando algumas das características discutidas aqui. O modelo proposto apresenta flexibilidade, uma vez que mais características podem ser incorporadas para o aumento de precisão.

2.2 Aprendizagem de Máquina

A área de aprendizagem de máquina está cada vez mais presente em sistemas inteligentes. Com o surgimento da era do *Big Data* existem cerca de 1 trilhão de páginas na Web, fazendo com que a

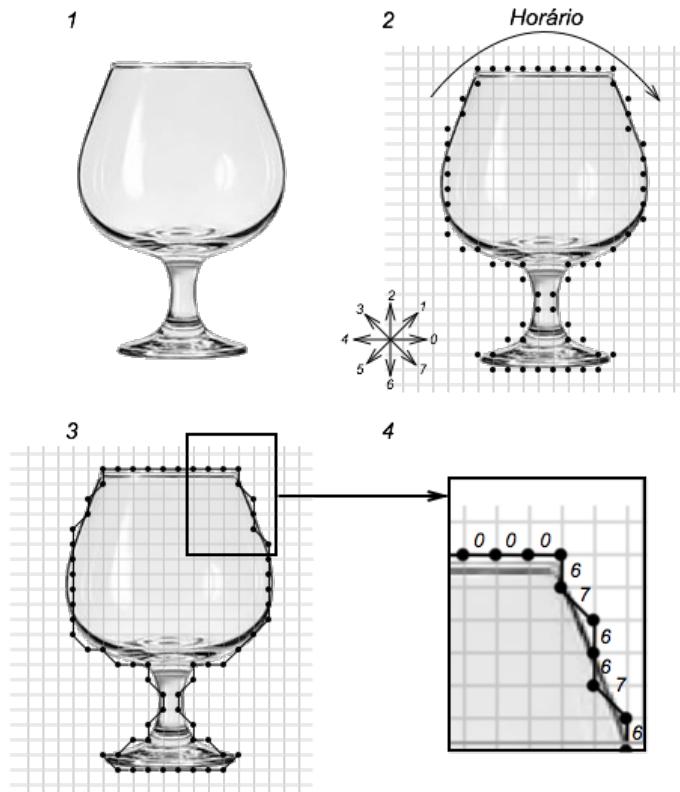


Figura 2.6: Exemplo de aplicação de código de cadeia sobre uma forma. **1:** Imagem original. **2:** Definição das propriedades do algoritmo e identificação dos pontos do contorno. **3:** Identificação dos segmentos de reta e rotulação. **4:** Detalhe da rotulação

demandando métodos de análise de dados também cresça da mesma forma.

Dentre os métodos de análise de dados, a aprendizagem de máquina é um importante arcabouço que permite aos pesquisadores detectar padrões de comportamento e até mesmo desenvolver modelos estatísticos previsíveis. Outro importante uso dessa área são os modelos de tomada de decisão baseados na incerteza.

A área de aprendizagem de máquina é normalmente dividida em 3 sub-áreas, distinguindo-se pelo seus modos de aprendizagem: supervisionados, não-supervisionados e aprendizagem por reforço.

Uma extensa fonte de informação sobre esta área pode ser encontrada em (MITCHELL, 1997; WERNICK et al., 2010; DUDA; HART; STORK, 2000).

2.2.1 Aprendizagem Supervisionada

A aprendizagem supervisionada é um conjunto de técnicas de aprendizagem de máquina que utiliza dados de treinamento para inferir uma função classificadora. Mais formalmente, um classificador supervisionado é uma função

$$\hat{g}(x) : X \rightarrow Y$$

onde X é o espaço de entrada e Y é o espaço de saída. Normalmente, o espaço de saída Y é o conjunto de classes $Y = \{1, \dots, c\}$, onde c é o número total de classes. A função $\hat{g}(x)$ é responsável por encontrar a classe esperada e, muitas vezes, toma a forma de uma maximização, como mostrado a seguir:

$$\hat{g}(x) = \arg \max_y f(x, y)$$

A função $f : X \times Y \rightarrow \mathbb{R}$ confronta uma entrada x com uma classe y e retorna um valor que representa a nota da classificação de x como pertencente à classe y . Assim, a classe esperada \hat{y} para a entrada x será dada pelo valor de y que maximiza a função $f(x, y)$.

Os algoritmos de aprendizagem supervisionada possuem um papel fundamental quando se deseja trabalhar com bases de dados de exemplos de treinamento. De fato, nesse tipo de metodologia, a base de exemplos atua como um “professor”, fornecendo custos ou classes esperadas, ajudando-o na busca de uma função $f(x, y)$.

Dentro do contexto de aprendizagem supervisionada, encontramos algoritmos que podem trabalhar em duas categorias de problemas: regressão e classificação. A regressão é uma técnica que relaciona variáveis dependentes e independentes através de uma função. De maneira mais específica, a regressão estuda como o valor de uma, ou mais variáveis, (variável dependente) variam em função de outra, ou outras variáveis.

Por outro lado, a classificação é o processo de aprender uma função que mapeia dados de entrada, x , em dados de saída y . O que se espera de um classificador é que ele consiga identificar um padrão em x que esteja associado à um valor em y .

Exemplos práticos de aplicação das duas categorias podem ser dados a partir de dois pro-

blemas: interpolação e reconhecimento de padrões. No primeiro caso, a interpolação estima uma valor desconhecido no conjunto de dados a partir de valores conhecidos. Tome como exemplo um nadador que percorre $d = 100$ metros em $t = 60$ segundos. Podemos interpolar este dado e descobrir um novo ponto no conjunto quando o $t = 30$. Nesse caso o nadador estaria em $d = 50$ metros.

Diferentemente da interpolação, o reconhecimento de padrões tenta associar um dado a uma categoria. Nessa abordagem podemos tomar como exemplo um filtro de spam de e-mails. Esta ferramenta tenta classificar um e-mail em dois grupos distintos: *spam* ou *não-spam*.

Na literatura, uma das primeiras técnicas em aprendizagem supervisionada para detecção de objetos utilizavam as *Redes Neurais* (YANG; SHU; SHAH, 2013; ROWLEY; BALUJA; KANADE, 1998; FUKUSHIMA, 1980; BEHNKE, 2003). A vantagem de se utilizar redes neurais está em sua habilidade de classificar dados não lineares, tais como aqueles sem correlação entre as variáveis, erro aleatório com média zero e homoscedasticidade. Contudo, as redes neurais são modelos auto-organizáveis; isto é, a partir de um conjunto de exemplos ou amostras, seu modelo fará ajustes para minimizar o erro. Isso gera dois pontos negativos. O primeiro ponto é que a complexidade do modelo torna-o difícil de se depurar, uma vez que o mesmo se torna uma “caixa-preta” (conjunto de pesos sinápticos). Outro problema é que não se pode incluir fatos conhecidos na rede sem que ela seja treinada por novas amostras (ou exemplos). O efeito disso é que deve-se estar muito atento à criação do conjunto de treinamento, uma vez que o mesmo deve representar de maneira global todas as classes.

Um exemplo clássico da aplicação de redes neurais pode ser encontrado em (FUKUSHIMA, 1980). Neste trabalho, o autor define um modelo, chamado de *Neocognitron*, para reconhecimento de padrões invariantes à posição em imagens digitais. O autor elabora um modelo multi-camadas e argumenta que esse modelo segue a organização cerebral dos vertebrados. No entanto, modelos de reconhecimento de objetos ou mesmo interpretação automática de cenas, inspirados na biologia, ainda são raros e apresentam muitos pontos em aberto, sendo portanto motivos de controvérsia na literatura.

Outra limitação importante sobre redes neurais é que sua implementação baseia-se em 2 etapas: treinamento e predição. A etapa de treinamento é feita para configurar os pesos sinápticos para que a rede “aprenda” a função de classificação. A segunda etapa, a predição, é utilizada para classificar os dados novos de fato. A separação destas duas etapas torna as redes neurais limitadas, pois não se pode aprender novos casos sem antes treiná-las novamente. Assim, em sistemas dinâmicos, sempre será necessário voltar à etapa de treinamento para reconfigurar a rede. A partir dessa limitação, criou-se o conceito de aprendizagem infinita (*Never-Ending Learning* (CARLSON et al., 2010)).

Em (CARLSON et al., 2010), os autores propõem um modelo computacional de aprendizagem infinita para extrair informações da *WEB*. Esse nome é dado pois espera-se que esse sistema permaneça aprendendo sempre que estiver operando. O modelo é constituído por um agente computacional inteligente capaz ler informações da internet e popular uma crescente base de conhecimento. Basicamente, esse sistema busca por ontologias; isto é, uma coleção de predicados que definem relações entre entidades. Cada predicado encontrado na web é então analisado por submódulos que dão uma probabilidade do mesmo estar correto. Por exemplo, se este predicado é encontrado com muita frequência, ele deve ser promovido à base de conhecimento.

Atualmente, está sendo muito utilizado em reconhecimento de objetos máquinas de vetores de suporte (SVM). O SVM é um classificador supervisionado baseado em busca de um hiperplano de separação que trabalha em duas classes, portanto esse é um classificador binário. O que diferencia o SVM de outros classificadores binários é que seu hiperplano de separação está localizado na região mais crítica de separação: as fronteiras. A função de maximização deste modelo cria um hiperplano que está localizado na maior distância entre os pontos mais próximos; isto é, há uma grande margem entre a fronteira das classes, tornando o erro da generalização, em geral, menor.

Em (LAN et al., 2013), é proposto um modelo para reconhecimento de objetos baseado em classificadores SVM. O objetivo do artigo é propor um arcabouço para reconhecer e rotular objetos em cenas. O artigo utiliza diversos classificadores SVM, cada um representando uma classe, para classificar os objetos em cena. Além disso, o processo de reconhecimento utiliza objetos já reconhecidos como informação extra para inferir outros objetos na cena. O problema

deste modelo é que deve-se ter um classificador SVM para cada classe de objetos. Para sistemas onde existem muitas classes, ter múltiplas instâncias de um classificador pode consumir excessiva memória computacional e tempo para o treinamento.

Outra teoria conhecida para reconhecimento de padrões em aprendizagem de máquina supervisionada é a teoria de Redes Bayesianas (NEAPOLITAN, 2003). Esse método baseia-se fundamentalmente em uma análise estatística do problema e tem sido amplamente utilizado por trabalhos recentes para reconhecimento de objetos.

Um exemplo de classificador utilizando redes bayesianas é encontrado em (VASCONCELOS; LIPPMAN, 1998), onde os autores propõem um sistema de recuperação de imagens baseado em inferência estatística. O trabalho mantém foco em três principais aspectos: deve ser um sistema intuitivo de recuperação; deve integrar diferentes modalidades de conteúdo; suportar aprendizagem estatística e, portanto, pode ser treinada automaticamente.

Um exemplo de trabalho que utiliza Redes Bayesianas para reconhecimento de objetos é o trabalho de (MENG et al., 2011). Neste trabalho, o autor utiliza a contagem de eventos atômicos para relacionar as probabilidades *a priori* e *a posteriori*, com a finalidade de identificar padrões em imagens. Além disso, a rede proposta é implementada em um dispositivo FPGA (Dispositivo programável de portas lógicas), comparando seus resultados com uma Rede Neural SOM. Tais resultados mostraram que a rede bayesiana utilizou menos recursos que a SOM, alcançando os mesmos resultados. Como trabalhos futuros, mais descritores de imagens devem ser incorporados ao modelo, como histograma de orientação de arestas. O autor argumenta que mais descritores podem melhorar o desempenho do sistema.

Recentemente, uma ferramenta que tem sido utilizada para modelar sistemas de reconhecimento de objetos são as chamadas Redes Complexas. Esses modelos têm sido inspirados devido a problemas atuais que só recentemente foram observados, tais como: sistemas biológicos (cadeias de DNA), sistemas sociais, redes de utilidade pública (elétrica, hidrovias, rodovias, aeroportos, etc...) e comportamentos de sistemas biológicos como o espalhamento de vírus e patologias, entre muitas outras. Uma revisão recente da área pode ser encontrada em (NEWMAN; BARABASI; WATTS,

2006), (NEWMAN, 2010) e (CALDARELLI; CATANZARO, 2012).

As redes complexas, conforme será abordado em mais detalhes na Seção 2.4, são basicamente grafos. Contudo, o seu poder de relacionamento de entidades, em conjunto com a área de sistemas complexos e estatística, permite o estudo das dinâmicas dos sistemas (NEWMAN; BARABASI; WATTS, 2006; ALBERT; BARABÁSI, 2002; BARABÁSI; ALBERT, 1999). Há tempos alguns sistemas eram por vezes estudados a partir de hipóteses e experimentos empíricos, ou por serem difíceis de se modelar ou por envolverem consumo excessivo de processamento computacional, o que os tornavam inviáveis. Um exemplo disso ocorre com (MILGRAM, 1967). Nesse trabalho, o autor fala sobre a organização da rede social e o problema do *mundo pequeno*. De maneira geral, o trabalho trata sobre a seguinte questão: dada uma rede de amizades, quantas pessoas, no máximo, são necessárias para conectar quaisquer duas outras?

Esse estudo foi feito de maneira empírica, porém não computacionalmente, e sim de fato, pois foram enviadas cartas a pessoas remotas com mensagens informando o nome de destinatários finais que deveriam receber essas mensagens. Cada pessoa que recebia essa mensagem informava seu nome e passava para outra pessoa que acreditava conhecer o destinatário final. O experimento indicou que existem 6 passos intermediários, em média. Porém, o estudo foi conduzido por pessoas que vivem somente nos Estados Unidos e haviam diversas limitações que poderiam invalidar o resultado final do experimento. Recentemente, trabalhos tratando do mesmo problema foram publicados graças ao surgimento das Redes Complexas (NEWMAN; BARABASI; WATTS, 2006; WATTS; STROGATZ, 1998; WATTS, 2004).

Atraídos pelos resultados encontrados na área social, os cientistas da área cognitiva (SPORNS, 2002), linguística (CANCHO; SOLÉ, 2001) e visão computacional incorporaram as redes complexas em seus estudos (TANG et al., 2012; COSTA, 2009; ZHANG et al., 2012; BACKES; CASANOVA; BRUNO, 2013a; CUADROS et al., 2012; BACKES; CASANOVA; BRUNO, 2013b; CASANOVA; BACKES; BRUNO, 2013). Em (BACKES; CASANOVA; BRUNO, 2013b), os autores propõem uma nova representação computacional de texturas de imagens para servir como descritores de texturas. De uma forma genérica, texturas são definidas como uma estrutura bidimensional de *pixel*. Nesse artigo, os autores representam cada *pixel* da textura através de um nó da rede complexa. A conexão entre

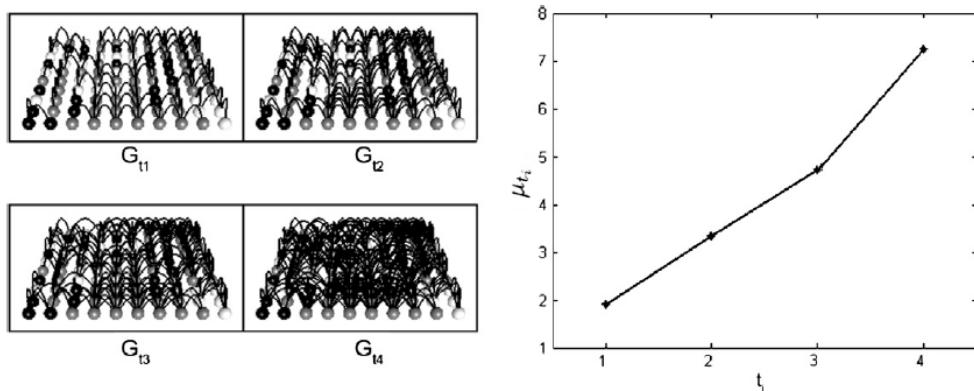


Figura 2.7: Esquerda: Visualização de uma rede complexa com diferentes valores de t . **Direita:** Grau médio da rede para cada valor t (BACKES; CASANOVA; BRUNO, 2013b)

dois nós é feita sempre que a distância euclidiana entre dois *pixels* for menor que um valor r , sendo representada por uma aresta não direcionada. Além disso, essas arestas possuem um peso que está relacionado à diferença da intensidade de cores entre os dois *pixels* que ela conecta. De posse dessa estrutura, o autor pode extrair diversas características como conectividade média, grau médio, coeficiente de clusterização médio, entre outras. Contudo, o trabalho vai além da extração de características em uma rede estática. Para se criar dinamismo à rede, o autor elimina gradativamente arestas que possuem pesos menores que uma variável t . Ao variar t , obtém-se uma sub-rede com menos arestas, porém representando ainda a mesma textura. Ao medir novamente as características da nova rede, é possível analisar o seu dinamismo. Cada textura irá produzir diferentes comportamentos, sendo essa diferenciação que pode ser utilizada como descritor de texturas.

A Figura 2.7 ilustra a evolução e a variação do grau médio de uma rede complexa para valores crescentes de t . Observa-se na figura da direita que o gráfico do grau médio da rede aumenta a medida que t aumenta. Essa curva é vista como um descritor da textura. O que se obtém nesse tipo de modelagem é uma rede que relaciona a diferença de intensidades de cores para cada par de *pixels* próximos. Os autores desenvolvem diversas associações de redes complexas com texturas e propõem uma assinatura própria para cada textura. Os resultados mostraram que os descritores são robustos o suficiente para discriminar as texturas em suas respectivas classes, superando métodos tradicionais.

Resultados similares foram encontrados em (CASANOVA; BACKES; BRUNO, 2013). Nesse ar-

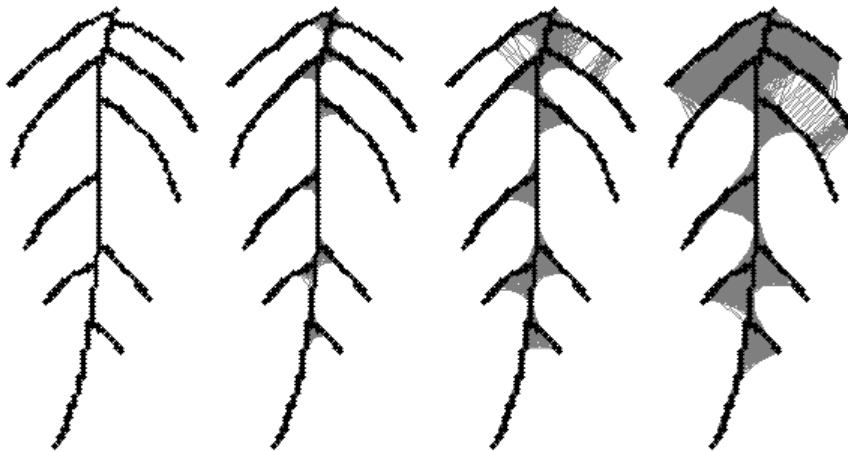


Figura 2.8: Representação das nervuras de uma folha utilizando a evolução de redes complexas(CASANOVA; BACKES; BRUNO, 2013).

tigo, os autores estão interessados em desenvolver um modelo genérico o bastante para caracterização de sinais, curvas e conjuntos de pontos. O artigo é apresentado e validado utilizando como apoio o problema de reconhecimento de tipos de plantas; mais especificamente, identificando plantas a partir da análise de suas folhas. Nesse caso, a rede complexa é construída a partir dos pontos discretizados das nervuras das folhas; isto é, cada nó da rede complexa representa um ponto da nervura de uma folha. Inicialmente, todos os pontos são conectados entre si através das arestas, onde o peso de cada aresta representa a distância euclidiana entre os dois pontos. Esses pesos são então discretizados entre 0 e 1 e passam por um processo de limiarização. Nesse processo, diversos limiares são aplicados à rede com o objetivo de extrair informações de sua dinâmica. Aqui é utilizado o mesmo processo proposto em (BACKES; CASANOVA; BRUNO, 2013b), o dinamismo da rede complexa é gerado ao remover arestas que possuem pesos abaixo de um limiar variável t . A Figura 2.8 apresenta a evolução de uma rede complexa de acordo com a variação desse limiar. De posse da evolução dessa rede, o trabalho extrai, para cada valor de limiar, o grau máximo e o grau médio da rede, compondo um vetor de características. Para cada planta, o processo é repetido e são gerados diversos vetores de características. Finalmente, o trabalho aplica a técnica do LDA para separar linearmente as classes. A taxa de classificação obtida pelos experimentos foi de 93%. Os autores sugerem que esse método pode ser utilizado para outros problemas de classificação.

Outros artigos, tais como (CHALUMEAU L. DA F. COSTA, 2012; BACKES; BRUNO, 2010; BACKES; CASANOVA; BRUNO, 2009) mostram mais exemplos na literatura onde as redes complexas são

utilizadas. De forma geral, a maioria destes artigos modela um pixel ou uma região da imagem através dos nós e estabelecem o relacionamento entre eles através de arestas. Esses relacionamentos podem conter informações de cores, formas e texturas.

O uso de redes complexas para modelos genéricos é promissor e traz novas possibilidades para a área de reconhecimento de objetos.

Esta seção apresentou três técnicas baseadas em aprendizagem supervisionada: redes neurais, redes bayesianas e redes complexas. Existem outras técnicas que se enquadram nessa categoria de aprendizagem, porém não serão abordados nesta tese, uma vez que o objetivo principal aqui é entender sua evolução na literatura de uma maneira mais ampla.

2.2.2 Aprendizagem Não Supervisionada

Dentre as técnicas de aprendizagem de máquina, a aprendizagem não supervisionada utiliza a própria estrutura estatística do conjunto de entrada de dados como uma base para discriminar padrões. Desse modo, diferentemente da aprendizagem supervisionada, não se tem um valor de saída esperado para cada entrada (DAYAN; SAHANI; DEBACK, 1999).

Um dos primeiros trabalhos que apresentou a aprendizagem não supervisionada é o de David Marr (1970), também encontrado em (VAINA, 1991). Neste trabalho, Marr propõe teorias computacionais baseadas no *neocortex* cerebral. Uma de suas teorias diz que as células do *neocortex* cerebral são classificadores flexíveis; ou seja, elas aprendem a estrutura estatística dos padrões de entradas a medida que combinações repetidas de padrões são apresentadas. A partir dessa constatação, os cientistas puderam concentrar seus esforços para criar novas técnicas de aprendizagem que fossem capazes de aprender sem qualquer tipo de supervisionamento. Marr publicou outros trabalhos apresentando teorias que continuam a influenciar modelos matemáticos modernos.

De forma mais explicativa, podemos dizer que a aprendizagem não supervisionada deve estimar um modelo probabilístico dos dados. Por exemplo, considere um sistema de previsão de tempo. Nesse sistema, deve-se inferir um modelo probabilístico que represente a distribuição de

probabilidade de um novo valor x_t a partir de dados passados x_1, \dots, x_{t-1} . O que temos no final da aprendizagem é o modelo de $P(x_t|x_1, \dots, x_{t-1})$.

Uma técnica derivada da aprendizagem não supervisionada é a clusterização. Esta técnica é responsável por encontrar, de maneira automática, padrões nos dados de entrada para a formação de grupos (ou *clusters*), de tal forma que objetos em um mesmo grupo são mais semelhantes que objetos entre grupos distintos.

De uma maneira mais formal, muitos algoritmos de clusterização são iterativos. A cada iteração, a clusterização é submetida a uma equação que avalia sua qualidade. A Equação (2.8) mostra um exemplo de como avaliar uma clusterização.

$$J = \sum_{j=1}^k \sum_{i \in C_j} \|x_i - \mu_j\|^2 \quad (2.8)$$

Na Equação (2.8), k é o número de *clusters*, C_j é o conjunto de elementos pertencente da mesma classe e $\|x_i - \mu_j\|$ é uma medida de distância que relaciona o elemento x_i do cluster C_j ao centro μ_j do cluster C_j .

O que se espera de um algoritmo de clusterização é que a distância $\|x_i - \mu_j\|$ seja a menor possível os elementos dentro de um mesmo cluster. Assim, os algoritmos de clusterização muitas vezes utilizam técnicas de minimização e aplicam a Equação (2.8) como uma função objetivo. A cada arranjo dos elementos nos grupos, o algoritmo testa a qualidade da clusterização utilizando esta equação.

Outro tipo de aplicação onde a aprendizagem não supervisionada é utilizada é a redução de dimensionalidade. Considere um sistema capaz de classificar se uma pessoa está doente ou não. O sistema é alimentado através de um prontuário contendo as seguintes informações: tipo sanguíneo, número de plaquetas no sangue, idade da pessoa, tom de pele e hora da internação. Dessas 5 características pode-se retirar algumas que não estão relacionadas à presença de doença como, por exemplo, o tipo sanguíneo, tom de pele e hora da internação. As duas variáveis restantes (número de plaquetas e idade da pessoa) estão mais relacionadas às doenças e são as que mais contribuem para a classificação. Ao utilizar um algoritmo de clusterização, somente as duas variáveis que

nesse tipo de problema que muitas vezes a redução de dimensionalidade é utilizada.

Existem muitos métodos capazes de fazer a redução de dimensionalidade, cada um com abordagens diferentes. Um método bem conhecido baseia-se na decomposição SVD (*Singular Value Decomposition*), que basicamente fatora uma matriz M em três novas matrizes: U , Σ e V . O SVD é então uma transformação linear da matriz original M , de tal forma que:

$$M = U\Sigma V$$

Pode-se interpretar a equação acima através de 3 operações lineares. A partir de uma matriz com vetores unitários V , um escalonamento Σ e uma rotação final U pode-se obter M . Uma observação importante deve ser feita sobre a matriz Σ . Uma vez que esta matriz faz uma operação de escalonamento sobre V , os maiores valores de Σ estão associados às dimensões que apresentam maiores variâncias na matriz M . Isso significa que a matriz Σ pode dar informações sobre as dimensões mais importantes nos dados de entrada. É nesse ponto que podemos descobrir quais dimensões são as mais importantes para separar os dados em grupos. Muitos trabalhos utilizam as dimensões com maiores variâncias como entrada em classificadores.

Contudo, qualquer método de redução de dimensionalidade deve ser utilizado com cautela, uma vez que os dados de entrada devem ser tratados para que haja uma proporção correta entre as dimensões. Tome como exemplo o sistema classificação anterior de doentes. A escala de medida para contagem de plaquetas no sangue em uma pessoa saudável varia entre 150.000 a 450.000 por mililitro cúbico, enquanto a idade da pessoa varia entre 0 a 100 anos aproximadamente. Evidentemente, não se pode tentar fazer uma redução de dimensionalidade sem antes normalizar estes dados para uma faixa comum, caso contrário isso levaria a matriz de escalonamento Σ a ter valor maior sobre a dimensão de número de plaquetas do sangue, por conter números de maior grandeza.

Além da normalização, um ponto importante deve ser discutido. Atualmente não se conhece um método geral que informe quantas dimensões são suficientes para representar o conjunto de entrada de forma fidedigna e sem interferência no classificador. Assim, muitos trabalhos recorrem a métodos empíricos a fim de observar o efeito da dimensionalidade sobre a classificação dos dados.

Há diversas vantagens em se utilizar técnicas de aprendizagem não supervisionada. A primeira delas está relacionada com a base de dados. Quando trabalhamos com técnicas não supervisionadas, podemos dispensar o uso de bases supervisionadas e simplesmente treinar o classificador com dados reais. Essa técnica é interessante principalmente para problemas que demandam por grandes bases de dados. Muitas vezes não temos uma base de dados supervisionada grande o suficiente para treinar um classificador, pois poderia ser muito trabalhoso criar uma base que conseguisse representar genericamente um problema. Assim, se um classificador supervisionado fosse treinado com uma base de tamanho menor que o necessário (ou não genérica), o mesmo poderá se tornar muito especializado aos dados de treinamento e não ser genérico o suficiente para classificar novos dados de entrada. Isso mostra como os classificadores não supervisionados podem ser úteis como um ferramental para se trabalhar com grandes bases de dados não supervisionadas.

Uma outra vantagem de aprendizagem não supervisionada é que podemos treinar um classificador não supervisionado, encontrando as principais classes de dados e, posteriormente, utilizar técnicas supervisionadas para identificar as classes encontradas (DUDA; HART; STORK, 2000). Tome como exemplo um reconhecedor de caracteres (OCR). Podemos apresentar diversos padrões das letras do alfabeto sem qualquer supervisão e separá-los em 52 grupos (26 pares de letras maiúsculas e minúsculas). Após essa separação, podemos usar técnicas de supervisão para rotular os grupos encontrados apenas comparando um elemento de cada grupo com um caractere supervisionado.

Além das duas vantagens apresentadas anteriormente, podemos acrescentar outros benefícios de se utilizar técnicas não supervisionadas. Muitas vezes, lidamos com sistemas dinâmicos em que o meio externo está em constante mudança. Nesses casos, utilizar uma técnica de aprendizagem que possa se adaptar a essas variações é essencial para o modelo. Para esses casos, a aprendizagem não supervisionada também pode ser utilizada para fazer constantes ajustes, a medida que o ambiente se altera.

Finalmente, a aprendizagem não supervisionada pode ser utilizada como uma ferramenta para encontrar características discriminantes nos dados de entrada. Frequentemente, nos deparamos com bases de dados contendo muitas dimensões (ou características). Para esses tipos de bases, escolher quais características considerar no modelo do sistema é uma tarefa inherentemente

difícil. Ao utilizar técnicas de classificação não supervisionadas, podemos obter as características que mais separam, ou discriminam, os dados de entrada.

Por outro lado um ponto negativo da aprendizagem não supervisionada é, a excessiva generalidade dos algoritmos, que pode limitar a performance em uma atividade específica. Uma técnica não supervisionada utiliza noções gerais para identificar padrões ou encontrar estruturas interessantes nos dados de entrada. Isso significa que qualquer especialização necessária para algum tipo de entrada não será possível de se treinar.

Outro ponto negativo é que, como seu método de aprendizagem não é guiado (ou supervisionado), a presença de erros na base de dados pode induzir a aprendizagem a um caminho indesejado. Suponha um sistema que faça o controle de qualidade em uma linha de produção. Se, durante o treinamento, houve erro de leitura dos sensores, esse erro poderá ser “aprendido”, o que pode trazer resultados indesejáveis durante a classificação de novos dados.

Na literatura, diversos trabalhos recentes utilizam a aprendizagem não supervisionada para área de visão computacional e reconhecimento de objetos em cenas. Nessa linha de pesquisa, (BO; REN; FOX, 2012; VINCENT et al., 2008; COLLINS; SINGER, 1999; LEE et al., 2009; COATES; NG, 2011; BO; REN; FOX, 2011) são exemplos de trabalhos que utilizam esta técnica.

Um dos problemas envolvidos na área de reconhecimento de objetos é descobrir quais características (ou dimensões) nos dados de entrada são fundamentais para separá-los em grupos. A essas características, damos o nome de *características discriminantes*.

Em (BO; REN; FOX, 2011, 2012), os autores propõem um método para encontrar características discriminantes utilizando o *K-SVD* (AHARON; ELAD; BRUCKSTEIN, 2006), que é um decompositor de sinais de tal forma que um sinal y pode ser reconstruído através de uma matriz D de tamanho reduzido e de um vetor de coeficientes x , conforme a seguinte equação:

$$y \simeq D.x$$

onde D é chamada de matriz dicionário e x contém os coeficientes de representação do sinal y . Em (BO; REN; FOX, 2011), os autores utilizam essa ferramenta para descobrir um dicionário a partir de



Figura 2.9: Reconstrução de uma imagem a partir de uma matriz de dicionário. **Esquerda:** Imagem original, **Centro:** Imagem reconstruída com $k = 2$. **Direita:** Imagem reconstruída com $k = 5$

diversas grades de *pixels* obtidas de imagens coloridas com informações de profundidade. Assim, um dicionário para uma grade de 5×5 *pixels* irá formar vetores de tamanho $5 \times 5 \times 8$, onde o último componente é responsável por guardar informações de tons de cinza, RGB, profundidade e normais das superfícies. Contudo, o trabalho treina cada característica separadamente, havendo um dicionário para cada tipo de característica.

Um ponto em aberto deixado pelo trabalho é o valor de K escolhido para os experimentos. É mostrado, visualmente, que um dicionário aprendido com $K = 5$ pode ser bem próximo da imagem original, veja Figura 2.9. Contudo, esta afirmação não pôde ser esclarecida matematicamente e nem uma justificativa para esse valor foi discutida além da similaridade na aparência. Assim, não se sabe se a quantidade de dados perdida pela reconstrução foi prejudicial ao algoritmo de reconhecimento.

A escolha por um valor de K (ou quantidade de grupos) ideal ainda é um problema em aberto na comunidade científica. Em (ERDMANN et al., 2013), os autores utilizam uma meta-heurística chamada *Firefly* para encontrar valores de limiares para segmentação de imagens. Uma base de dados de imagens segmentadas manualmente foi utilizada para comparar cada supervisão humana com a segmentação. Os resultados mostraram que existe um valor de K ideal associado aos valores de limiares que aproxima a segmentação das imagens às imagens supervisionadas. Isso significa que o valor de K (ou quantidade de limiares) é tão importante quanto os próprios valores dos limiares.

Alguns trabalhos como (ROSE; GUREWITZ; FOX, 1990; TEMEL, 2014; NEWELL et al., 2013;

YU; LIU; WANG, 2014) estudam métodos capazes de computar o valor de K automaticamente. Em (YU; LIU; WANG, 2014) os autores propõem um método bayesiano associado à na teoria dos conjuntos aproximativos (ou *rough set model*) para clusterização de dados. De uma maneira geral, o trabalho cria um modelo capaz de representar estatisticamente a tomada de uma ação α sobre um objeto O e uma classe C com respeito a três possíveis ações. A primeira delas é classificar $O \in C$. A segunda é classificar $O \notin C$. A última ação, é classificar O como um objeto na fronteira da classe C . O algoritmo para determinação de K é desenvolvido utilizando o conceito bottom-up; isto é, o algoritmo assume inicialmente que cada objeto está contido em sua própria classe. A cada passo iterativo, o algoritmo compara a similaridade entre dois objetos em em relação à similaridade média do conjunto de dados. Caso haja algum par de objeto entre *clusters* diferentes que tenham similaridade menor do que a similaridade média entre todos os pares de objetos, haverá uma junção entre estes *clusters*. O algoritmo termina assim que não haja mais qualquer par de objetos em *clusters* diferentes que tenham similaridade menor do que a similaridade média entre todos os pares.

Outros métodos para o cálculo de K são baseados em modelos estatísticos. Em (TEMEL, 2014), o autor propõe um modelo estatístico baseado em similaridade para encontrar as fronteiras entre os *clusters*. Inicialmente, uma medida de similaridade entre dois objetos i, j é feita utilizando uma função gausiana. Para cada objeto i da massa de dados é calculada uma distribuição de probabilidade p_i . Esta distribuição é construída medindo a distância entre i e todos os outros objetos da base. Para que p_i seja uma distribuição de probabilidade, o autor normaliza a distribuição de tal forma que $\sum p_i = 1$. Após este processo, a entropia é calculada sobre todos os vetores p_i . O resultado de p_i será baixo caso todas as distâncias computadas através de i sejam muito próximas ou muito distantes. Contudo, p_i será alto caso as distâncias sejam muito diferentes entre si. Assim, os pontos que estão na fronteira dos *clusters* terão o valor entrópico maior e os centros dos *clusters* terão valor entrópico menor. Finalmente, o autor sugere que o número de *clusters* seja o mesmo número de vales sobre essa função entrópica. Apesar do autor ter utilizado entropia, o modelo estatístico parece não seguir algumas leis da teoria da informação. Há mais de um século, sabe-se, por exemplo, que a entropia é uma função utilizada para medir o caos entre os estados de

um sistema. Assim, para a modelagem proposta em (TEMEL, 2014), se o conjunto de dados fosse regular, a entropia deveria ser máxima.

Apesar da literatura propor novas maneiras de computar o valor de K , ainda não há um método suficientemente preciso e aceito pelos cientistas. Além disso, não se sabe exatamente como medir a qualidade de um cluster. Isso deve-se ao fato de que a clusterização está relacionada à intuição humana. Por isso, muitos trabalhos assumem o valor de K empiricamente, deixando em aberto discussões mais profundas relacionadas com a cognição humana.

Em (MESNIL et al., 2013) são utilizadas outras técnicas de redução de dimensionalidade como PCA e CAEs (Contractive Auto-Encoders) (RIFAI et al., 2011) para tratar grandes vetores de características de objetos para classificação de cenas. Os autores utilizam dados obtidos a partir de diversos extratores de características e reduzem a dimensionalidade dos dados aplicando técnicas não supervisionadas, tal como o PCA. Os resultados mostram que a técnica pode ser relevante para a área, uma vez que conseguiu melhorar a performance da classificação em até 10% em relação aos trabalhos recentes. Os autores destacam que a redução de dimensionalidade sobre os vetores de características estudados conseguem extraír dados de alto nível (mais representativo) eliminando-se a dependência linear entre os atributos. Apesar do trabalho ter alcançado resultados satisfatórios, foram necessários diversos experimentos para se chegar a uma arquitetura que apresentasse os melhores resultados. Assim, é necessário o uso de uma técnica supervisionada para aplicar o PCA a cada classe.

Além dos problemas específicos discutidos até aqui, destacamos mais dois problemas comuns na área de aprendizagem não supervisionada. O primeiro deles está relacionado ao vetor de características modelado. Muitos trabalhos, como é o caso de (MESNIL et al., 2013), controem um vetor de características a partir de diferentes características heterogêneas. Dependendo da técnica utilizada, essa modelagem pode trazer efeitos indesejáveis à modelagem, uma vez que o algoritmo de otimização pode apresentar restrições quanto às variáveis. Uma dessas restrições, a normalização das dimensões, foi discutida anteriormente. Nesse caso específico, o modelo poderá se tornar inválido devido a mistura dos tipos de variáveis, tais como: discretas e contínuas, qualitativa e quantitativa, nominal e ordinal entre outras. Portanto, a mistura entre tipos de variáveis em

um mesmo vetor de características deve ser analisada cuidadosamente para que não haja violação das restrições.

Outro tipo de problema ocorre quando o método utilizado não está adaptado para sistemas dinâmicos. Normalmente, os métodos de aprendizagem supervisionada e não-supervisionada trabalham em três estágios. O primeiro é a aprendizagem, o segundo é a validação e o terceiro é a classificação. Contudo, para sistemas dinâmicos, não se pode treinar o sistema em um determinado momento e classificar em outro momento. Nesse caso, a aprendizagem deve ser “on-line”; ou seja, o treinamento ocorre conjuntamente com a classificação.

Uma aplicação que demanda por um sistema de aprendizagem e classificação de forma conjunta é a navegação de robôs em ambientes dinâmicos. Alguns trabalhos, como em (HADSELL et al., 2007), consideram que o ambiente onde o robô, ou agente, navega contém obstáculos que mudam de posição durante a fase de classificação. Assim, o ambiente em que eles interagem é dinâmico e necessita de um aprendizado adaptativo e constante em termos temporais.

Esta seção destacou alguns pontos positivos e negativos sobre aprendizagem não supervisionada. Na literatura científica, na área de reconhecimento de objetos, frequentemente são encontrados trabalhos utilizando aprendizagem não supervisionada como uma técnica para redução de dimensionalidade do vetor de características dos objetos. Contudo, essa prática pode ter efeitos indesejáveis como eliminação excessiva de dados, valor de corte ideal desconhecido e limitações devido à junção de características distintas em um mesmo vetor de características.

2.2.3 Aprendizagem por Reforço

Uma terceira técnica para aprendizagem de máquina é conhecida como *reinforcement learning*, ou aprendizagem por reforço. Esta técnica baseia-se no relacionamento entre recompensas e ações feitas por agentes em um ambiente. Portanto, esta técnica de aprendizagem está mais relacionada com as interações de um agente com o seu ambiente de operação.

Como descrito em (SUTTON; BARTO, 1998), a aprendizagem por reforço é uma técnica que está intrinsecamente relacionada à definição do problema a ser resolvido, ao invés da elaboração

de algoritmos para resolvê-lo. Assim, a modelagem de problema que será resolvido com aprendizagem por reforço é feita de acordo com um ou mais objetivos. A partir da definição desses objetivos, alguns algoritmos utilizam conceitos oriundos da psicologia, como é o caso de “reforço”, para alcançar o objetivo.

Na área da psicologia, Ivan Pavlov define que o reforço é um estímulo dado a um indivíduo com o objetivo de associar e consolidar a resposta gerada a partir de uma ação (SAUNDERS, 2006). Para os humanos, o reforço pode ser gerado a partir dos estímulos senso-motores ou até mesmo de outros indivíduos.

Um exemplo prático de reforço pode ser observado em crianças desenvolvendo a habilidade de andar. Nota-se que a aprendizagem acontece a medida que a criança testa diversas combinações de movimentos, ganhando mais equilíbrio com o passar do tempo. Nesse caso, as quedas agem como reforços negativos e são os estímulos que a criança precisa para não repetir o que havia testado anteriormente.

Na área da computação, este princípio se mantém. O agente aprende através de tentativas e erros e a informação de reforço é processada a cada nova tentativa. As técnicas de aprendizagem por reforço têm por objetivo mapear situações (ou estados) para cada possível ação que um agente pode realizar.

Dentro do contexto de aprendizagem por reforço alguns conceitos são importantes para compreender diferentes métodos. O *objetivo* é um estado do agente que deve ser alcançado. Uma *política* define uma sequência de ações do agente na tentativa de se alcançar um *objetivo*. A *função de recompensa* mapeia estados (ou pares de estados-ações) em números que representem a *recompensa* por um agente estar em um estado (ou tomar uma determinada ação em um determinado estado). A *política ótima* é uma política que maximiza a *função de recompensa*.

Há duas formas de encontrar uma política ótima. A primeira delas é a chamada *on-line* e tem características exploratórias. A técnica *on-line* parte de uma política inicial (aleatória ou não) e a maximiza iterativamente até chegar à política ótima. Por outro lado, a técnica *off-line* descobre uma política ótima sem se basear em outra política. Assim, essa última é útil quando deseja-se

treinar um sistema sem caráter exploratório e utilizá-lo em um momento posterior.

Muitos trabalhos que implementam a aprendizagem por reforço utilizam o modelo *MDP* (*Markov Decision Process*) como um ferramental matemático para descrever o sistema estudado. O *MDP* é uma tupla $\langle S, A, \delta, \tau \rangle$, onde S é o conjunto de estados, A é o conjunto de ações, δ é a função de transição e τ é o conjunto de recompensas. De posse desse modelo, os trabalhos utilizam técnicas de otimização para encontrar o conjunto de ações (política ótima) para que se maximize a recompensa.

Alguns trabalhos, como em (BANDERA et al., 1996; DRAPER; BINS; BAEK, 1999; MINUT; MAHADEVAN, 2001; PIÑOL et al., 2012), utilizam a aprendizagem por reforço na área de reconhecimento de objetos em imagens. Em (DRAPER; BINS; BAEK, 1999), os autores propõem um modelo adaptativo para detecção de tipos de moradias. Essa detecção é feita a partir de imagens aéreas e são estudados 4 tipos de casas. O trabalho utiliza um modelo *MDP* e o *Q-learning* como técnica de otimização para definir um conjunto de módulos de visão computacional para reconhecimento de objetos. A metodologia proposta tem como objetivo encontrar a melhor combinação entre os módulos definidos que possa levar o sistema a obtenção da melhor performance possível. Neste caso, a função de recompensa está associada à qualidade da detecção. Sendo assim, foi necessária um base de dados supervisionada. A partir desse meta-modelo, os autores desenvolvem 4 modelos definitivos para detecção de 4 tipos de casas. Os resultados mostraram que o trabalho encontrou modelos que se aproximam dos modelos ótimos para detecção de cada classe de casa. Contudo, para a criação desse meta-modelo é necessário que haja uma biblioteca de procedimentos de visão computacional, tais como: correlação, detecção de picos, segmentação, modelos deformáveis entre outros. Porém, criar uma biblioteca suficientemente grande para detectores genéricos é uma tarefa difícil. Assim, no caso deste trabalho, que tem por objetivo detectar somente casas, a modelagem se torna viável pois é possível trabalhar em um subconjunto seletivo de procedimentos e técnicas de visão computacional.

Outros exemplos de aplicações de aprendizagem por reforço na área de visão computacional são encontrados em (PALETTA; PINZ, 2000; PALETTA; FRITZ; SEIFERT, 2005). Em (PALETTA; PINZ, 2000), um sistema ativo de reconhecimento de objetos é proposto utilizando aprendizagem

por reforço. A ideia do trabalho é encontrar o melhor ponto de visão em uma cena para que se tenha maior discriminação dos objetos. Inicialmente, o trabalho apresenta um modelo utilizando funções de base radial capaz de aprender a aparência visual dos objetos em diferentes pontos de visão. Para isso, algumas características são extraídas e projetadas em um espaço 3-dimensional. Após o modelo ser aprendido, ele é então utilizado como uma função de utilidade capaz de quantificar a hipótese de presença de objetos na cena. De posse desse modelo, os autores criam um *MDP* para representar os processos de decisão para que se tenha o melhor ângulo de visão possível. Os estados desse modelo representam a probabilidade de se discriminar os objetos naquele ponto de visão, as ações são os movimentos possíveis do ponto de visão e a recompensa é encontrada a partir da função de utilidade. O trabalho é validado através de um experimento composto por um sistema senso-motor com uma câmera. Os resultados sugerem que o *Q-Learning* é um algoritmo promissor para detecção de objetos em cenas 3D e sugere que é um modelo promissor para a área de reconhecimento de objetos. Como trabalhos futuros, deve-se estudar formas alternativas para a função de utilidade, uma vez que há muita dependência entre os resultados e a função de utilidade. Outro ponto em aberto é que a modelagem dos objetos em um subespaço pode encontrar os mesmos problemas discutidos na Seção 2.2.2.

Em (PIÑOL et al., 2012), onde os autores proproem um método de aprendizagem por reforço com o objetivo de selecionar as melhores características em imagens para que um classificador tenha a maior performance possível. O trabalho utiliza um *MDP* como modelo de representação dos processos de decisão. Os autores propõem que os estados sejam os conjuntos de características que descrevem a imagem de entrada, tais como: cor mediana, cor média, número de cores, entre outras. O conjunto de ações do modelo *MDP* representa as escolhas que o sistema poderá fazer para melhorar o processo de reconhecimento de objetos da cena. Assim, o conjunto *A* do *MDP* foi definido como 4 possíveis extractores de características: *SIFT (Scale-Invariant Feature Transform)* (LOWE, 2004), *SPIN* (LAZEBNIK; SCHMID; PONCE, 2005), *SURF (Speeded Up Robust Feature)* (BAY et al., 2008) e *PHOW (Pyramid Histogram Of visual Words)* (BOSCH; ZISSERMAN; NOZ, 2007). A função de transição δ foi ligeiramente alterada no trabalho. Em vez desta função retornar um novo estado, ela retorna a nova representação da imagem após aplicar uma ação α . Finalmente, a função de

recompensas τ retorna recompensas altas sempre que o classificador optar pela classe correta do objeto, portanto é necessário o uso de um *ground truth*. Dada esta modelagem, o autor encontra a política ótima utilizando o *Q-Learning*(SUTTON; BARTO, 1998). Após o treinamento, o sistema estará apto a escolher corretamente os extratores de características a partir de uma imagem de entrada e enviar para um classificador. Apesar desse trabalho apresentar uma abordagem interessante, não se pode garantir que para qualquer imagem o sistema irá extraír as melhores características, uma vez que o conjunto S é descrito a partir de características genéricas. Outro ponto negativo é que a base de treinamento deve ser suficientemente grande para generalizar todo o universo de imagens.

Um vantagem em se utilizar a aprendizagem por reforço é que o problema pode ser descrito a partir de um modelo *MDP*. Isso significa que devemos apenas definir os estados e ações do sistema, as transições e criar uma função de recompensa que esteja relacionada ao objetivo. Assim, o foco do modelo está muito mais ligado à função recompensa que propriamente uma saída esperada.

No entanto, o uso de métodos iterativos para encontrar a política ótima têm algumas desvantagens. A primeira delas está relacionada com a função de recompensa, que deve ser avaliada a cada passo da iteração. Muitas vezes, como é o caso dos trabalhos abordados nesta seção, a função de recompensa é computacionalmente cara. Isso significa que o uso de outras técnicas e modelos, como estrutura de dados avançadas, devem ser utilizados para diminuir a complexidade computacional.

Outra desvantagem em utilizar aprendizagem por reforço é a dificuldade em encontrar parâmetros ótimos para o modelo. Por exemplo, muitos algoritmos utilizando esta técnica possuem um parâmetro associado à taxa de aprendizagem. Se este parâmetro for muito pequeno, o sistema pode demorar para convergir e, em alguns casos, um sistema aleatório pode convergir mais rapidamente que a aprendizagem por reforço. Se a taxa de aprendizagem for muito alta, poderá ocorrer uma especialização nos últimos dados de entrada do sistema. Isso poderá levar a um sistema muito sensível e instável. Assim, um valor ótimo para a taxa de aprendizagem normalmente é encontrado empiricamente. Alguns trabalhos, como é o caso de (DRAPER; BINS; BAEK, 1999), compararam o sistema de aprendizagem com um sistema aleatório na busca de convencer o leitor de

que o modelo converge de maneira mais rápida.

As técnicas estudadas até aqui mostraram alguns recursos para a modelagem de sistemas capazes de identificar padrões a partir de uma massa de dados. Essas técnicas foram apresentadas como ferramentas de aprendizagem de máquina. Na Seção 2.3, apresentaremos alguns aspectos que devem ser levados em consideração quando deseja-se incorporar ao modelo características relacionadas à consciência.

2.3 Influência da Neurociência nos Modelos de Visão Computacional

A inteligência artificial sempre buscou criar modelos inspirados nos seres humanos adotando a cognição, ou aquisição de conhecimento, como um dos seus principais pilares (RUSSELL; NORVIG, 2003). Contudo, pesquisadores da área de Neurociência parecem concordar que existem diversos aspectos, interligados ou não, relacionados à tarefa de interpretação de uma cena ou objeto em uma cena.

Alguns desses aspectos, possíveis de mencionar hoje, são: mecanismo de consciência, mecanismo de atenção, visão *top-down* e *bottom-up* e competição/colaboração dos mecanismos que constituem as diversas características de percepção visual, tais como: cor, forma, textura, relacionamento espacial, co-ocorrência entre objetos, entre muitos outros. Esta seção apresenta a influência da área da Neurociência nos modelos de Visão Computacional propostos na literatura. Dentre esses quatro aspectos encontrados na literatura de neurociência, dois deles, *top-down/bottom-up* e competição/colaboração, estão entre os mais abordados na literatura de visão computacional. Por esse motivo, esses dois serão abordados com maior abrangência nesta seção.

O primeiro aspecto discutido aqui que a Neurociência relaciona ao processo de interpretação de uma cena é a consciência. A consciência é um conceito estudado em diversas áreas, tais como: Filosofia, Neurociência, Psicologia, Física, Inteligência Artificial, entre outras. Para cada área citada, o mecanismo de consciência é definido a partir de diferentes pontos de vista, levando a várias definições para cada uma delas.

Embora não haja na literatura, até onde sabemos, uma definição única e formal sobre a

consciência, pode-se dizer que ela está relacionada com alguns conceitos secundários, tais como: mecanismo de atenção, processo de decisão, percepção e cognição (CALVIN; OJEMANN, 1994; STARZYK; PRASAD, 2011; COHEN; ALVAREZ; NAKAYAMA, 2011).

Alguns autores fundamentam o conceito de consciência a partir do conceito de experiência e objetivo (BAARS; GAGE, 2010). Uma experiência pode ser vista como uma memória entre causa e efeito, ou a habilidade em fazer alguma coisa, entre outras generalidades relacionadas com o conhecimento e cognição (BAARS; GAGE, 2010). Por outro lado, um objetivo pode ser entendido como um alvo ou lugar que deve ser alcançado por um indivíduo. Assim, objetivo é uma ferramenta para fornecer a um indivíduo uma orientação futura. Da mesma forma, em (CHELLA; MANZOTTI, 2007) argumenta-se que a consciência é um mecanismo capaz de relacionar experiências passadas a fim de criar novas experiências e definir novos objetivos.

Podemos então utilizar a definição acima para argumentar que um robô em um ambiente dinâmico, como uma sala, não é consciente, uma vez que ele implementa, no máximo, um algoritmo estruturado capaz de modificar seu estado, mantendo seu objetivo inicial de explorar o ambiente. Uma vez que o objetivo do robô se mantém o mesmo durante toda sua atividade, não se pode considerar que esse robô é consciente, segundo as definições de (CHELLA; MANZOTTI, 2007; BAARS; GAGE, 2010).

Na área da Inteligência Artificial, o opção pelo estudo da consciência é motivo de controvérsias entre os pesquisadores, uma vez que há pelo menos duas opiniões conhecidas a respeito (ARKIN, 1998). A primeira opinião, muito difundida entre os roboticistas, é que esse assunto deve ser tratado apenas como um conhecimento filosófico, uma vez que não há consequências práticas diretas. Assim, eles preferem discutir questões mais práticas como visão computacional de baixo nível, representação de conhecimento, resolução de problemas, meta-heurísticas, entre outros problemas.

Contudo, outro grupo de cientistas é adepto de que a consciência é mais do que uma questão filosófica, podendo trazer informações sobre um modelo computacional hierárquico que pode ser utilizado pela área de visão computacional.

Recentemente, foi cunhado um novo termo chamado “Consciência Artificial” (CHELLA; MANZOTTI, 2007). Em (STARZYK; PRASAD, 2011), os autores propõem um modelo computacional teórico com características relacionadas à consciência. Neste mesmo trabalho, os autores sugerem que um modelo de Consciência Artificial deve ter uma percepção sensorial e ser capaz de prever tomadas de decisões em ambientes através de experiências obtidas anteriormente. Esta ideia está de acordo com as definições de (BAARS; GAGE, 2010; CHELLA; MANZOTTI, 2007).

A conclusão que pode-se tomar com base na literatura da Neurociência até o momento, com relação ao mecanismo de consciência, é que trata-se de um processo que envolve o binômio experiência/objetivo. No entanto, esse processo ainda não é completamente esclarecido, uma vez que muitos aspectos relacionados ainda estão no campo especulativo.

Por exemplo, é razoável pensar que consciência envolve tanto experiências quanto objetivos mutáveis, além disso pode abranger múltiplos objetivos que se inter-relacionam em diferentes contexto, dependendo da experiência acumulada.

Por outro lado, na área de Visão Computacional, são muito populares os mecanismos de aprendizagem de máquina (experiência) para classificação de objetos (específicos). Alguém poderia dizer que trata-se de um tipo de consciência, mesmo sendo em um menor grau. Embora isso pareça uma verdade, contudo considerando os amplos aspectos relacionados ao processo de consciência, abordados ainda especulativamente na área de Neurociência, pode-se dizer que os modelos até agora implementados na área de Visão Computacional tratam-se de modelos em estágios ainda embrionários em se tratando do mecanismo de consciência.

Isso significa que, para ampliar a capacidade dos modelos de Visão Computacional no sentido de implementar maior grau de consciência, é necessário incluir no modelo muitos aspectos que na área da Neurociência ainda estão em debate. No entanto, ainda é possível incluir alguns desses aspectos, tais como: inclusão de múltiplos objetivos através da inclusão de detecção de múltiplos objetos na cena; realimentação de experiência, através da inclusão de reforços ou enfraquecimentos de padrões e talvez a inclusão de aprendizagem infinita.

Esses mecanismos são possíveis de serem implementados. No entanto, não há atualmente

na literatura de Visão Computacional um modelo que inclua todos esse aspectos simultaneamente de modo a ampliar a implementação de um mecanismo de experiência em relação ao mecanismo tradicional de aprendizagem de máquina para o simples reconhecimento de padrões específicos.

O segundo aspecto da Neurociência, citado nesta tese, que está relacionado com o processo de interpretação de uma cena é o mecanismo de atenção. De acordo com (BAARS; GAGE, 2010), o mecanismo de atenção é responsável por definir o objetivo da consciência. A atenção é a habilidade mental em selecionar estímulos, memórias ou pensamentos que são relevantes entre muitos outros que são irrelevantes (CORBETTA et al., 1990).

De uma maneira mais prática, considere um ser humano em um local totalmente escuro sem qualquer noção de onde se encontra. Suponha, inicialmente, que o objetivo principal desse indivíduo seja responder a pergunta: “onde estou?”. Diversos componentes sensoriais enviarão sinais que o ajudarão a entender melhor o cenário. Primeiramente, a audição poderá informar se está perto de uma estrada, cachoeira ou cidade. O sistema nervoso periférico da superfície cutânea enviará informações sobre a temperatura do local. O seu conhecimento prévio sobre o mundo o ajudará a utilizar estas informações para compor uma resposta final. Contudo, muitos outros sensores, que podem não ser tão relevantes como os citados anteriormente (paladar, dores superficiais, pensamentos sobre outros assuntos), podem desviar o indivíduo de sua resposta correta. Além disso, a quantidade de informação, caso todos esses sinais fossem considerados, seria tão grande que a pergunta poderia não ser respondida em tempo hábil.

Outro ponto importante é que, mesmo em um único sensor, várias informações competem por atenção entre si. Por exemplo, quando um pessoa conversa com outra em uma festa, muitas outras conversas acabam sendo escutadas por ambas. Contudo, os sinais capturados pelo sistema auditivo parecem ser “filtrados”, de tal modo que somente as informações relevantes para o diálogo são focadas pelo indivíduo.

Dessa maneira, sem um mecanismo de atenção para focar em cada um dos sensores, poderia não ser possível de obter respostas e ações em tempo hábil e de maneira correta. Assim, como também argumentado em (STARZYK; PRASAD, 2011), não pode haver consciência sem o



Figura 2.10: Exemplos de mapas de saliência. As imagens localizadas na parte superior são originais. As imagens localizadas na parte inferior são os mapas de saliência. (CHENG et al., 2011)

mecanismo de atenção.

Especificamente para o sistema visual humano, estudos na área da psicofísica indicam que há dois estágios para o mecanismo de atenção: pré-atentivo e atentivo. O estágio pré-atentivo é responsável por extrair informações de baixo-nível sobre a cena em observação. Algumas dessas características são: cor, forma, textura, linhas, simetria, excentricidade, posicionamento, entre outras.

Na área de visão computacional, alguns pesquisadores implementam o mecanismo pré-atentivo como parte integrante de seus modelos. Uma das maneiras tradicionais de representar esse mecanismo é através do uso dos mapas de saliência. De uma maneira geral, os mapas de saliência são gerados a partir da utilização de uma ou mais medidas que relacionam uma região da imagem original com suas regiões vizinhas.

Tradicionalmente, os mapas de saliência são representados por imagens em tons de cinza de tal forma que, quanto maior a intensidade de um pixel ou conjunto de *pixels* em um determinada região, mais atenção deve ser dada a essa região. Exemplos de mapas de saliências são mostrados na Figura 2.10. O valor da intensidade de cada pixel no mapa de saliência será proporcional ao seu destaque em relação a sua vizinhança; isto é, quanto mais uma região se diferencia de sua vizinhança, maior destaque ela terá.

O outro estágio envolvido no mecanismo de atenção é o atentivo. Este estágio é responsável efetivamente pelo processo de decisão para a escolha da região mais importante para receber o

foco do sistema. Esse estágio relaciona informações advindas dos próprios conhecimentos e experiências aprendidas pelo indivíduo como também os sinais recebidos pelo estágio pré-atentivo. Essas relações de alto-nível são referenciadas na literatura como modelo *top-down*, pois partem de um processamento de alto nível, como conhecimento, e descem para níveis mais baixo, como as características processadas no estágio pré-atentivo.

Na área de Visão Computacional, o trabalho de Torralba (OLIVA et al., 2003) propõe um modelo de atenção visual para reconhecimento de objetos. Esse modelo é baseado nas probabilidades de ocorrência de características locais (orientação, cores, texturas e contraste) em um determinado ponto (x, y) das imagens da base. Inicialmente, são computadas as probabilidades de cada características ocorrerem em cada ponto sobre todas as imagens da base. Em seguida, o sistema estima qual característica ocorre com maior frequência em cada região. O mapa de saliência é criado em função dessas probabilidades. Os autores argumentam que as regiões relacionadas com as altas probabilidades de ocorrência não são interessantes para o mapa de saliência, uma vez que elas aparecem com muita frequência nas imagens da base e, portanto, são comuns e não oferecem nenhum grau de discriminação. Assim, os pontos com menores probabilidades são selecionados para fazerem parte do mapa de saliência, que pode vir a ser utilizado diretamente em um modelo de Visão Computacional capaz de implementar o aspecto relacionado ao mecanismo de atenção.

Neste mesmo trabalho, (OLIVA et al., 2003), também foi proposta a adição de um modelo *top-down* probabilístico baseado em informações contextuais. Para essa implementação, foi necessária uma base supervisionada contendo a localização dos objetos de interesse. De posse dessas localizações, foi aplicado o PCA sobre o vetor de características locais nas regiões onde haviam objetos de interesse. A redução de dimensionalidade sobre o vetor de características foi necessária uma vez que esses dados fariam parte de um modelo probabilístico. Essa transformação nos dados torna a aprendizagem mais eficiente e mantém apenas informações relevantes no modelo. Assim sendo, foi criado um modelo probabilístico associando uma classe de objetos as suas possíveis características locais. Finalmente, um mapa de saliência é gerado a partir dos dois modelos probabilísticos utilizando uma regra de associação. Os mapas de saliência obtidos a partir dos experimentos gerados nesse trabalho foram comparados com os padrões de escaneamento dos olhos

de observadores humanos. Os resultados mostraram que as informações contextuais consideradas no trabalho foram capazes de aproximar os mapas de saliência aos padrões de escaneamento dos humanos.

Considerando os conceitos vistos até aqui, podemos definir a atenção como a habilidade de fazer a escolha de um estímulo e/ou processo mental que está mais associado às intenções (ou objetivos) de um sistema.

O terceiro aspecto abordado nessa tese que está relacionado com reconhecimento visual de objetos são modelos *bottom-up* e *top-down*. Alguns cientistas da área da Neurociência (BAARS; GAGE, 2010), e até mesmo da computação (LAAR; HESKES; GIELEN, 1997), acreditam que esses modelos fazem parte do mecanismo de atenção.

Em (LAAR; HESKES; GIELEN, 1997), os autores discutem sobre diversas sub-tarefas que o mecanismo de atenção pode executar. Dentre elas, duas estão relacionadas com os modelos hierárquicos *bottom-up* e *top-down*: atenção exógena e atenção endógena.

A atenção exógena, também chamada de atenção *bottom-up*, é exercida sempre que uma interferência externa chama atenção de um indivíduo (LAAR; HESKES; GIELEN, 1997). Neste caso, imagine uma foto de um cenário como um céu claro. Nele, é comum identificarmos a cor azul de nossa atmosfera e algumas poucas nuvens brancas pairando. Contudo, sempre que um objeto se diferencia muito desse padrão de cores, chamará nossa atenção. Assim, se uma foto semelhante for apresentada, mas com um pássaro cruzando o céu, será rapidamente focada pelo nosso sistema visual. Nesse exemplo, apenas a característica de cor foi ressaltada. Contudo, outras características como forma, textura, tamanho, simetria, movimento, cantos, entre outras, podem ser capturadas pelo sistema de atenção *bottom-up*.

Outra sub-tarefa do mecanismo de atenção relacionada ao modelo hierárquico é a atenção endógena. Também chamada de atenção *top-down*, a atenção endógena é exercida sempre que se utiliza as próprias experiências e conhecimentos de um indivíduo. Assim, a atenção endógena é guiada apenas pelas intenções, estímulos e níveis cognitivos mais altos do observador, sem qualquer interferência direta externa (LAAR; HESKES; GIELEN, 1997).

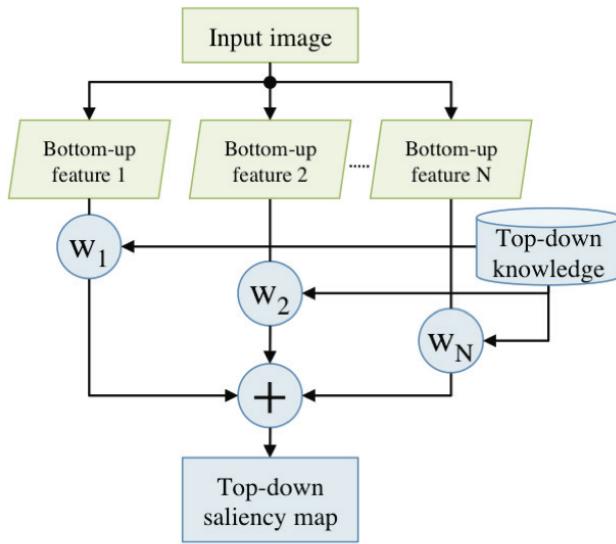


Figura 2.11: Modelo de atenção proposto por (WOLFE; CAVE; FRANZEL, 1989). Note a presença do modelo *bottom-up*, através da extração de características; e a presença do modelo *top-down*, através das informações *a priori* da base de conhecimento.

Para exemplificar o uso da atenção *top-down*, pode-se utilizar o exemplo de uma pessoa em uma festa conversando com um grupo de pessoas em particular. Assim que o assunto em discussão não faz mais parte do interesse dessa pessoa, sua atenção se voltará para algo que ache mais interessante na festa. Note que a mudança de atenção foi exercida a partir da própria intenção do indivíduo.

Na área de reconhecimento de objetos, alguns modelos computacionais baseiam-se em estruturas *top-down*. Como exemplo de algumas dessas modelagens, podemos citar os modelos piramidais (ANDERSON; ESSEN, 1987), grids regulares (YANG et al., 2009), entre outros que utilizam wavelets (NABOUT, 2013) e fourier (SARFRAZ; HASSAN; IQBAL, 2009).

O trabalho de (WOLFE; CAVE; FRANZEL, 1989) sugere um modelo *bottom-up* e *top-down* para representação da atenção visual. O processo *bottom-up* é feito a partir da representação da imagem por características de baixo nível. A representação do processo *top-down* é feita a partir de um conhecimento prévio sobre a importância de cada característica. Assim, há uma ponderação de cada sinal (característica) de acordo com as próprias experiências (base de conhecimento) do sistema (veja Figura 2.11). Finalmente, o mapa de saliência é feito a partir da somatória de todas as características observadas ponderadas pela base de conhecimento.

O último aspecto citado nesta tese relacionado ao processo de detecção de objetos em uma cena é a competição e colaboração entre mecanismos sensoriais. Segundo alguns cientistas da área da Neurociência, como (BAARS; GAGE, 2010), o cérebro humano trabalha com muitos sinais ao mesmo tempo. Essa característica abre uma questão importante sobre seu funcionamento: como todas essas informações em paralelo são gerenciadas e utilizadas pelo cérebro?

Com o objetivo de explicar melhor essa questão, considere o exemplo de um animal em busca de uma presa. Em um primeiro momento, sua atenção poderá estar focada em seu sistema auditivo, capaz de informar sobre a presença de uma presa ao seu redor. Tão logo sua visão volte-se ao local indicado pelo sistema auditivo, seu sistema visual será capaz de informá-lo mais precisamente o local possível da presa. Nesse momento, há duas possíveis interpretações para o predador: o sistema visual informa que o alvo é de fato uma presa; ou o sistema visual informa que o alvo não é uma presa e não é de seu interesse. Neste caso, o sinal enviado pelo sistema auditivo entrará em competição com o sinal advindo do sistema visual. A interpretação dos dois sinais fará com que o predador julgue a presença ou não dessa presa.

A questão da competição é apenas uma característica neural para tratar as informações vindas não somente diferentes sistemas sensoriais, mas também de um mesmo sistema sensorial. Além dessa característica, o cérebro parece ter outro mecanismo capaz de associar informações através de um aprendizado. Considerando o mesmo exemplo dado anteriormente, se o predador encontrar de fato a presa através do sistema auditivo e visual, o sistema olfativo poderá se beneficiar, uma vez que seu cérebro irá associar o cheiro sentido, no momento da caça, com os sinais auditivos e visuais. Em um segundo momento, o animal poderá utilizar o sistema olfativo para detectar outra presa semelhante, melhorando sua capacidade de identificação de presas.

Em (MORAN; DESIMONE, 1985; DESIMONE, 1998; REYNOLDS; CHELAZZI; DESIMONE, 1999; BOYNTON, 2005), os autores estudam a influência de dois estímulos visuais competitivos em cérebros de primatas. A Figura 2.12 ilustra a presença desses dois estímulos, indicados pelos números 1 e 2. Nessa figura, o ponto central representa a região onde um primata foi treinado para manter seu foco visual durante todos os experimentos.

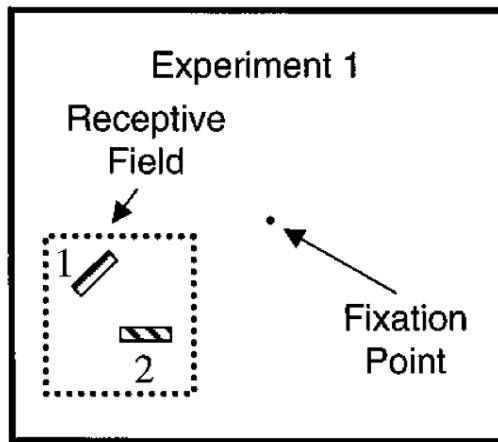


Figura 2.12: Ilustração dos estímulos presentes nos experimentos de (REYNOLDS; CHELAZZI; DESIMONE, 1999).

Inicialmente, foi elaborado um conjunto de 16 estímulos, todos formados com retângulos de diferentes orientações e cores. No total, 4 orientações ($0^\circ, 45^\circ, 90^\circ, 135^\circ$) e 4 cores (vermelho, azul, verde e amarelo) foram escolhidas.

O experimento consistiu em mostrar um dos 16 tipos de retângulos na região 1 ou 2 isoladamente para o primata e medir a quantidade de disparos neurais na região V4 (responsável principalmente pela percepção de cores) de seu cérebro. Quando o primata foi submetido a um teste com dois estímulos, apresentados simultaneamente nas regiões 1 e 2, a quantidade de atividade cerebral na região V4 ficou entre as medições obtidas quando cada estímulo foi apresentado isoladamente.

Mais tarde, em (REYNOLDS; CHELAZZI; DESIMONE, 1999), os autores constataram que a quantidade de atividade cerebral presente quando os dois estímulos são apresentados ao mesmo tempo é uma média ponderada sobre quantidade de atividade medida quando os dois estímulos foram apresentados isoladamente. Esse resultado sugere que os dois estímulos são considerados pelo cérebro ao mesmo tempo e, além disso, que pode haver uma “competição” por atenção sobre cada estímulo.

Na área de Visão Computacional, em (RODRIGUES; GIRALDI; ARAUJO, 2005) é proposto um modelo bayesiano para recuperação de imagens baseado no conteúdo (CBIR). Para este sistema, duas características na forma de vetores são analisadas: cor (*KC*) e forma (*KF*). Os autores

propõem um modelo matemático que baseado em probabilidade condicional. O modelo estima a probabilidade condicional de uma imagem I_j ser retornada dada uma imagem de busca Q . Mais formalmente, o sistema pode ser representado através da equação bayesiana $P(I_j|Q) = P(I_j|KC, KF)$.

Contudo, $P(I_j|KC, KF)$ não é obtido da maneira convencional. Neste caso, os autores propõem a Equação (2.9) como uma maneira alternativa baseada na lógica de *ou-exclusivo* na forma $1 - (1 - t_1) \times (1 - t_2) \times \dots \times (1 - t_n)$, onde t é um termo probabilístico.

$$P(I_j|KC, KF) = 1 - (1 - P(I_j|KC))^u \times (1 - P(I_j|KF))^v \quad (2.9)$$

A Equação (2.9) apresenta diversos pontos interessantes. O primeiro deles é que ela pode ser facilmente estendida para mais características, simplesmente adicionando novos termos no padrão $(1 - P(I_j|Kx))$, onde Kx é um novo vetor de características. O segundo ponto interessante é que há parâmetros potencializadores para cada termo; isto é, as variáveis u e v podem modelar o grau de importância de cada característica. No trabalho de (RODRIGUES; GIRALDI; ARAUJO, 2005), os valores dos parâmetros u e v , nomeados como ‘parâmetros de replicação de evidências semânticas’, foram calculados experimentalmente.

O terceiro ponto, mais relevante para esta seção, é que podemos interpretar esta equação como um modelo competitivo e colaborativo. No caso da característica de competição, podemos interpretar que todas as imagens I_j do banco de dados estão competindo entre si para serem escolhidas pelo sistema.

Para explicar o comportamento colaborativo, pode-se dividir o modelo em 3 casos principais: quando a probabilidade de um dos termos é igual a 0; quando a probabilidade de um dos termos é igual a 1, e quando a probabilidade dos dois (ou mais) termos são diferentes de 0 e 1.

Quando o valor da probabilidade de um termo é igual a 0, o seu termo $(1 - P(x))$ será 1. Isso significa que outro termo será responsável por alterar o valor final da equação. Assim, se um termo “não tem certeza” sobre sua característica, ele deixará a cargo de outro(s) termo(s) decidir(em), havendo a colaboração desse termo.

Quando o valor da probabilidade de um termo é igual a 1, o seu termo $(1 - P(x))$ será igual

a 0. Isso significa que, o valor da equação será 1. Note que basta um termo ser igual a 0 que a equação ficará igual a 1. Isso significa que, se um termo “tem certeza” sobre uma característica, sua escolha é “cedida” pelos outros termos, havendo a colaboração dos outros termos.

Finalmente, quando os valores das probabilidades de todos os termos são diferentes de 0 e 1, todos contribuirão para o resultado final. Isso significa que “o conhecimento” de cada termo será considerado na resposta final, havendo colaboração entre todos os termos.

Outros trabalhos na área da Inteligência Artificial também exploram esses e outros conceitos da Neurociência para a criação de modelos computacionais.

A Tabela 2.1 apresenta alguns modelos computacionais citados nessa tese que foram utilizados em trabalhos com o objetivo de refletir alguns comportamentos baseados na Neurociência. Nesta tabela, são apresentadas algumas vantagens e desvantagens da utilização de cada modelo.

Modelo Computacional	Utilização do modelo em conceitos da Neurociência	Vantagens (V) e Desvantagens (D)	Referências
Modelos Supervisionados			
Redes Neurais	<ul style="list-style-type: none"> • Cognição Humana • Reconhecimento de Objetos 	<p>V: Habilidade de classificar dados não lineares.</p> <p>D: O modelo aprendido se torna uma “caixa-preta” (não é possível de depurar).</p> <p>D: Só é possível incluir conhecimento na rede a partir de exemplos de treinamento.</p>	(YANG; SHU; SHAH, 2013) (ROWLEY; BALUJA; KANADE, 1998) (FUKUSHIMA, 1980) (BEHNKE, 2003)
Never Ending Learning (Aprendizagem Infinita)			
Máquinas de Vetores de Suporte (SVM)	<ul style="list-style-type: none"> • Cognição Humana • Adaptação a ambientes dinâmicos 	<p>V: Capacidade de aprender em tempo de classificação.</p> <p>V: Capaz de aprender novos casos em ambientes dinâmicos.</p>	(CARLSON et al., 2010)

Redes Bayesianas	<ul style="list-style-type: none"> • Cognição Humana • Reconhecimento de Objetos • Mecanismo pré-atentivo 	<p>V: Capacidade de modelar incerteza de maneira estatística.</p> <p>D: A falta de dados estatísticos pode demandar por técnicas alternativas para estimar as probabilidades <i>a posteriori</i>.</p>	(NEAPOLITAN, 2003)
Redes Complexas	<ul style="list-style-type: none"> • Cognição Humana • Relacionamento entre entidades • Memória • Adaptação a ambientes dinâmicos 	<p>V: As propriedades das redes complexas podem prever comportamentos de sistemas dinâmicos.</p> <p>V: Flexibilidade para descrever diversos sistemas naturais.</p> <p>D: Algoritmos estatísticos são de alta complexidade computacional e técnicas heurísticas devem ser propostas para trabalhar com grandes massas de dados.</p>	(BACKES; CASANOVA; BRUNO, 2013b)

Modelos Não-Supervisionados

Decomposição de Valores Singulares (SVD)	<ul style="list-style-type: none"> • Aprendizagem autônoma • Capacidade de generalização • Identificação das características mais importantes de um sinal 	<p>V: Redução de dimensionalidade considerando as principais tendências de um sinal.</p> <p>D: Para uma matriz $m \times n$, sua complexidade é $O(\min(n^2m, nm^2))$.</p> <p>D: As matrizes U e V são densas.</p>	(BO; REN; FOX, 2011)
			(BO; REN; FOX, 2012)

PCA	<ul style="list-style-type: none"> ● Aprendizagem autônoma ● Capacidade de generalização ● Identificação das características mais importantes de um sinal 	<p>V: Redução de dimensionalidade considerando as principais tendências de um sinal.</p> <p>D: As direções que maximizam a variância dos dados podem não ser as direções que maximizam as informações.</p> <p>Para maximizar informações utiliza-se LDA (Linear Discriminant Analysis), porém os dados devem ser supervisionados.</p>	(MESNIL et al., 2013)
-----	--	---	-----------------------

Modelos de Aprendizagem por Reforço

Q-Learning	<ul style="list-style-type: none"> ● Mecanismo de recompensa ● Mecanismo de Decisão 	<p>V: Modelagem através de MDP (<i>Markov Decision Process</i>).</p> <p>D: Para modelos com muitos estados e ações, há excessivo uso de memória principal.</p>	(DRAPER; BINS; BAEK, 1999) (PALETTA; PINZ, 2000) (PIÑOL et al., 2012)
------------	---	--	---

Tabela 2.1: Modelos computacionais e a influência da Neurociênci a computação

Esta seção apresentou algumas características sobre o processo de reconhecimento de objetos advindas da Neurociência que podem ser utilizadas em modelos computacionais. Além disso, foram apresentadas algumas influências da Neurociência em modelos computacionais utilizados em trabalhos científicos. Na Seção 3, alguns aspectos da Neurociência serão considerados para criação de um modelo computacional para reconhecimento de objetos.

2.4 Redes Complexas

As Redes Complexas são utilizadas para descrever os mais diversos tipos de sistemas (NEWMAN; BARABASI; WATTS, 2006). Exemplos de redes que só recentemente foram possíveis de modelar e estudar são: redes sociais (tais como *orkut*, *facebook*, comunidades sociais em geral), redes biológicas (cadeias de *DNA*, Genoma), redes epidêmicas (BOSE et al., 2013), redes de predador e presa (DUNNE; WILLIAMS; MARTINEZ, 2004), redes econômicas (GLATTFELDER, 2010), entre outras. Essas redes, até pouco tempo, eram desconhecidas ou difíceis de representar matematicamente, ou por falta de um modelo matemático adequado, ou devido à ausência de hardware com capacidade compatível para sua implementação.

A principal característica das Redes Complexas é que as ligações entre seus elementos baseiam-se em regras² previamente estabelecidas (Figura 2.13) (NEWMAN; BARABASI; WATTS, 2006). Contudo, seu crescimento acontece de uma forma natural e, algumas vezes, de forma imprevisível. Ao definir-se a regra das ligações, tem-se ao final uma rede composta de muitas interações, tornando possível a extração de características físicas importantes para o entendimento do comportamento do sistema, tais como: tamanho, diâmetro, grau de conexão médio, densidade, entre outras.

A modelagem de uma rede complexa é semelhante a de um grafo (NEWMAN; BARABASI; WATTS, 2006). Há algumas formas de se modelar um grafo (CORMEN et al., 2001). Uma delas é através da utilização de lista de adjacência de arestas, onde normalmente é implementada utilizando uma tabela de três colunas, onde cada linha define uma aresta do grafo. A primeira coluna

²Essas regras são definidas pelo próprio problema proposto. Por exemplo, no caso de rede social, cada nó representa um indivíduo e cada aresta representa amizade entre dois indivíduos.

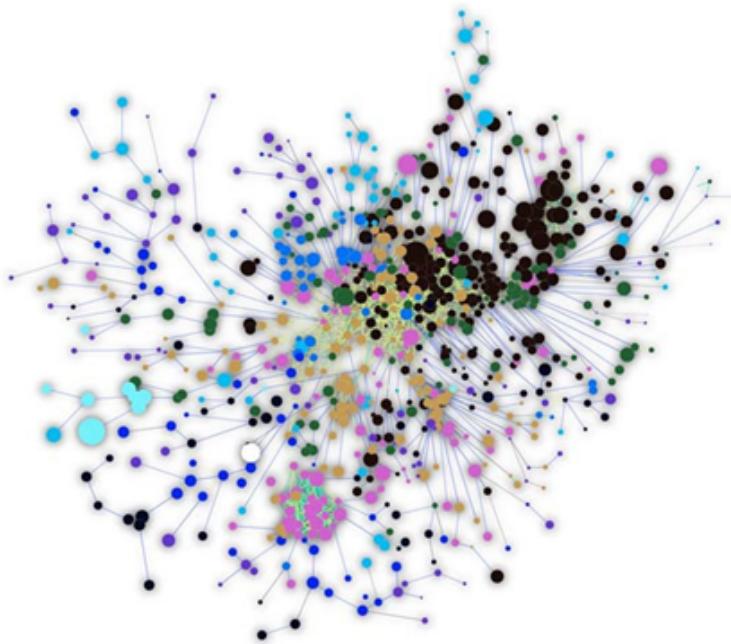


Figura 2.13: Rede complexa representando todos os produtos comercializáveis (HIDALGO et al., 2007). Com esse tipo de modelagem é possível tomar decisões estratégicas locais da economia emergente. O tamanho de cada nó representa a grandeza (em dólares) dos produtos, e as ligações representam a utilização de um produto para produzir outro.

contém o índice do nó de onde a aresta parte, a segunda informa qual é o nó de destino dessa aresta e a última informa o peso da aresta. Uma vez que a tabela da lista de adjacência apresenta um número fixo de colunas e a quantidade de linhas é proporcional a quantidade de arestas, esta modelagem consome espaço da ordem de $O(E)$, onde E é o número de arestas.

Outro tipo de modelagem, cria uma matriz de adjacência de tamanho $N \times N$, onde N é o número de nós. Portanto, o espaço para armazenamento da rede utilizando essa modelagem é da ordem de $\theta(N^2)$. Cada célula dessa matriz representa uma aresta, podendo assumir os valores 0 ou 1, para redes não ponderadas ou outros valores para redes ponderadas. Além disso, se a rede for não-dirigida essa matriz contém somente $N^2/2$ células, dado que somente a parte superior da diagonal principal pode ser utilizada.

A escolha por qual modelagem deve ser feita está diretamente relacionada à densidade do grafo. Para redes esparsas, a modelagem por lista de adjacência parece ser a melhor opção, uma vez que o consumo de memória é $O(E)$. Para redes densas, a matriz de adjacência parece ser a melhor escolha, pois seu consumo é $O(N^2)$.

Na Seção 2.4.1 serão apresentados os três modelos de redes complexas mais estudados na literatura, e na Seção 2.4.2 será visto algumas de suas aplicações.

2.4.1 Modelos de Redes

Nesta seção serão abordados os três tipos de modelos teóricos de Redes Complexas. A primeira delas, Redes Aleatórias, é o modelo mais antigo que possui suas conexões baseadas em uma propriedade conhecida por probabilidade de conexão. O segundo modelo, Redes de Mundo Pequeno, é usado em muitos trabalhos para modelagem de redes sociais e, finalmente, Redes Livres de Escala que baseia-se na dinâmica do crescimento das redes.

2.4.1.1 Redes Aleatórias

A teoria das Redes Aleatórias foi desenvolvida inicialmente por (SOLOMONOFF; RAPOPORT, 1951) e posteriormente estudada por (ERDŐS; RÉNYI, 1959). Esse tipo de rede é construída conectando aleatoriamente seus nós em uma proporção conhecida. Há duas representações matemáticas para este tipo de rede. A primeira delas, $G_{n,p}$, onde n é o número de nós e p é a probabilidade de conexão entre quaisquer dois nós. A segunda representação, $G_{n,m}$, onde n é o número de nós e m é o número de arestas. Essas duas representações matemáticas têm interpretações diferentes, uma vez que em $G_{n,p}$ o número de arestas é alterado proporcionalmente a n e, em $G_{n,m}$, o número de arestas é fixo.

Alguns estudos de Redes Complexas Aleatórias encontram algumas características físicas fundamentais para o estudo de seu crescimento, tais como: existência de *giant component*, fase de transição, *small components*, entre outros (STAUFFER; AHARONY, 1992).

As Redes Aleatórias conseguem modelar algumas redes do mundo real. Porém, alguns fatores podem interferir nos resultados uma vez que, frequentemente, a distribuição de “graus” dos nós em redes reais tende a ser exponencial, *power-law* ou, principalmente, distribuição com picos concentrados em graus pequenos. Uma vez que as redes aleatórias apresentam distribuição de graus binomial, isso impossibilita as redes aleatórias de representar alguns sistemas reais e pode trazer resultados imprecisos caso sejam utilizadas para descrevê-los (ALBERT; JEONG; BARABÁSI,

2000; COHEN et al., 2000).

Como será visto mais adiante, as redes aleatórias servem como parâmetro de medição para comparação entre os modelos. A Equação (2.10) fornece uma medida de quanto uma rede tem características de rede aleatória para as medidas de coeficiente de clusterização e diâmetro. Essa equação resulta em 1 para redes aleatórias e um número muito maior que 1 para redes de mundo pequeno (WALSH, 1999).

$$\mu = \frac{C/C_{rg}}{\ell/\ell_{rg}} \quad (2.10)$$

Na Equação (2.10), C e ℓ são respectivamente o coeficiente de clusterização médio e a distância média entre todos os nós da rede estudada, C_{rg} e ℓ_{rg} são respectivamente o coeficiente de clusterização médio e distância média entre todos os nós estimados para uma rede randômica com o mesmo número de nós e arestas (ver também Seção 2.4.3.3 para coeficiente de clusterização).

2.4.1.2 Redes de Mundo Pequeno

Com o objetivo de modelar redes sociais e estudar o comportamento da proliferação de doenças, internet, redes metabólicas, entre outras, foi criado um modelo de rede complexa que utiliza uma variedade de técnicas da física estatística (WATTS; STROGATZ, 1998). Duas principais características observadas em redes do mundo real levaram à criação das redes de mundo pequeno. A primeira delas é que a média das distâncias entre os nós da rede cresce logaritmicamente de acordo com o número total de nós. Isso significa que a medida em que a rede cresce, suas distâncias crescem mais lentamente. Para fazer a medição dessa característica nas redes, deve-se calcular a média aritmética das distâncias entre todos os nós da rede. A segunda característica é que redes de mundo pequeno possuem alto Coeficiente de Clusterização médio (visto adiante na Seção 2.4.3.3). Para isso acontecer, a vizinhança de um nó deve ser altamente conectada entre si.

O cálculo do coeficiente de clusterização de um nó i é feito a partir da relação entre a quantidade de conexões existentes entre os vizinhos de i e a quantidade máxima possível (NEWMAN; BARABASI; WATTS, 2006). Por exemplo, considere a rede apresentada pela Figura 2.14. Pode-se

calcular o coeficiente de clusterização do nó “A” da seguinte forma:

$$CC_A = \frac{(número\ de\ vizinhos\ de\ A\ conectados\ entre\ si)}{|Vizinhança(A)| \times (|Vizinhança(A)| - 1)} = \frac{4}{4 \times 3} \quad (2.11)$$

Onde $Vizinhança(A)$ é o conjunto de nós que se conectam a A (em qualquer direção).

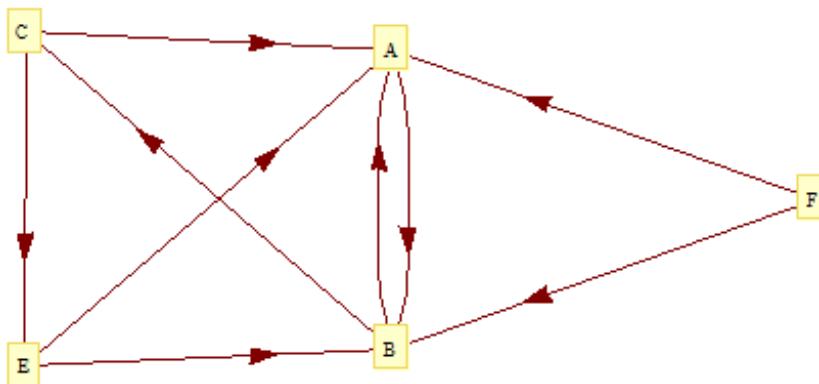


Figura 2.14: Rede com 5 nós

As redes de mundo pequeno assumiram uma importância fundamental para o estudo teórico e prático das redes complexas, uma vez que permitiram a análise do comportamento de alguns sistemas naturais e a possibilidade de observar o comportamento da distribuição de graus, fundamental para o surgimento das Redes Livres de Escala, abordadas na Seção 2.4.1.3 (NEWMAN; BARABASI; WATTS, 2006).

2.4.1.3 Redes Livres de Escala

As redes livres de escala surgiram a partir da observação da distribuição de graus de alguns modelos estudados nos trabalhos de (PRICE, 1965; ALBERT; JEONG; BARABASI, 1999; FALOUTSOS; FALOUTSOS; FALOUTSOS, 1999; BRODER et al., 2000; ALBERT; BARABÁSI, 2002). Algumas contribuições científicas mostraram que a lei de potência é uma característica comum observada sobre a distribuição de graus extraídas a partir de redes do mundo real, tais como: Web (ALBERT; JEONG; BARABASI, 1999), atores de filmes (WATTS; STROGATZ, 1998) e redes de citações científicas (REDNER, 1998).

A lei de potência, descrita pela Equação (2.12), contém um parâmetro de ajuste γ e fornece

uma estimativa sobre a probabilidade de um nó ter grau k .

$$P(k) \sim k^{-\gamma} \quad (2.12)$$

As redes livres de escala também apresentam o efeito de mundo pequeno. Sendo assim, as distâncias entre os nós da rede tendem a crescer logaritmicamente em função da quantidade de nós.

As redes livres de escala são fundamentadas na dinâmica do crescimento de redes naturais; isto é, diferentemente das redes aleatórias, onde o número de nós é fixo e as arestas são inseridas aleatoriamente, as redes livres de escala crescem de acordo com o conceito *ligação preferencial* a cada inserção de um novo nó. A *ligação preferencial* é uma característica que rege a forma com que novas arestas são inseridas na rede. Mais especificamente, quando um novo novo nó é adicionado, a probabilidade desse nó ligar-se com outro nó de grau elevado é proporcionalmente maior do que ligar-se com um nó de baixo grau (ALBERT; BARABÁSI, 2002). A Figura 2.15 ilustra a evolução de uma rede livre de escala quando novos nós (verdes) são inseridos na rede.

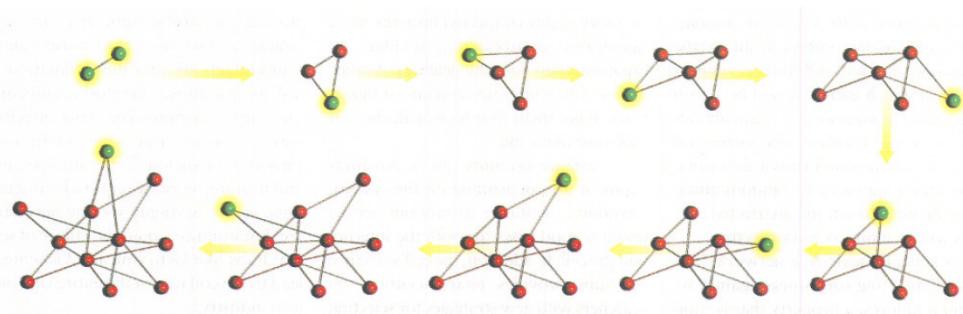


Figura 2.15: Evolução de uma rede livre de escala (BARABASI; BONABEAU, 2003). Nesta figura, um novo nó é representado pela cor verde e nós antigos são representados pela cor vermelha

2.4.2 Exemplos e Aplicações de Redes Complexas

Todo o ferramental desenvolvido para as Redes Complexas permitiu sua aplicação em diversas áreas. Esta seção abordará algumas delas.

2.4.2.1 Redes Sociais

Redes sociais são construídas para estudos de relacionamentos interpessoais. Nestas redes, os nós representam pessoas e as arestas suas relações. Essas relações podem modelar amizades, doenças, casamentos entre famílias, comunidades de negócio, colaboração no trabalho, contatos telefônicos, comunicação por e-mail e até mesmo relações sexuais (LILJEROS et al., 2001).

2.4.2.2 Redes de Informação

As redes de informação são construídas a partir de bases de conhecimento formal. Nestas redes, os nós representam informações e as arestas a relação entre essas informações.

Em (REDNER, 1998), foi estudado um modelo de redes complexas para representar as citações entre artigos acadêmicos. As características dessa rede permitiram entender a dependência entre a distribuição de graus dos artigos, que é descrita por uma lei de potência, e o *rank* de classificação de acordo com o número de citações (PRICE, 1965).

2.4.2.3 Redes Tecnológicas

Redes tecnológicas são redes complexas utilizadas para modelar a distribuição de facilidade ou recursos, tais como: água, malha elétrica, transporte, linhas aéreas, telefonia (fixa), internet (cabeamento e roteadores), entre outros.

Em (GOVINDAN; TANGMUNARUNKIT, 2000), foi proposto um modelo de redes complexas para elaboração de heurísticas capazes de aumentar a fidelidade dos roteadores. O mesmo estudo faz uma análise sobre o mapa físico da internet.

2.4.2.4 Redes Biológicas

Alguns sistemas biológicos tais como vascular, nervoso, circulatório, entre outros, são naturalmente identificados como redes complexas. Um exemplo da aplicação dessas redes é o trabalho de (SPORNS, 2002), no qual foi analisado o cortex cerebral de primatas através do estudo das características físicas da rede.

Outro exemplo da aplicação de redes complexas nessa área é o estudo da dependência entre proteínas. Nesse caso, uma rede complexa pode ser modelada utilizando seus vértices para representar as proteínas e as arestas suas dependências para sua síntese (FARKAS et al., 2002; GUELZIM et al., 2002; SHEN-ORR et al., 2002). O estudo de cadeias alimentares também é outro exemplo onde pode-se aplicar redes complexas para o entendimento de diversos ecossistemas (ALLESINA; PASCUAL, 2008).

2.4.3 Características Físicas de Redes Complexas

Frequentemente, ao modelar um sistema através de redes complexas não é possível classificá-la entre aleatória, mundo pequeno ou livres de escala até que algumas propriedades sejam extraídas (NEWMAN; BARABASI; WATTS, 2006). Algumas dessas propriedades, necessárias para este trabalho, são citadas a seguir:

2.4.3.1 Grau de Entrada e Saída (*In-Out Degree*)

Dentre muitas características importantes de um nó, pode-se encontrar a quantidade de arestas que chegam ou saem dele: grau de entrada ou grau de saída, respectivamente. Por definição, o somatório do grau de entrada com o de saída resulta no grau de conexão k de um nó. O grau de conexão médio de uma rede complexa é denotado por $\langle k \rangle$, que é computado através da Equação (2.13), onde E é o número total de arestas e N é o número total de nós da rede.

$$\langle k \rangle = \frac{E}{N} \quad (2.13)$$

Uma vez que os nós de uma rede complexa podem ter diferentes graus de conexão, utilizando a função de densidade de probabilidade desses graus pode ajudar a entender o tipo específico da rede.

Teorias sólidas desenvolvidas indicam que redes livres de escala seguem a distribuição da lei de potência (Equação (2.12)), onde $P(k)$ é a probabilidade do grau k ocorrer na rede (NEWMAN; BARABASI; WATTS, 2006). Esse comportamento sustenta a idéia que existem muitos nós com baixo grau de conectividade e poucos nós com alto grau de conectividade (chamados *hubs*).

Outra utilidade importante encontrada para o grau de entrada e saída está na checagem de erros na implementação rede. Seja $In(i)$ o grau de entrada de um nó I e $Out(i)$ o grau de saída. A Equação (2.14) mostra o relacionamento entre as duas propriedades físicas.

$$\sum_i In(i) = \sum_i Out(i) \quad (2.14)$$

2.4.3.2 Distribuição de Pesos

Muitas das características apresentadas nesta tese podem ajudar a extrair propriedades estruturais topológicas das redes, uma vez que elas podem não só fornecer informações de grupos e relações entre as entidades, mas também semelhanças entre elas.

No caso da distribuição de pesos, esta é uma propriedade que apresenta a frequência com que cada peso aparece na rede. No trabalho de (BARRAT et al., 2004) os autores propõem um método que utiliza informações topológicas em conjunto com a distribuição de grau para extraír informações de heterogeneidade de duas redes complexas: uma gerada a partir de dados de transporte aéreo e outra de colaboração científica.

Uma maneira interessante de se observar como o comportamento da distribuição dos pesos está associado à topologia da rede é calcular a entropia da distribuição. Uma entropia baixa significa concentração de informação, levando possivelmente a uma rede sem topologia clara, uma vez que nesse caso, isso só é possível com muitas arestas de pesos semelhantes. No caso contrário, quando a tendência da entropia é alta, isso significa uma distribuição de pesos mais heterogênea, significando que existem vários conjuntos, de mesma quantidades de arestas, com pesos iguais dentro dos conjuntos e diferentes entre esses conjuntos. Nesse caso, não é possível inferir a topologia da rede sem a análise de outras propriedades.

Para a extração da distribuição de pesos deve-se antes determinar o número de discretizações que será utilizada. Após esta etapa, inicia-se a varredura de e contagem de todas as arestas sobre cada intervalo discretizado. Ao final, tem-se a quantidade com que cada faixa de valores de arestas está presente na rede, gerando um histograma de valores absolutos.

O algoritmo para extração da distribuição de pesos é da ordem de $O(E)$ (proporcional ao número de arestas que, no pior caso, pode chegar a $O(N^2)$), uma vez que deve-se percorrer todas as arestas da rede.

2.4.3.3 Coeficiente de Clusterização

A média geral do Coeficiente de Clusterização (ACC) é uma importante característica física de redes complexas com implicações em diversas aplicações. Por exemplo, (WATTS; STROGATZ, 1998) definiu uma rede complexa como mundo pequeno se ela apresenta duas propriedades em conjunto. A primeira delas é que a média das distâncias entre todos os vértices da rede (ℓ) deve comparável àquela de redes aleatórias, $\ell/\ell_{rg} \sim 1$. A segunda é que o coeficiente de clusterização médio deve ser muito maior do que o de uma rede aleatória, $ACC/ACC_{rg} \gg 1$. Ambas as propriedades são definidas para uma rede de mesma quantidade de nós e arestas (NEWMAN; BARABASI; WATTS, 2006).

Algumas aplicações importantes como a Internet, World Wide Web, Colaboração Biológica, Co-ocorrência de Palavras, entre outras, apresentam tais características (NEWMAN; BARABASI; WATTS, 2006), enfatizando a necessidade de computar o coeficiente de clusterização da rede.

O cálculo do coeficiente de clusterização expressa o quanto os nós de uma vizinhança de um nó i estão conectados entre si mesmas. Este valor varia entre 0 e 1, onde 0 é uma vizinhança totalmente desconectada e 1 é uma vizinhança totalmente conectada.

Formalmente, considere i como qualquer nó de uma rede complexa, L_i o conjunto de nós que têm uma conexão com i , e $W = \{w_{u,v} | u, v \in L_i\}$ um conjunto de pesos de arestas que conectam cada nó de L_i a outro nó de L_i . A Equação (2.15) mostra o cálculo do coeficiente de clusterização para uma rede dirigida.

$$CC_i = \frac{|W|}{|L_i| \times (|L_i| - 1)} \quad (2.15)$$

Esta equação relaciona o número $|W|$ de arestas existentes na vizinhança de i com o máximo número de arestas possíveis para a quantidade de $|L_i|$ nós. A Equação (2.15) é capaz de calcular o coeficiente de clusterização, porém ela não considera os pesos das arestas, mas apenas a quantidade

de elementos. Em casos de redes ponderadas, essa equação é obviamente limitada e, em alguns casos, não se aplica.

Tendo isso em mente, propomos uma nova equação para o cálculo do CC e adicionamos o peso às arestas, trocando a quantidade $|W|$ na Equação (2.15) das arestas pela soma dos pesos dessas arestas $\sum W_e$:

$$CC_i = \frac{\sum W_e}{|L_i| \times (|L_i| - 1)} \quad (2.16)$$

Então, para uma rede com N nós, o coeficiente de clusterização médio ACC é dado por

$$ACC = \frac{1}{N} \sum_i CC_i \quad (2.17)$$

A Equação (2.16) é idêntica à Equação (2.15) com as seguintes modificações: o numerador $|W|$ na Equação (2.15) é o tamanho do conjunto de pesos $w_i \in \{0, 1\}$, e na Equação (2.16) $0 \leq w_i \leq 1$. Isso permite que consideremos valores ponderados entre 0 e 1. Dessa forma, na Equação (2.15), o coeficiente de clusterização é computado apenas para cada nó j vizinho à i conectado por uma aresta $w_{i,j} = 1.0$. Porém, na Equação (2.16), a conexão entre i e um nó vizinho j é ponderada. A consequência dessa modelagem é que o coeficiente de clusterização de i não é somente computado levando em consideração a conexão entre a vizinhança, mas também o quanto i está conectado à ela.

A Figura 2.16 mostra um exemplo. Para calcular o coeficiente de clusterização do nó $i = 1$, deve-se primeiramente identificar seus vizinhos, nesse caso $L_1 = \{2, 3, 5, 6\}$. O método usado aqui considera todas as arestas conectando quaisquer dois nós do conjunto L_1 , gerando o conjunto de arestas $E = \{2 \rightarrow 3, 3 \rightarrow 5, 6 \rightarrow 2\}$ e o conjunto $W_e = \{0.5, 0.3, 0.4\}$. Então, aplicamos a Equação (2.16), encontrando $CC_1 = 0.1$. Para a Equação (2.15), $CC_1 = 0.25$.

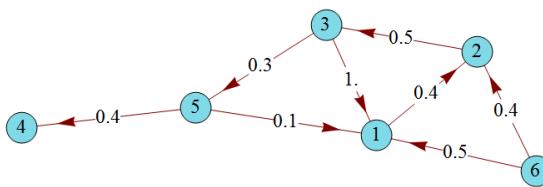


Figura 2.16: Coeficiente de Clusterização

Note que, para o coeficiente de um nó ser igual a 1, é necessário que todas as arestas possíveis da vizinhança tenham $w_{i,j} = 1$. Nesse caso, a Equação (2.16) se reduz à Equação (2.15), portanto, ela é uma generalização (veja Figura 2.17).

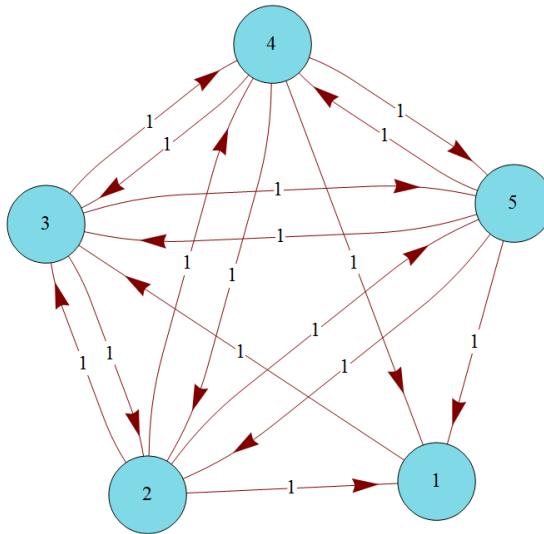


Figura 2.17: Coeficiente de Clusterização Máximo

Note que a Equação (2.16) também pode ser utilizada para redes não dirigidas. Para isso, teremos a metade do total de arestas possíveis entre os vizinhos. Sendo assim, o denominador da equação será $|L_i| \times (|L_i| - 1)/2$ e a Equação (2.16) pode ser re-escrita como:

$$CC_i = \frac{2 \times \sum W_e}{|L_i| \times (|L_i| - 1)} \quad (2.18)$$

2.4.3.4 Densidade de Conexão Média (K_{den})

A Densidade de Conexão Média (K_{den}) é uma medida física que relaciona a quantidade de arestas existentes em uma rede complexa com a quantidade de arestas existentes em uma rede totalmente conectada (completa), para o mesmo número de nós. Portanto, essa medida varia entre 0 e 1, onde 0 significa uma rede esparsa e 1 significa uma rede densa. Dessa forma essa característica está relacionada com a completude de um grafo.

Em (SPORNS, 2002), o autor comenta estudos que inferem a topologia de redes complexas a

partir de dados neurológicos através de medições de densidades locais (em uma determinada parte da rede) e global (toda a rede). Assim, essa medida também pode fornecer informações estruturais das redes.

A Equação (2.20) apresenta o cálculo do K_{den} para redes não dirigidas:

$$K_{den} = 2 \cdot \frac{|E|}{n^2 - n} \quad (2.19)$$

Para redes dirigidas, o K_{den} é calculado segundo a equação:

$$K_{den} = \frac{|E|}{n^2 - n} \quad (2.20)$$

onde $|E|$ é o conjunto de células da matriz de pesos com pesos diferentes de zero.

Conforme as Equações (2.19) e (2.20), quando seu valor está próximo a zero, há poucas arestas na rede e ela é considerada esparsa. Caso contrário, próximo a um, a rede contém muitas arestas e é considerada densa.

No caso da rede proposta por este trabalho, ao extraímos esta característica, estaremos analisando a quantidade de relações em uma mesma característica da imagem. Se o valor obtido for alto, isso sugere que a característica representada na rede apresenta alta conectividade entre diversas instâncias de valores diferentes. Caso contrário, há poucos relacionamentos entre valores distintos em uma determinada característica.

2.4.3.5 Índice de Semelhança de Conexão (ρ)

O Índice de Semelhança de Conexão $\rho_{i,j}$ (para $i \neq j$) de uma Rede Complexa mede o quanto o conjunto de conexões de um nó i é semelhante ao conjunto de conexões de um nó j (HILGETAG et al., 2000; SPORNS, 2002). O cálculo de $\rho_{i,j}$ é feito somando-se a quantidade de nós semelhantes de i e j (que se conectam aos mesmos nós) e dividindo-os pela quantidade total de nós conectados a i e j . Por exemplo, suponha A como conjunto de nós conectados a i , e B o conjunto de nós conectados a j . O índice de semelhança de conexão $\rho_{i,j}$ é dado pela equação

$$\rho_{i,j} = \frac{|A \cap B|}{|A \cup B|} \quad (2.21)$$

Em (SCHAEFFER, 2007), a autora comenta sobre clusterização baseada em similaridade de vértices. Um dos métodos abordados por ela consiste em comparar a vizinhança dos vértices e agrupá-los de acordo com a semelhança dessas vizinhanças. Contudo, essa abordagem gera um algoritmo de ordem $O(N^3)$.

Trazendo esta mesma idéia para a área de Redes Complexas, podemos comparar dois nós através de suas conexões e propor uma clusterização baseada nesta medida. Porém, estima-se que a Rede Complexa gerada neste trabalho apresenta dimensão muito elevada, o que inviabilizaria o tempo computacional envolvido na clusterização.

Uma possível solução para isso seria um método para redução da dimensionalidade da matriz de pesos utilizando descritores estatísticos. Diversos descritores podem ser utilizados, tais como: média, mediana, desvio-padrão, moda, variância, percentis, entre outros. A fim de se obter uma maior precisão sobre a comparação das distribuições, pode-se calcular os descritores em algumas faixas da distribuição original. Por exemplo, considere v_i um vetor com os pesos de conexão de um nó i . Para representar essa distribuição, desmembra-se o vetor em p partes. Para cada uma dessas partes calcula-se todos os descritores. Esse processo é repetido para todos os nós da rede, gerando-se uma matriz de descritores, onde cada linha representa as características da distribuição de vizinhança de um nó da rede complexa e cada coluna um descritor.

Uma vez que o foco desse trabalho foca na clusterização de redes complexas, optamos por deixar esta idéia em aberto para futuros estudos.

2.4.3.6 Grau de Reciprocidade (ρ)

Conexões Recíprocas são pares de arestas que conectam dois nós em ambos os sentidos. Formalmente, uma conexão recíproca existe se $M_{i,j} > 0$ e $M_{j,i} > 0$ para $i \neq j$ (SPORNS, 2002). A divisão da quantidade de arestas recíprocas pela quantidade de arestas da rede é uma medida conhecida como “Grau de Reciprocidade” (ou “Fração de Conexões Recíprocas”) e é simbolizada por ρ .

2.4.3.7 Probabilidade de Ciclos

Ciclos são caminhos que conectam um nó j a ele mesmo com vértices e arestas distintas. Basicamente, esta medida informa a probabilidade de um caminho ser cíclico (SPORNS, 2002). Por ser uma medida que demanda alto custo computacional, não será abordada nesse trabalho, uma vez que trataremos com bases de dados da ordem de Gigabytes.

2.4.3.8 Matriz de Distâncias, Excentricidade, Raio, Diâmetro

A Matriz de Distâncias armazena o tamanho do menor caminho, d_{ij} , entre um nó i e um nó j (HARARY, 1969). Se nenhum caminho existe entre i e j então $d_{i,j} = \infty$.

A Excentricidade de um nó i é a distância finita máxima para todos os outros nós da rede. Dessa forma, pode-se obter a excentricidade de um nó i a partir da linha da matriz de distâncias: $ecc(i) = \max_{j=1}^N \{d_{i,j}\}$ para $d_{i,j} \neq \infty$.

O Raio de uma Rede Complexa é a excentricidade mínima da rede: $\min_{i=1}^N \{ecc(i)\}$. O Diâmetro de uma Rede Complexa é a excentricidade máxima da rede: $\max_{i=1}^N \{ecc(i)\}$.

2.4.3.9 Matriz de Alcance

A Matriz de Alcance informa se existe pelo menos um caminho que conecta um nó i a um nó j . Se o caminho existe, denotado por $r_{i,j}$, ele recebe o valor 1 (caso contrário, recebe 0) (SPORNS, 2002).

Assim como a probabilidade de ciclos, as outras medidas físicas conhecidas mencionadas aqui (matriz de distâncias, excentricidade, raio, diâmetro e matriz de alcance, bem como, coeficiente espectral, são computacionalmente inviáveis para grandes bases de dados, como as estudadas nessa tese. Sendo assim, como já foi explicado, decidimos não implementá-las, optando por um aprofundamento nas discussões das demais medidas.

2.4.3.10 Modularidade

A modularidade baseia-se na comparação de Redes Aleatórias com o modelo observado. Isso significa que, quanto mais distante uma rede for em relação à uma rede aleatória de mesmas características, mais organizada ela será. Essa organização está relacionada ao conceito de *cluster*. Um conjunto de nós é um *cluster* se a quantidade de arestas entre eles é maior do que aquela esperada se a rede fosse totalmente aleatória. A qualidade de um *cluster* C_k para um conjunto de nós $V \in C_k$ pode ser obtida através da Equação (2.22).

$$Q(C_k) = \sum_{i,j \in V} \left[\frac{A_{ij}}{2m} - P_{ij} \right] \delta(c_i, c_j) \quad (2.22)$$

Onde

$$P_{ij} = \frac{k_i^{out} k_j^{in}}{m^2} \quad (2.23)$$

é a probabilidade esperada de conexão entre os nós i e j para uma rede aleatória dirigida de mesmas características (quantidade de nós (N) e densidade de arestas (K_{den})), m é a soma dos pesos da matriz de pesos, A_{ij} é o peso de uma aresta que sai do nó i em direção ao nó j e

$$\delta(c_i, c_j) = \begin{cases} 1 & : \text{ se } i \text{ e } j \text{ estiverem na mesma comunidade.} \\ 0 & : \text{ caso contrário.} \end{cases}$$

Em (NEWMAN; GIRVAN, 2004) a Equação (2.22) foi proposta como medida de qualidade da clusterização e também foi apresentado e discutido várias aplicações onde essa medida foi usada com sucesso.

Devido a sua simplicidade e relativo baixo custo computacional, aliado aos bons resultados obtidos na literatura recentemente, este método foi o escolhido para medir a qualidade das clusterizações feitas nos experimentos desta tese.

3 PROPOSTA

3.1 Base de Dados

Conforme será explicado com maiores detalhes na Seção 3.2.2, este trabalho demanda, inicialmente, por uma base de dados com imagens supervisionadas. Uma vez que este trabalho utiliza informações específicas das imagens para inferir o objeto, é necessária uma base que contenha tanto as regiões que delimitam os objetos em uma cena como também os rótulos dos objetos.

Em (XIAO et al., 2010), os autores propõem a criação de uma base de dados, nomeada “SUN Database”, que contém 899 categorias de objetos em mais de 130.000 imagens supervisionadas por humanos. Além da supervisão das classes dos objetos, a base também contém informações da classificação das cenas de uma forma geral. A Tabela 3.1 mostra os 20 rótulos de objetos e cenas mais referenciados pelas supervisões da base de dados. Trabalhos recentes da área de reconhecimento de objetos e classificação de cenas têm utilizado esta base de dados como suporte para treinamento e validação (MOTTAGHI et al., 2014; AGRAWAL; GIRSHICK; MALIK, 2014). Nessa tese, esta base de dados será utilizada como fonte inicial de aprendizado (veja Seção 3.2.2).

A “SUN Database” está organizada através de uma estrutura hierárquica de pastas. No primeiro nível, existem duas pastas: “Images” e “Annotations”. A pasta “Images” contém subpastas cujos nomes se referem aos rótulos das imagens. Dentro de cada pasta, um conjunto de imagens que se enquadram no rótulo são armazenadas. Todas as imagens estão em formato JPEG.

A pasta “Annotations” contém as mesmas subpastas da pasta “Images”, porém dentro dessas subpastas existem diversos arquivos XML de supervisão das regiões e objetos. Para cada imagem da base um arquivo XML de supervisão é disponibilizado.

Os arquivos XML são padronizados para descreverem um conjunto de regiões (polígonos) com seus respectivos rótulos. A Figura 3.1 mostra um exemplo de um arquivo XML de supervisão e a imagem correspondente com sobreposição das posições informadas no XML. A Figura 3.1 a tag

Categorias de Cena	Nºm. de Imagens	Categorias de Objeto	Nºm. de Imagens
Living room	2385	Wall	20213
Bedroom	2117	Window	16080
Kitchen	1755	Chair	7971
Beach	1223	Floor	7227
Dining room	1187	Sky	6328
Airport terminal	1152	Ceiling lamp	6268
Castle	1126	Person	6202
Church outdoor	1058	Building	6043
House	972	Trees	5785
Bathroom	956	Ceiling	5284
Playground	909	Tree	4956
Conference room	872	Car	4240
Bridge	870	Door	4135
Highway	861	Cabinet	3102
Market outdoor	853	Plant	3095
Golf course	841	Table	2999
Gazebo exterior	818	Painting	2784
Skyscraper	807	Person sitting	2696
Restaurant	800	Curtain	2525
Warehouse indoor	793	Grass	2427

Tabela 3.1: As 20 categorias mais rotuladas para cenas e objetos

```

<annotation>
  <filename>sun_bhlazecjfmiyxbtg.jpg</filename>
  <source>
    <sourceImage>The MIT-CSAIL database of objects and scenes</sourceImage>
    <sourceAnnotation>LabelMe Webtool</sourceAnnotation>
  </source>
  <object>
    <name>trees</name>
    <deleted>0</deleted>
    <verified>0</verified>
    <date>09-Feb-2010 16:41:17</date>
    <id>0</id>
    <polygon>
      <username>anonymous</username>
      <pt>
        <x>0</x>
        <y>51</y>
      </pt>
      ...
      <pt>
        <x>0</x>
        <y>0</y>
      </pt>
    </polygon>
  </object>
  <object>
    ...
  </object>
  <imagesize>
    <nrows>240</nrows>
    <ncols>320</ncols>
  </imagesize>
</annotation>

```



Figura 3.1: Exemplo de arquivo XML descrevendo as regiões da imagem.

“name”, dentro da tag “object”, é o rótulo da região descrita pelos pontos dentro da tag “polygon”. Neste trabalho, usaremos estas informações para compor o conjunto de entrada de treinamento.

3.2 Metodologia

Neste trabalho, é proposto um modelo de reconhecimento de objetos em cenas naturais que considera várias características armazenadas em uma topologia baseada em redes complexas. Esse modelo é inspirado em modelos biológicos recentemente discutidos na literatura de Neurociências. Sua principal contribuição é o aprendizado infinito, que permite a remodelagem topológica das redes a medida que novos padrões de entrada são apresentados.

Também é proposta uma metodologia experimental, que visa investigar tanto a eficiência do modelo, quanto o seu propósito. De maneira geral, esse modelo baseia-se no registro de observações de características de imagens e seus relacionamentos, aqui chamados de co-ocorrências.

Quando se fala em observação de características, o ponto de interesse está no valor obtido após a extração de algum valor feito sobre uma imagem inteira ou região de interesse (*region of interest*, ROI). Contudo, dependendo da característica observada, este valor pode ser unidimensional ou multidimensional. Nesta tese, cada valor (uni ou multi-dimensional) será referenciado como uma “instância de característica”.

A hipótese central desta tese é que um modelo de co-ocorrência entre as instâncias de características é o principal fator que pode melhorar a qualidade da classificação de um sistema de reconhecimento de objetos. Por exemplo, considere as imagens da Figura 3.2. Nesta figura, a imagem à direita contém regiões segmentadas por humanos (XIAO et al., 2010). Nela, é possível identificar 3 grandes objetos segmentados: céu, oceano e praia. A hipótese desta tese é que a informação de frequência com que cada par de objetos ocorre em cenas da mesma classe pode contribuir significativamente com a qualidade do classificador. No caso da Figura 3.2, as co-ocorrências do céu com o oceano, oceano com praia e céu com praia podem ser úteis quando uma nova imagem que contenha oceano for observada no sistema. A criação da base de co-ocorrência, bem como o uso destas informações no reconhecedor, serão detalhados mais adiante neste capítulo.



Figura 3.2: Imagens ilustrando a segmentação feita por humanos. Esquerda: Imagem original. Direita: Imagem supervisionada

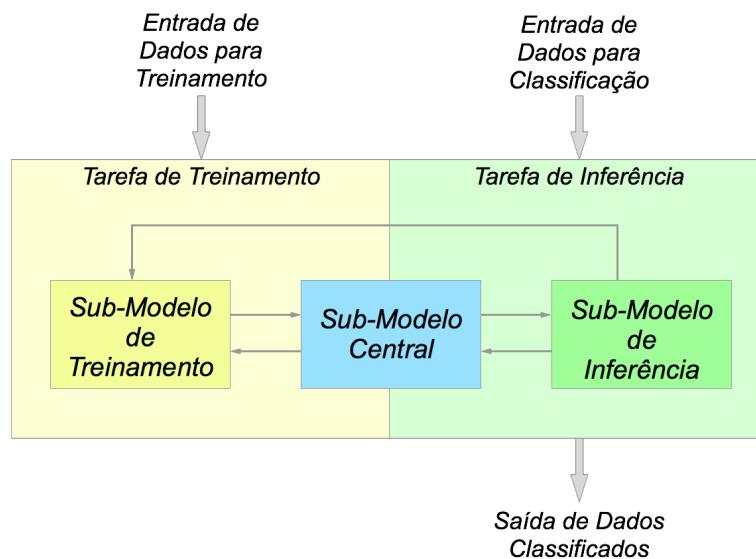


Figura 3.3: Estrutura do Modelo principal proposto

Esta tese sugere um modelo geral composto por sub-modelos e estruturas de dados. Contudo, esses sub-modelos e estruturas de dados não estão rigorosamente atrelados ao Modelo principal, pois podem ser alterados ou adaptados. Desta forma, o Modelo principal proposto pode ser considerado como um Meta-Modelo, que pode ser alterado para outras aplicações.

O Modelo principal proposto opera sobre duas tarefas principais: treinamento e inferência. A tarefa de treinamento é responsável por ajustar os componentes do Modelo principal a partir da apresentação de dados de entrada e saída esperada. Esta tarefa é semelhante aos processos de aprendizagem supervisionada, vistos na Seção 2.2.1. Por outro lado, a tarefa de inferência é responsável por utilizar informações aprendidas para compor uma saída de dados classificados. A Figura 3.3 ilustra a estrutura geral do modelo proposto. Como pode ser observado nesta figura,

para cada uma das duas tarefas do Modelo principal é proposto um sub-modelo. Embora cada tarefa tenha seu próprio sub-modelo, um terceiro sub-modelo, chamado de Sub-Modelo central, é sugerido para ser responsável por centralizar as informações que são manipuladas em ambas as tarefas.

Este capítulo está organizado da seguinte forma. A Seção 3.2.1 descreve o Modelo principal de forma abrangente. A Seção 3.2.2 aborda o Sub-Modelo de Treinamento; na Seção 3.2.3, o Sub-Modelo Central será detalhado. Finalmente, a Seção 3.2.4 apresenta o Sub-Modelo de inferência. A Figura 3.4 apresenta o Modelo principal proposto de forma esquemática. Nas próximas seções, essa figura poderá ser utilizada como apoio.

3.2.1 Descrição Geral do Modelo

No Modelo principal proposto, a Rede Complexa contém informações de frequência com que cada par de instâncias de características aparecem em uma mesma imagem.

Um vez que essa rede deve estar previamente construída, antes que a segmentação e classificação ocorram de fato, é necessário um módulo capaz de enviar informações dessas frequências. Assim, o Sub-Modelo de Treinamento é responsável por fazer a leitura de bases de dados supervisionadas, extrair características dessas imagens e enviá-las para o construtor da rede complexa.

Uma vez treinada, a rede poderá então ser utilizada pelo segmentador e classificador para compor a saída. O módulo responsável por esta tarefa é o Sub-Modelo de Inferência. Neste módulo, uma imagem de entrada é recebida e encaminhada diretamente para um segmentador. Esse segmentador deve ser capaz de gerar um conjunto de diversas segmentações e enviá-lo para o extrator de características. Essas características são enviadas ao maximizador de função, que analisa qual segmentação deve ser escolhida para compor o resultado final. Essa escolha é feita através da maximização do somatório das co-ocorrências entre todos os pares de instâncias de características geradas por cada segmentação. A segmentação que maximizar o somatório será escolhida.

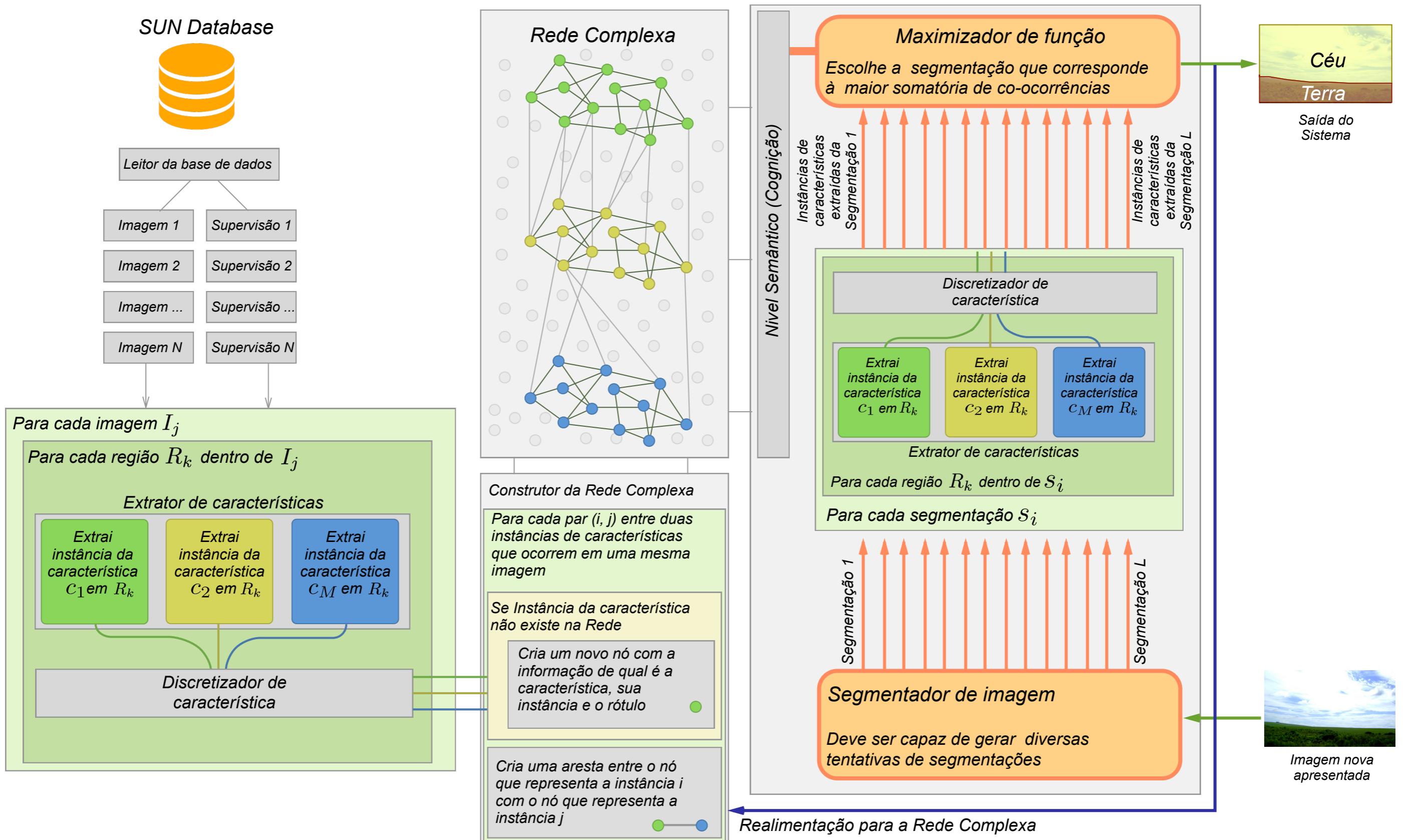


Figura 3.4: Modelo proposto detalhado

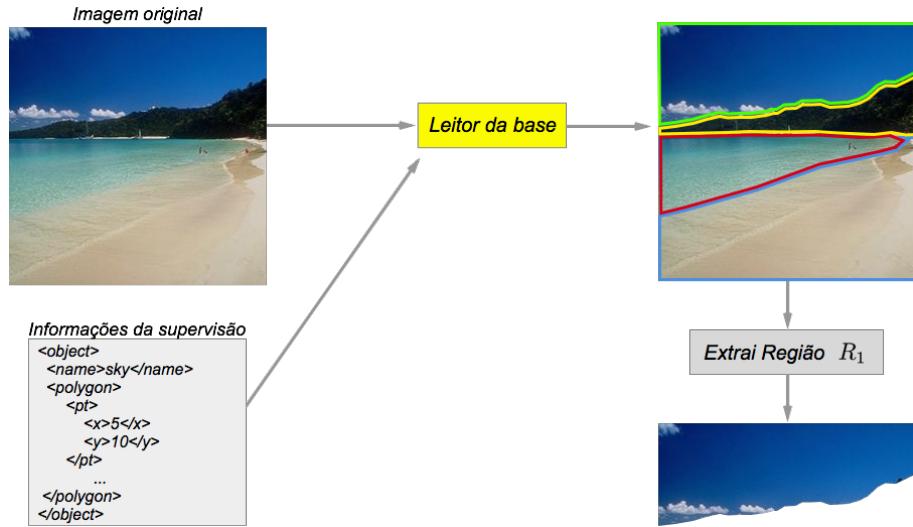


Figura 3.5: Leitor da base efetuando a extração da Região R_1

3.2.2 Sub-Modelo de Treinamento

O Sub-Modelo de Treinamento é responsável por fazer a leitura de bases de treinamento, extrair as características sobre todas as regiões de todas as imagens, discretizar estas informações e enviá-las para o Sub-Modelo Central.

Nesta seção, usaremos a notação I_j para representar a j -ésima imagem do banco de dados e R_k para representar a k -ésima região de uma determinada imagem. Além disso definimos o conjunto de características $\chi = \{c_1, c_2, c_3, \dots, c_m\}$ e uma instância da característica c_i extraída a partir de uma região R_k como (c_i, i_k) .

3.2.2.1 Leitor da Base de Dados

O Leitor da Base de Dados supervisionada recebe como entrada uma base de dados com imagens e arquivos de supervisão, normalmente em XML ou JSON, contendo a descrição de cada região R_k de cada imagem I_j . O conjunto de regiões obtido é então enviado ao módulo Extrator de Características. A Figura 3.5 ilustra essa operação.

Um ponto adicional a ser informado é que o Leitor da Base pode ser especializado para outras bases de dados. Assim, este módulo não está restrito somente à base de dados SUN.

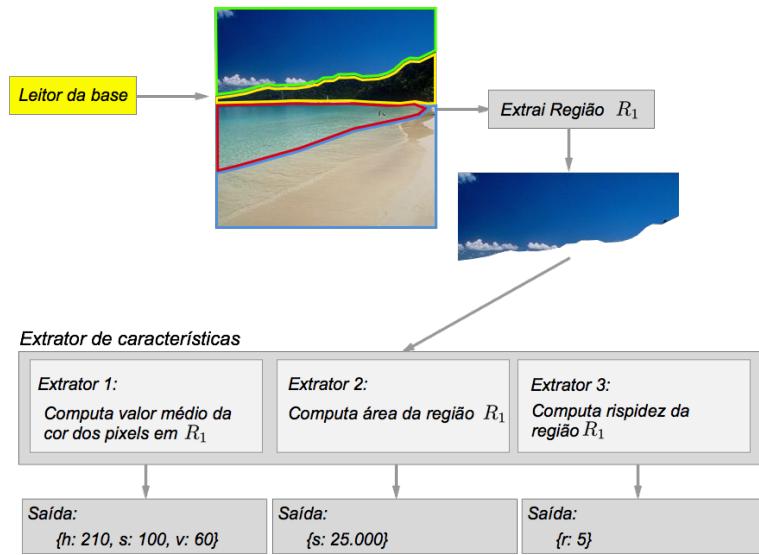


Figura 3.6: Extração de 3 características a partir de uma região

3.2.2.2 Extrator de Características

O Extrator de Características é um módulo da tarefa de treinamento responsável por extraír todas as instâncias de características em todas as regiões de todas as imagens da base e enviá-las ao Sub-Modelo Central.

O Extrator de Características contém um conjunto de sub-extratores, específicos para cada característica $c_i \in \chi$. Na Seção 2.1.1, algumas dessas características são citadas. Na Figura (3.4) esses sub-extratores são representados pelos quadros verde, amarelo e azul. O conjunto de sub-extratores pode ser alterado para que se tenha maior precisão no Sub-Modelo Central. Assim, será definido que o tamanho do conjunto de sub-extratores é M .

O extrator de característica inicia sua operação tendo como entrada o conjunto de N imagens da base de dados já processadas pelo Leitor da Base. Para cada imagem I_j desse conjunto, são extraídas M instâncias de características sobre cada uma das R_k regiões de I_j . Isso significa que, se $M = 3$, $N = 1000$ e cada imagem contém 5 regiões, o processamento da base inteira irá gerar 15.000 instâncias de características. Essas 15.000 instâncias farão parte do conjunto de entrada do Discretizador de Característica.

A Figura 3.6 exemplifica a extração de 3 características sobre uma região de uma imagem supervisionada. Como pode ser observado nessa figura, as características retornam um vetor com

valores escalares.

3.2.2.3 Discretizador de Característica

O discretizador de característica exerce um papel fundamental no modelo proposto. Conforme explicado na seção anterior, um banco de dados poderá gerar milhares de instâncias de características. Contudo, se todas as instâncias geradas tiverem valores distintos entre si, não será possível capturar informações suficientes de co-ocorrência para compor a Rede Complexa, uma vez que cada instância será representada por um nó na rede complexa e a co-ocorrência é proporcional ao peso da aresta entre os nós.

Para exemplificar, considere as duas imagens da Figura 3.7. Ao extrair a característica de cor média sobre as duas figuras, obtemos um conjunto de instâncias. Considere apenas as 2 instâncias extraídas a partir das regiões destacadas em vermelho. Essas 2 instâncias farão parte da entrada de dados do Sub-Modelo Central. Conforme será abordado na Seção 3.2.3, os nós correspondentes a essas 2 instâncias serão buscados e, caso não existam, serão criados. Observe que essas instâncias são muito semelhantes entre si. Entretanto, essa semelhança não justifica a criação de dois nós distintos. Por essa razão, o Discretizador de Característica estabelece o número máximo de instâncias para cada característica discriminando seus valores. Dessa forma, um nó já pertencente à rede que contém uma instância de característica próxima o suficiente será re-aproveitado, gerando assim, mais co-ocorrências. Uma pergunta experimental, que será investigada nessa tese, é a dimensão da discretização que maximiza a separação das regiões ou objetos da cena.

3.2.3 Sub-Modelo Central

Conforme explicado anteriormente, o Sub-Modelo Central é responsável por agregar e gerenciar todas as informações que serão utilizadas em seus sub-modelos vizinhos.

Uma vez que a hipótese central desta tese refere-se às informações de co-ocorrência, é necessário ter um modelo capaz de representá-las. Dessa forma, foi escolhido o modelo de Redes Complexas, uma vez que ele não só é capaz de relacionar entidades, assim como um grafo, como também contém ferramentas capazes de fornecer informações topológicas da rede.

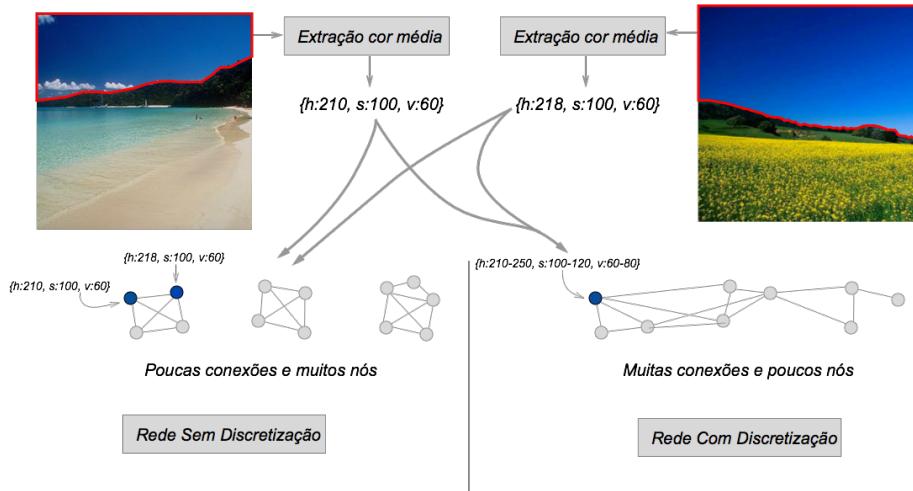


Figura 3.7: Exemplo de duas redes geradas: uma com discretização de instâncias e outra sem discretização

Conforme citado na Seção 3.2, cada instância de característica é relacionada a outra instância de característica sempre que ambas forem encontradas em uma mesma imagem. Essa informação é armazenada através dos nós e arestas da rede complexa; isto é, cada nó da rede é uma instância de característica e cada aresta é a co-ocorrência entre duas instâncias. Mais especificamente, cada nó da Rede Complexa guarda o valor obtido a partir da extração de uma característica em uma determinada região da imagem.

A Seção 2.1.1 apresentou alguns modelos de representação de imagens através de características. Algumas dessas características serão utilizadas para fazer parte da Rede Complexa. Dessa forma, tanto a informação da instância da característica quanto a própria informação de qual é a característica extraída devem ser modeladas na Rede Complexa. Essa informação será adicionada como um novo atributo nos nós da rede. A Figura 3.8 ilustra a representação de duas características extraídas a partir de uma imagem. A primeira vantagem em se utilizar um modelo de Redes Complexas está no fato de que as arestas podem armazenar pesos. Podemos associar os pesos com maior valor às co-ocorrências mais frequentes entre as instâncias. Esta informação pode ser utilizada posteriormente pelo Sub-Modelo de Inferência para aperfeiçoar a classificação dos objetos em cena.

Uma segunda vantagem em se utilizar um modelo de Redes Complexas é que as arestas também podem carregar outras informações, tais como as condições das co-ocorrências. Por

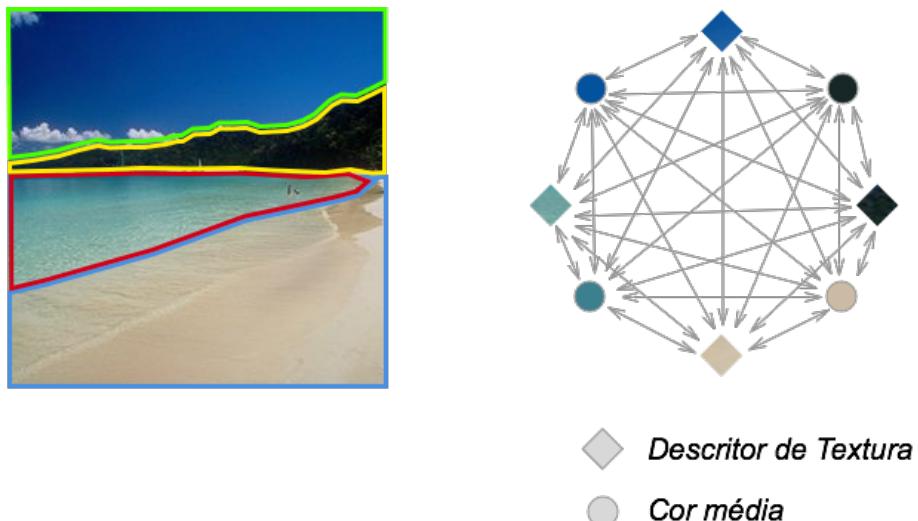


Figura 3.8: Representação das regiões da imagem a partir de duas características diferentes: cor média e textura. Todos os nós estão conectados entre, uma vez que todas as características co-ocorrem em uma mesma imagem.

exemplo, no caso da Figura 3.2, o céu co-ocorreu acima da praia. Pode-se então utilizar mais um atributo na aresta para guardar a informação de localização espacial relativa.

A terceira vantagem em se utilizar este modelo está no fato de que os nós também são entidades que podem ter seus atributos customizados. Isto é útil quando se deseja guardar informações extras, assim como qual é a característica que está sendo representada por ele. Na Figura 3.4 essa informação está sendo representada através das cores dos nós na Rede Complexa.

3.2.3.1 Construção da Rede Complexa

O Sub-Modelo Central é responsável por criar e gerenciar a Rede Complexa. Para construir uma rede de modo a minimizar inconsistências (ex. dois nós com a mesma instância de característica, duplicidade de arestas ou nós isolados), um conjunto de regras é definido.

A manutenção da Rede Complexa é feita sempre que uma nova imagem é analisada. Este módulo recebe um conjunto de informações obtidas a partir do Discretizador de Características sobre todas as regiões da imagem. Matematicamente, o conjunto dessas informações pode ser representado a partir da equação:

$$F = \{i_{m,k} : \forall m, k\} \quad (3.1)$$

Na Equação (3.1), m é o índice da característica que fora extraída da imagem e i_k é a instância

discretizada dessa característica em uma determinada região R_k da imagem.

A modelagem proposta sugere que cada nó da Rede Complexa seja uma representação de uma instância de característica. É importante destacar que uma mesma instância pode ser obtida a partir de duas imagens diferentes, bastando para isso elas possuírem o mesmo valor para que seja considerada a mesma instância. Isso significa que um nó poderá representar regiões de diversas imagens.

A relação entre duas instâncias de características é feita através das arestas da Rede Complexa. Essa relação pode ser criada utilizando diversos critérios. Um deles é associar a quantidade de co-ocorrência incrementando o peso da aresta com valores unitários. Um desvantagem dessa abordagem é que ela utiliza valores absolutos durante toda sua operação. Isso significa que não haverá um ponto de estabilidade no aprendizado e sim, somente incrementos.

Uma abordagem mais precisa é considerar uma unidade de tempo para que seja possível re-aprender as co-ocorrências. Apesar da abordagem anterior re-ponderar os pesos das arestas, ela não “desaprende” as co-ocorrências que não foram reforçadas com o tempo. Essa característica é importante quando se fala em aprendizado infinito. Matematicamente, essa abordagem pode ser modelada através da Equação (3.2):

$$w_{i,j} = w_{i,j} + \alpha \left[R \left(\frac{\lambda}{\Delta t} + 1 \right) - w_{i,j} \right] \quad (3.2)$$

Na equação acima, $w_{i,j}$ é o peso da aresta que liga o nó i ao nó j , $0 < \alpha \leq 1$ é uma constante denominada taxa de aprendizagem, R é um kernel de reforço, λ é uma constante potencializadora do tempo e Δt é uma variação do tempo. Para ilustrar o comportamento dessa equação na construção da Rede Complexa, considere a Figura 3.9. Nessa figura, pode-se observar que houve uma reponderação na aresta \overrightarrow{ij} para um valor maior. Dois fatores contribuíram para esse incremento. O primeiro deles é o valor baixo de Δt e o segundo é o alto valor de R . Isso significa que, quanto menor for a diferença do tempo entre as reponderações, maior será $w_{i,j}$. O kernel de reforço R é uma medida que representa o grau de relacionamento entre duas regiões que geraram a co-ocorrência. Nesse exemplo, consideramos $R = 1$ sempre que duas regiões co-ocorrem. Contudo, pode-se atribuir ao R o inverso da distância entre os centróides das regiões envolvidas

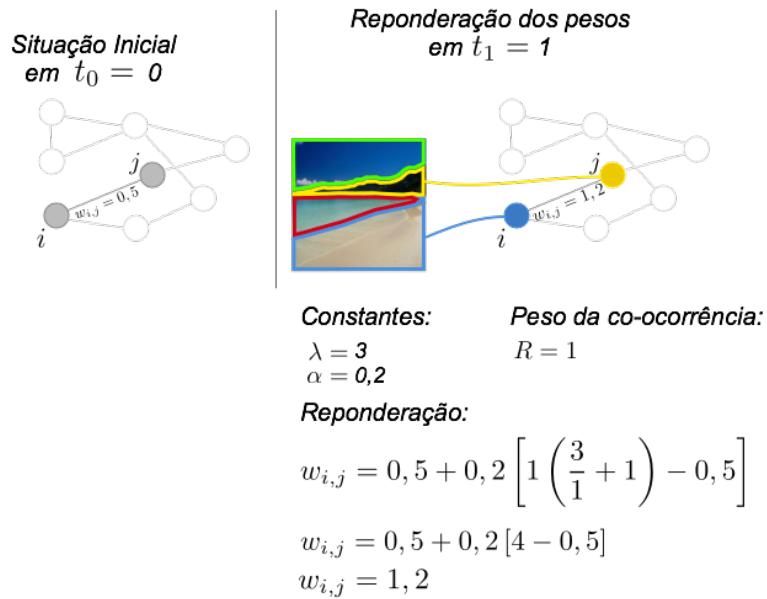


Figura 3.9: Reponderação da aresta \overline{ij} na Rede Complexa

na co-ocorrência. Isso significa que, quanto mais próximas duas regiões co-ocorrerem em uma mesma imagem, maior será o valor de R .

3.2.4 Modelo Intermediário de Inferência

O Sub-Modelo de Inferência é o componente responsável por fazer a análise de novas imagens. Esse modelo tem duas ligações com o Sub-Modelo Central. A primeira ligação é com a Rede Complexa e a segunda é com a re-alimentação para o aprendizado infinito.

Inicialmente, esse sub-modelo recebe uma imagem não-supervisionada diretamente em um segmentador. Esse segmentador gera um conjunto de segmentações e o envia para o extrator de características (módulo semelhante ao descrito na Seção 3.2.2.2). Após esta etapa, os conjuntos de instâncias de características de cada segmentação serão analisadas pelo maximizador de função. O maximizador de função contém um meta-modelo para escolher a melhor segmentação utilizando como base a Rede Complexa. A segmentação que for escolhida, terá suas regiões rotuladas pelos nós correspondentes a cada instância de característica. Finalmente, após a classificação e rotulação das regiões, esse resultado é apresentado ao construtor da rede para efetuar ajustes no modelo. As próximas seções detalham o processo descrito acima.

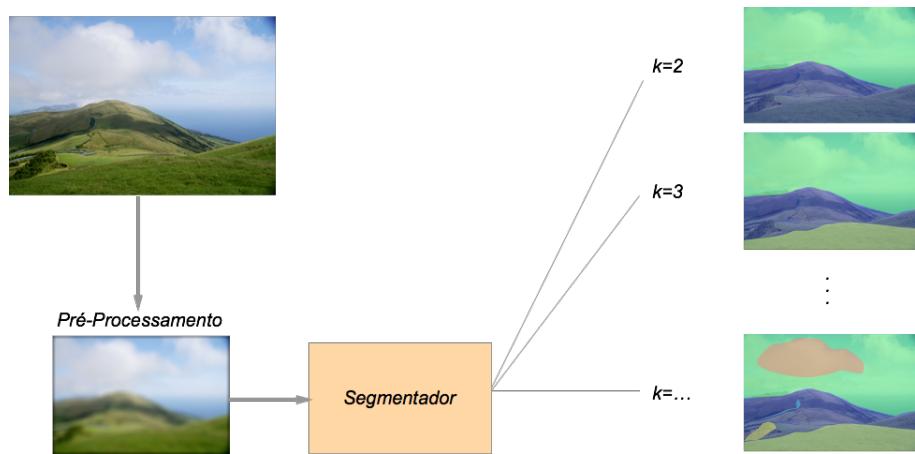


Figura 3.10: Exemplo de segmentações a partir de diferentes valores de k

3.2.4.1 Segmentador

O segmentador é o módulo responsável por gerar um conjunto de regiões para enviá-lo ao maximizador de função. Para essa tarefa, esse módulo é composto por uma etapa de pré-processamento e outra etapa de segmentação. A etapa de pré-processamento deve ser capaz de fazer um tratamento na imagem para eliminar grande parte dos ruídos. Essa etapa é necessária para evitar que os ruídos interfiram na segmentação. O pré-processamento pode ser implementado através de filtros gaussianos ou corte das altas frequências no domínio de Fourier. Após a etapa de pré-processamento, o algoritmo de segmentação recebe a imagem tratada.

Para efetuar a segmentação, o algoritmo de segmentação utiliza os valores de luminância dos pixels para determinar áreas semelhantes. Essas áreas semelhantes farão parte de uma mesma região. O que se espera do segmentador é que os pixels em uma mesma região sejam semelhantes entre si e os pixels em regiões diferentes sejam diferentes entre si.

Muitos segmentadores recebem como parâmetro um valor k para determinar quantas regiões deseja-se obter, tais como: k-means, baseados em algoritmos genéticos e baseados em redes neurais. Nesse trabalho, temos a possibilidade de variar o valor de k e armazenar todas as segmentações geradas no conjunto de saída. A Figura ilustra as diversas segmentações geradas a partir de diferentes valores de k . Alguns segmentadores serão estudados na fase experimental, tais como: k-menas, firefly baseado em histograma, region-growing, entre outros.

3.2.4.2 Extrator e Discretizador de Características

O módulo Extrator e Discretizador de Características do Sub-Modelo de Inferência são semelhantes ao Extrator e Discretizador de Características do Sub-Modelo de Treinamento descritos nas Seções 3.2.2.2 e 3.2.2.3. Contudo, neste módulo, um conjunto de segmentações S é apresentado na entrada, ao invés de uma única segmentação supervisionada. Para cada segmentação s_l , um conjunto de instâncias de características é extraído e discretizado utilizando as informações das regiões segmentadas. O resultado desse processo é enviado ao Maximizador de Função. Conforme explicado na Seção 3.2.3.1, o conjunto de instâncias de características é dado por $F = \{i_{m,k}\}$. Nesse módulo, uma segmentação s_l terá um conjunto de instâncias de características F_l definida por:

$$F_l = \{i_{m,k} : \forall m, k\} \quad (3.3)$$

onde c_m é o índice da característica, i_k é a instância de característica na região k da segmentação s_l . O resultado final desse módulo será o conjunto composto por todas as instâncias de características de todas as segmentações:

$$F = \{F_l : \forall l\} \quad (3.4)$$

3.2.4.3 Maximizador de Função

Das diversas segmentações geradas na etapa de segmentação, uma delas estará mais próxima da segmentação humana. O módulo responsável por escolher a melhor segmentação é o Maximizador de Função. Como o próprio nome diz, o maximizador de função encontra uma entrada s_{opt} para uma função f de tal forma que $f(s_{opt}) \geq f(s_i) \forall i$. Assim, $s_{opt} = \arg \max_{s_k} f(s_k)$.

Nesse trabalho, a função f utiliza a Rede Complexa do Sub-Modelo Central para dar uma “nota” a partir das instâncias de características F_l . Diversas funções podem ser estudadas, porém, nesse trabalho, focaremos na função ou-exclusivo, proposta inicialmente por (RIBEIRO; MUNTZ, 1996) e depois por (RODRIGUES; GIRALDI; ARAUJO, 2005).

$$f(s) = 1 - \prod_{(a,b) \in \chi \times \chi} \left(1 - \frac{1}{t} \sum_{(k,l)} w(i_{a,k}, i_{b,l}) \right) \quad (3.5)$$

Na Equação (3.5), χ é o conjunto de características (pág. 77), t é um fator para normalizar o somatório entre 0 e 1, $i_{a,k}$ é a instância da característica a na região R_k da segmentação s e $i_{b,l}$ é a instância da característica b na região R_l da segmentação s , $w(i_{a,k}, i_{b,l})$ é o peso da aresta que conecta o nó que representa a instância $i_{a,k}$ ao nó que representa a instância $i_{b,l}$ na Rede Complexa.

O comportamento dessa equação é o mesmo observado na Seção 2.3 (pág. 50); isto é, se um termo do produtório resultar em 1, a função f será 1. A Figura 3.11 ilustra o processo de escolha da melhor segmentação segundo a equação proposta.

Após identificar a melhor segmentação, os rótulos das regiões são preenchidos a partir dos rótulos que estão armazenados nos nós que foram utilizados para fazer a maximização. O resultado desse processo é uma imagem com as regiões delimitadas e com rótulos.

3.2.4.4 Aprendizado infinito

O Aprendizado Infinito é implementado através da re-alimentação da saída gerada pelo Maximizador de Função no Construtor da Rede. A re-alimentação ocorre sempre que uma nova imagem não supervisionada é apresentada ao classificador.

O que se espera desse modelo é que, a partir de um aprendizado inicial (realizado no Sub-Modelo de Treinamento), as novas classificações sejam suficientes para manter o sistema aprendendo, mesmo que não sejam apresentados dados supervisionados por humanos.

Um vez recebidos os dados, o construtor da rede utilizará as mesmas instâncias de características presentes na segmentação escolhida para re-ponderar os pesos dos nós correspondentes. Essa etapa de re-ponderação é semelhante ao processo descrito na Seção 3.2.3.1

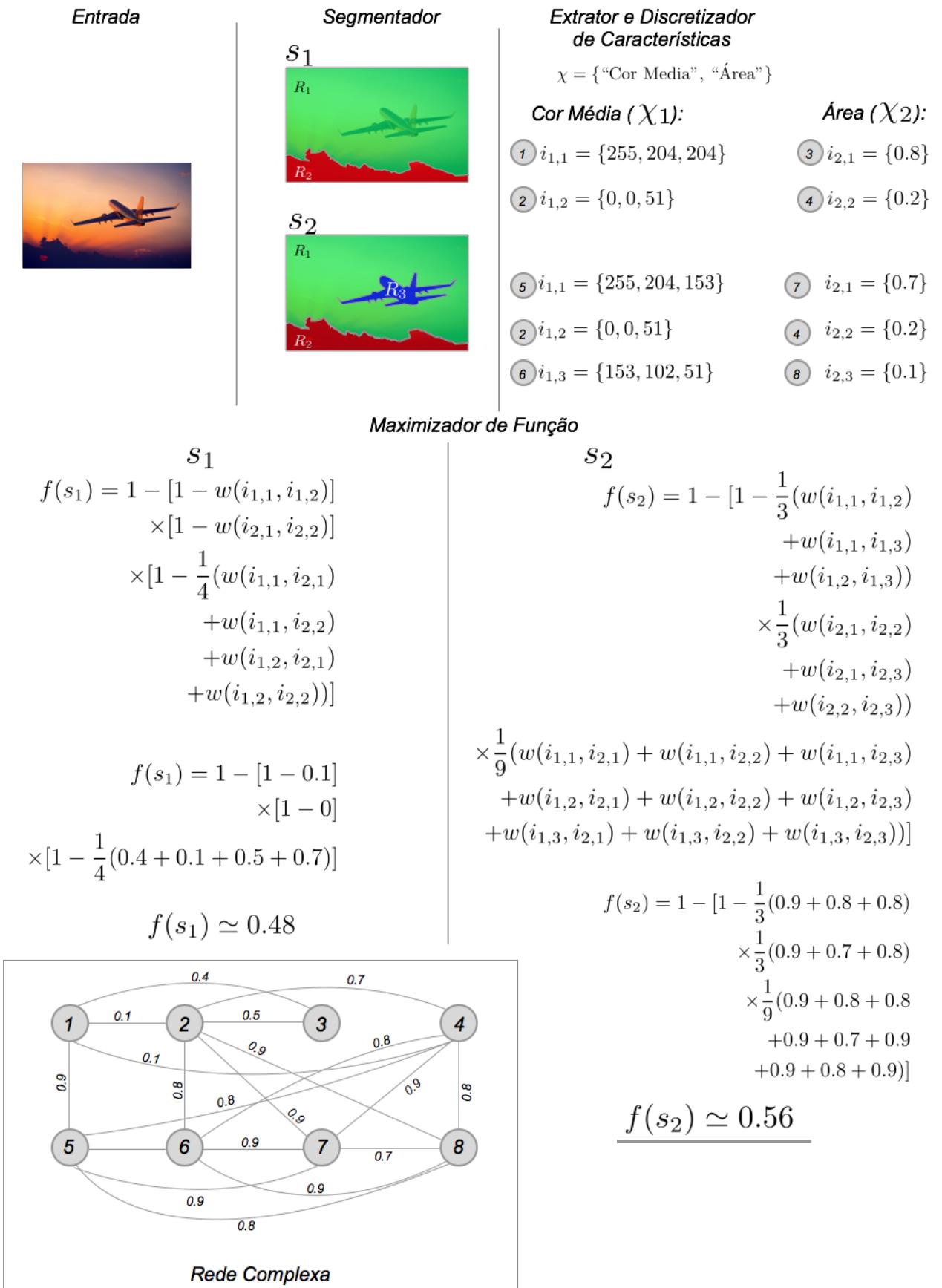


Figura 3.11: Ilustração do processo de escolha da melhor segmentação. Segundo a Equação (3.5), a melhor segmentação é s_2 .

4 PROPOSTA EXPERIMENTAL

O meta-modelo proposto na Seção 3 contém diversos módulos que podem ser ajustados para obter maior performance do classificador. Este capítulo apresenta alguns experimentos que serão realizados com o objetivo de estudar tanto o comportamento do meta-modelo como também analisar os aspectos da neurociência considerados no classificador.

As próximas seções estão organizadas de acordo com os sub-modelos apresentados na Seção 3.2. As Seções 4.1, 4.2 e 4.3 apresentam os experimentos a serem realizados respectivamente no Sub-Modelos de Treinamento, Central e Inferência.

4.1 Sub-Modelo de Treinamento

O Sub-Modelo de Treinamento contém dois principais aspectos que podem interferir nos Sub-Modelos seguintes. Um deles é a escolha das características que devem ser consideradas para a construção da rede. O segundo aspecto está relacionado ao grau de discretização dessas características para que se tenha uma quantidade de nós relativamente alta porém, com alta densidade de arestas. As duas próximas seções detalham os experimentos a serem feitos no Sub-Modelo de Treinamento.

4.1.1 Definição das Características a Serem Extraídas das Regiões

Este experimento consiste em analisar a contribuição de um conjunto de características consideradas no modelo. Para a realização desse experimento, serão propostos alguns conjuntos de características. Algumas dessas características são descritas na Seção 2.1.1.

Uma vez definido os conjuntos de características, o sistema realizará a segmentação e classificação dos objetos de algumas imagens supervisionadas pré-definidas. A medição da qualidade de cada conjunto será feita considerando as áreas das regiões obtidas na saída do classificador. Essa medida será proporcional à semelhança entre as regiões obtidas da saída do classificador com

relação às regiões supervisionadas.

4.1.2 Grau de Discretização das Instâncias de Características

Conforme descrito na Seção 3.2.2.2, a quantidade de nós, bem como a quantidade de arestas, são dependentes do grau de discretização das instâncias de características. Para uma discretização com poucos pontos, a quantidade de nós será baixa, porém a quantidade de arestas será alta. Esse experimento estuda a relação de número de nós com o número de arestas que apresenta a maior modularidade (veja Seção 2.4.3.10) da rede.

Nesse experimento, utilizamos uma medida de comparação entre a rede complexa gerada em cada experimento com uma rede complexa aleatória com o mesmo número de nós e arestas. A comparação é feita utilizando o Coeficiente de Clusterização Médio (veja Seção 2.4.3.3). A comparação das redes complexas geradas com uma rede aleatória de mesmas características é uma forma de medir o grau de organização topológica da rede.

Esse experimento consiste em definir diversos níveis de discretizações e gerar, para cada discretização, uma rede complexa treinada a partir da base de dados “SUN Database”. Esse treinamento é feito utilizando a mesma técnica descrita na Seção 3.2.2.

Uma vez criada, cada rede complexa será avaliada utilizando a razão do coeficiente de clusterização de rede gerada com uma rede complexa aleatória com a mesma quantidade de nós e arestas. Os resultados serão discutidos considerando cada característica discretizada.

4.2 Sub-Modelo Central

O Sub-Modelo Central é o componente que gerencia a rede complexa a partir dos dados provenientes dos sub-modelos vizinhos. Um dos principais pontos de estudo nesse sub-modelo é a equação de co-ocorrência (veja Equação (3.2)). A próxima seção descreve o experimento a ser realizado com a equação de co-ocorrência.

4.2.1 Estudo Sobre a Equação de Co-Ocorrência

O experimento proposto nesta seção tem por objetivo estudar o comportamento da rede complexa através da variação dos parâmetros da Equação (3.2). Nessa equação, três parâmetros ajustáveis são utilizados para re-ponderar os pesos da rede complexa.

O primeiro parâmetro a ser testado nesse experimento é a taxa de aprendizagem α . A taxa de aprendizagem é um parâmetro que determina o quanto um novo valor, recentemente descoberto, influencia na atualização do peso de uma aresta. Quanto mais próximo a um, maior será a influência da atualização. Contudo, quanto mais próximo a zero, mais estável será a atualização.

Outro parâmetro a ser avaliado é o kernel R da função. No exemplo descrito na Seção 3.2.3.1 foi considerado que esse kernel sempre teria valor 1. Contudo, uma medida de relação entre as regiões da imagem envolvidas na co-ocorrência pode ser utilizada. Uma possível medida é utilizar uma gaussiana em função da distância entre os centróides das regiões. Quanto mais próximas as duas regiões estiverem, maior será o valor do kernel. Isso significa que somente regiões próximas terão alta influência no peso da aresta da rede complexa.

O terceiro parâmetro ajustável é o λ . Este parâmetro potencializa a influência do tempo na re-ponderação; isto é, quanto maior o valor de λ maior será a influência do tempo na re-ponderação.

Para cada caso de teste, três valores serão escolhidos para α , R e λ . A qualidade da combinação dos três parâmetros será medida através do coeficiente de clusterização médio da rede complexa gerada em cada caso de teste (veja Seção 2.4.3.3). O valor do coeficiente de clusterização médio obtido será comparado com o coeficiente de clusterização médio para um rede complexa aleatória de mesmas características (número de nós e número de arestas). Quanto maior a diferença entre as duas medições, melhor será considerada a configuração dos parâmetros, uma vez que houve maior organização topológica da rede complexa (maior modularidade).

4.3 Sub-Modelo de Inferência

Para que o Sub-Modelo de Inferência tenha maior taxa de acerto, tanto na escolha da melhor segmentação quanto na rotulagem dos objetos, dois fatores podem ser considerados. O primeiro deles é o conjunto de segmentações enviado ao maximizador de função. O segundo fator é a própria função de maximização considerada pelo maximizador. Esses dois fatores farão parte dos experimentos apresentados nas próximas seções.

4.3.1 Experimentos na Etapa de Segmentação

Conforme descrito na Seção 3.2.4.3, o maximizador de função deve escolher a melhor segmentação segundo a Equação (3.5). Contudo, o conjunto de segmentações enviado ao maximizador deve conter, pelo menos, uma segmentação mais próxima à supervisão humana. Nesse experimento, estamos interessados em investigar quais parâmetros do segmentador torna o conjunto de segmentações mais próximo das supervisões humanas.

Para que este estudo seja feito, alguns segmentadores propostos pela literatura, tais como: k-means, clusterização hierárquica, algoritmos genéticos e firefly.

A avaliação dos algoritmos testados será feita utilizando uma medida de distância entre imagens segmentadas por humanos e imagens segmentadas pelo segmentador.

Como proposta inicial, a diferença entre duas segmentações será dada pela comparação do contorno de cada região segmentada com cada região supervisionada. Os contornos das regiões, tanto segmentados quanto supervisionados, serão comparados através da distância PDM (veja o artigo (ERDMANN et al., 2013) em anexo). A nota final de uma segmentação será dada através do somatório das menores distâncias PDM entre cada região segmentada por todas as regiões supervisionadas.

4.3.2 Estudo Sobre a Função do Maximizador

Um ponto importante que deve ser considerado no modelo proposto é a função que deve ser maximizada. Este experimento tem por objetivo obter uma análise sobre como esta função influencia o

resultado final da segmentação e rotulagem das regiões.

A primeira etapa deste estudo consiste em desmembrar os termos do produtório da Equação (3.5). Cada termo desse produtório é um somatório dos pesos das arestas relacionadas a duas características específicas ($(a, b) \in \chi \times \chi$). Isso significa que, ao desmembrar-mos esses termos, pode-se analisar quanto a relação entre duas características (diferentes ou não) influenciam no maximizador.

A segunda etapa deste estudo consiste em elaborar outras funções para o maximizador considerando os resultados obtidos na etapa anterior. Por exemplo, podemos considerar que cada termo da Equação (3.5) tenha um fator de potencialização $u_{a,b}$:

$$f(s) = 1 - \prod_{(a,b) \in \chi \times \chi} \left(1 - \frac{1}{t} \sum_{(k,l)} w(i_{a,k}, i_{b,l}) \right)^{u_{a,b}} \quad (4.1)$$

Dessa forma, cada termo poderá ser potencializado de acordo com os resultados obtidos anteriormente.

5 EXPERIMENTOS E RESULTADOS PRELIMINARES

A hipótese central desta tese, conforme apresentada na Seção 3.2, é que um modelo de co-ocorrência entre as instâncias de características é o principal fator que pode melhorar a qualidade da classificação de um sistema de reconhecimento de objetos.

Com o objetivo de estudar esta hipótese, foi proposto um experimento para analisar o comportamento das co-ocorrências utilizando a base supervisionada de imagens “SUN Database”. Nesse experimento, somente os dados supervisionados foram utilizados e nenhum processamento de imagem foi realizado.

A Seção 3.1 apresentou o formato dos arquivos XML de supervisão da “SUN Database”. Nesse experimento, todos os 16.873 arquivos XML de supervisão foram utilizados para computar o número de co-ocorrência entre todos os pares de rótulos. Define-se por co-ocorrência todos os pares de objetos diferentes que aparecem em uma mesma cena. O Algoritmo 1 descreve o processo. O Algoritmo 1 recebe como entrada o caminho da pasta dos arquivos supervisionados. Um dicionário, D que associa um par de rótulos ao número de co-ocorrências é inicializado. Para cada arquivo, em formato XML, o algoritmo percorre todos os pares de co-ocorrências, extraíndo rótulos envolvidos. Assim que cada par de rótulos é descoberto, o algoritmo verifica se esse par já existe no dicionário de co-ocorrências. Caso não exista, ele cria uma entrada e atribui o valor 1. Caso a entrada já exista, seu valor é atualizado através de um incremento.

O dicionário de saída D do algoritmo relaciona um par de rótulos ao número de co-ocorrências com os outros objetos da cena. Esse dicionário foi ordenado de forma decrescente por contagem de co-ocorrências e as 30 co-ocorrências mais altas são mostradas na Tabela 5.1. Nessa tabela, nota-se que os primeiros 6 rótulos co-ocorrem com os mesmos rótulos. É importante destacar que apesar dos rótulos serem os mesmos, os objetos são diferentes; isto é, existe mais do que um objeto de mesmo rótulo na cena.

Nessa tabela, pode-se notar que os rótulos que co-ocorrem têm uma relação semântica (de

Rótulo 1	Rótulo 2	Número de co-ocorrências
window	window	168680
chair_occluded	chair_occluded	38914
books	books	31040
ceiling_lamp	wall	22784
floor	wall	22416
window	building	19503
person_sitting_occluded	person_sitting_occluded	19422
ceiling	wall	18887
window	wall	18248
wall	chair_occluded	15106
box	box	14466
plants	plants	13830
tree	window	13277
ball	ball	11046
wall	cabinet	10332
sky	window	10160
trees	trees	9938
door	window	9786
chair_occluded	ceiling_lamp	9495
cabinet	cabinet	9296
ceiling_lamp	ceiling	8520
magazines	magazines	7912
wall	painting	7621
floor	ceiling_lamp	7320
book	book	7242
tree	trees	7136
wall	door	7128
wall	curtain	7122
person_occluded	wall	6635
picture	wall	6471

Tabela 5.1: As 30 co-ocorrências mais altas na base “SUN Database”.

Entrada: P - Pasta Dos Arquivos de Supervisão
Saída: D - Dicionário Contendo o número de co-ocorrências para cada par de rótulos (r_1, r_2)

```

 $D \leftarrow$  Inicializa Dicionário Vazio
para cada Arquivo  $F$  Dentro da Pasta  $P$  faça
    para cada Para cada objeto  $o_1$  em  $F$  faça
        para cada Para cada rótulo  $o_2$  em  $F$  faça
            se  $o_1 \neq o_2$  então
                 $r_1 \leftarrow$  Extrair Conteúdo da tag “<name>” do objeto  $o_1$ 
                 $r_2 \leftarrow$  Extrair Conteúdo da tag “<name>” do objeto  $o_2$ 
                 $c \leftarrow (r_1, r_2)$ 
                se  $c$  existe em  $D$  então
                     $D[c] = D[c] + 1$ 
                senão
                     $D[c] = 1$ 
                fim
            fim
        fim
    fim
retorna  $D$ 

```

Algorithm 1: Algoritmo para contagem de co-ocorrências na base “SUN Database”

mesmo contexto) entre si, tais como: “floor” e “wall”, “ceiling” e “wall”, “window” e “sky”, entre outros. Este pode ser um indício de que haja formação de clusters na Rede Complexa estudada.

Este indício induz a realização de um segundo experimento: analisar a distribuição de graus da Rede Complexa gerada. Para este experimento, uma varredura sobre toda a Rede foi realizada para gerar uma lista contendo o grau de cada nó (veja Seção 2.4.3.1). Nesse experimento, consideramos que o grau $d(u)$ de cada nó u é o somatório de todos os pesos das arestas que incidem sobre ele, de acordo com a Equação (5.1):

$$d(u) = \sum_v w(u, v) \quad (5.1)$$

O histograma de graus é apresentado na Figura 5.1. Conforme pode ser observado, o comportamento da distribuição de graus parece seguir a lei de potência, uma vez que há muitos nós com baixo grau e poucos nós com alto grau. Esse é mais um indício que a Rede Complexa estudada segue o comportamento de Redes Livres de Escala.

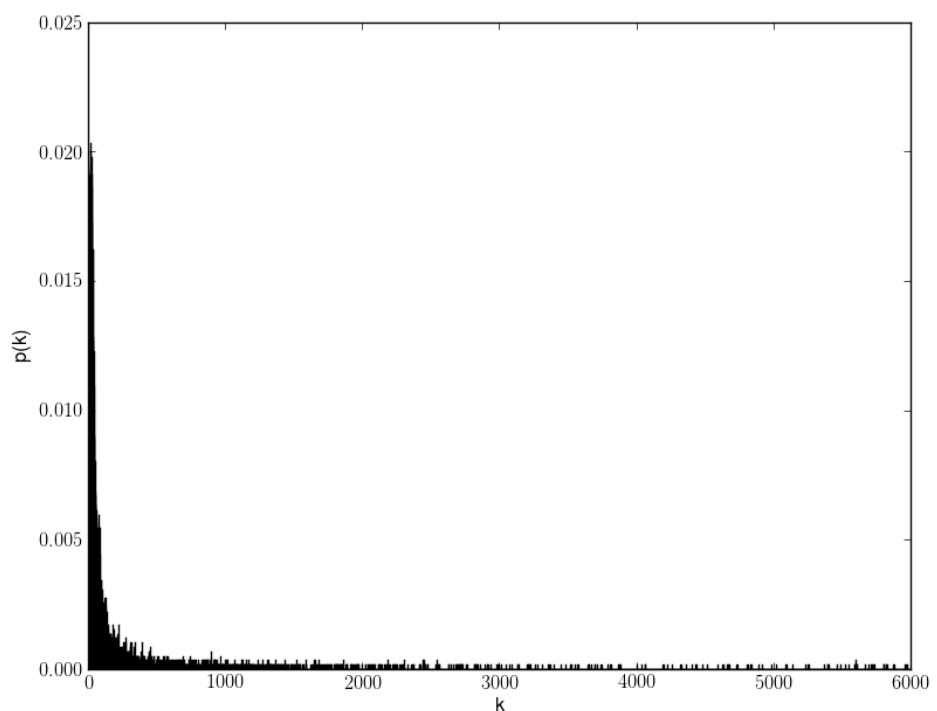


Figura 5.1: Histograma de graus segundo a Equação (5.1)

6 CRONOGRAMA

A Fig. 6.1 apresenta o cronograma para a conclusão da tese. A defesa final não poderá exceder o dia 16/09/2016.

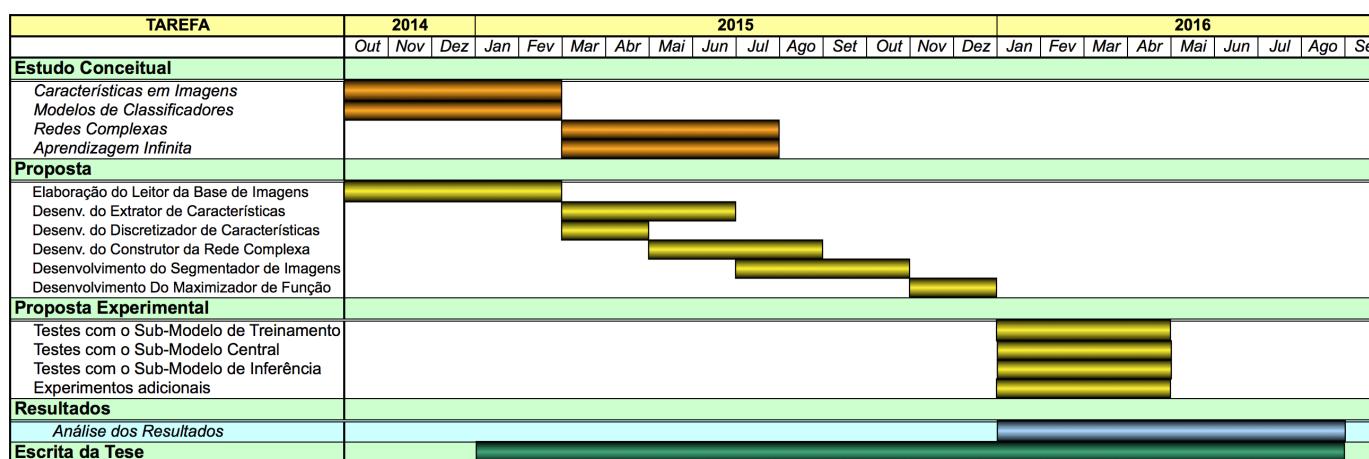


Figura 6.1: Cronograma para o próximo período de 24 meses

REFERÊNCIAS

- AGRAWAL, P.; GIRSHICK, R.; MALIK, J. Analyzing the performance of multilayer neural networks for object recognition. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. [S.l.: s.n.], 2014.
- AHARON, M.; ELAD, M.; BRUCKSTEIN, A. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *Signal Processing, IEEE Transactions on*, IEEE, v. 54, n. 11, p. 4311–4322, nov. 2006. ISSN 1053-587X.
- ALBERT, R.; BARABÁSI, A. L. Statistical mechanics of complex networks. *Reviews of Modern Physics*, American Physical Society, v. 74, n. 1, p. 47–97, Jan 2002.
- ALBERT, R.; JEONG, H.; BARABASI, A. L. The diameter of the world wide web. *Nature*, v. 401, p. 130–131, 1999.
- ALBERT, R.; JEONG, H.; BARABÁSI, A.-L. Attack and error tolerance of complex networks. *Nature*, n. 406, p. 376–382, 2000.
- ALLESINA, S.; PASCUAL, M. Network structure, predator prey modules, and stability in large food webs. *Theoretical Ecology*, v. 1, n. 1, p. 55–64, mar. 2008.
- ANDERSON, C. H.; ESSEN, D. C. V. Shifter circuits: A computational strategy for dynamic aspects of visual processing. *National Academy of Sciences*, v. 84, p. 6297–6301, 1987.
- ANDREOPoulos, A.; TSOTSOS, J. K. 50 years of object recognition: Directions forward. *Computer Vision and Image Understanding*, v. 117, n. 8, p. 827–891, 2013.
- ARKIN, R. C. *An Behavior-based Robotics*. 1st. ed. Cambridge, MA, USA: MIT Press, 1998. ISBN 0262011654.
- BAARS, B.; GAGE, N. *Cognition, Brain, and Consciousness: Introduction to Cognitive Neuroscience*. Elsevier Science, 2010. (Introduction to Cognitive Neuroscience Series). ISBN 9780123814401. Disponível em: <<http://books.google.com.br/books?id=lEDyN5-80E8C>>.
- BACKES, A. R.; BRUNO, O. M. Shape classification using complex network and multi-scale fractal dimension. *Pattern Recognition Letters*, v. 31, n. 1, p. 44–51, 2010.
- BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. A complex network-based approach for boundary shape analysis. *Pattern Recognition*, v. 42, n. 1, p. 54–67, 2009.
- BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. Texture analysis and classification: A complex network-based approach. *Inf. Sci.*, v. 219, p. 168–180, 2013.
- BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. Texture analysis and classification: A complex network-based approach. *Inf. Sci.*, Elsevier Science Inc., New York, NY, USA, v. 219, p. 168–180, jan. 2013. ISSN 0020-0255.
- BANDERA, C.; VICO, F. J.; BRAVO, J. M.; HARMON, M. E.; III, L. C. B. Residual q-learning applied to visual attention. In: SAITTA, L. (Ed.). *ICML*. [S.l.]: Morgan Kaufmann, 1996. p. 20–27. ISBN 1-55860-419-7.

- BARABÁSI, A.-L.; ALBERT, R. Emergence of scaling in random networks. *Science*, American Association for the Advancement of Science, Department of Physics, University of Notre Dame, Notre Dame, IN 46556, USA., v. 286, n. 5439, p. 509–512, out. 1999. ISSN 1095-9203.
- BARABASI, A.-L.; BONABEAU, E. Scale-free networks. *Scientific American*, p. 50–59, Mai 2003.
- BARRAT, A.; BARTHÉLEMY, M.; PASTOR-SATORRAS, R.; VESPIGNANI, A. The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences of the United States of America*, v. 101, n. 11, p. 3747–3752, mar. 2004.
- BAY, H.; ESS, A.; TUYTELAARS, T.; GOOL, L. V. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, Elsevier Science Inc., New York, NY, USA, v. 110, n. 3, p. 346–359, jun. 2008. ISSN 1077-3142.
- BEHNKE, S. *Hierarchical Neural Networks for Image Interpretation*. [S.l.]: Springer-Verlag, 2003. (Lecture Notes in Computer Science, v. 2766).
- BO, L.; REN, X.; FOX, D. Hierarchical matching pursuit for image classification: Architecture and fast algorithms. In: *NIPS*. [S.l.: s.n.], 2011. p. 2115–2123.
- BO, L.; REN, X.; FOX, D. Unsupervised Feature Learning for RGB-D Based Object Recognition. In: *ISER*. [S.l.: s.n.], 2012.
- BOSCH, A.; ZISSERMAN, A.; NOZ, X. M. Image classification using random forests and ferns. In: *ICCV*. [S.l.]: IEEE, 2007. p. 1–8.
- BOSE, S.; BODINE-BARON, E.; HASSIBI, B.; WIERNAN, A. The cost of an epidemic over a complex network: A random matrix approach. *CoRR*, abs/1309.2236, 2013.
- BOYNTON, G. M. Attention and visual perception. *Current Opinion in Neurobiology*, v. 15, n. 4, p. 465 – 469, 2005. ISSN 0959-4388. Sensory systems.
- BRODER, A.; KUMAR, R.; MAGHOUL, F.; RAGHAVAN, P.; RAJAGOPALAN, S.; STATA, R.; TOMKINS, A.; WIENER, J. Graph structure in the web. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 33, n. 1-6, p. 309–320, 2000. ISSN 1389-1286.
- CALDARELLI, G.; CATANZARO, M. *Networks: A Very Short Introduction*. United Kingdom: Oxford University Press, Inc., 2012.
- CALVIN, W.; OJEMANN, G. *Conversations with Neil's brain: the neural nature of thought and language*. Addison-Wesley Pub. Co., 1994. (A William Patrick Book). ISBN 9780201632170. Disponível em: <<http://books.google.com.br/books?id=iKLuAAAAMAAJ>>.
- CANCHO, R. F. ; SOLÉ, R. V. The small world of human language. *Proceedings of The Royal Society of London. Series B, Biological Sciences*, v. 268, p. 2261–2266, 2001.
- CARLSON, A.; BETTERIDGE, J.; KISIEL, B.; SETTLES, B.; JR., E. R. H.; MITCHELL, T. M. Toward an architecture for never-ending language learning. In: *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)*. [S.l.: s.n.], 2010.
- CASANOVA, D.; BACKES, A. R.; BRUNO, O. M. Pattern recognition tool based on complex network-based approach. *Journal of Physics: Conference Series*, v. 410, n. 1, p. 012048, 2013. Disponível em: <<http://stacks.iop.org/1742-6596/410/i=1/a=012048>>.

- CHALUMEAU L. DA F. COSTA, O. L. F. Texture discrimination using hierarchical complex networks. In: *Proceedings of the Second International Conference on Signal-Image Technology and Internet-Based Systems*. [S.l.: s.n.], 2012. p. 543–550.
- CHANG, L.; DUARTE, M. M.; SUCAR, L. E.; MORALES, E. F. Object class recognition using sift and bayesian networks. In: *MICAI (2)*. [S.l.: s.n.], 2010. p. 56–66.
- CHELLA, A.; MANZOTTI, R. *Artificial consciousness*. [S.l.]: Imprint Academic, 2007. ISBN 9781845400705.
- CHENG, M.-M.; MITRA, N. J.; HUANG, X.; TORR, P. H. S.; HU, S.-M. *Salient Object Detection and Segmentation*. [S.l.], Oct 2011. Submission NO. TPAMI-2011-10-0753. Disponível em: <<http://mmcheng.net/salobj/>>.
- COATES, A.; NG, A. The importance of encoding versus training with sparse coding and vector quantization. In: GETOOR, L.; SCHEFFER, T. (Ed.). *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. New York, NY, USA: ACM, 2011. (ICML '11), p. 921–928. ISBN 978-1-4503-0619-5.
- COHEN, M. C.; ALVAREZ, G.; NAKAYAMA, K. Natural scene perception requires attention. *Psychological Science*, 2011.
- COHEN, R.; EREZ, K.; BEN-AVRAHAM, D.; GAVLIN, S. Resilience of the internet to random breakdowns. *Phys. Rev. Lett.*, n. 85, p. 3682–3685, 2000.
- COLLINS, M.; SINGER, Y. Unsupervised models for named entity classification. In: *In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*. [S.l.: s.n.], 1999. p. 100–110.
- CORBETTA, M.; MIEZIN, F. M.; DOBMEYER, S.; SHULMAN, G. L.; PETERSEN, S. E. Attentional modulation of neural processing of shape, color, and velocity in humans. *Science*, v. 248, p. 1556–1559, 1990.
- CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. *Introduction to Algorithms*. 2nd revised edition. ed. [S.l.]: The MIT Press, 2001. Taschenbuch. ISBN 0262531968.
- COSTA, L. da F. Systems biology through complex networks, signal processing, image analysis, and artificial intelligence. In: *Proceedings of the 16th International Conference on Digital Signal Processing*. Piscataway, NJ, USA: IEEE Press, 2009. (DSP'09), p. 1306–1313. ISBN 978-1-4244-3297-4.
- CUADROS, O.; BOTELHO, G.; RODRIGUES, F.; NETO, J. B. Segmentation of large images with complex networks. *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 24–31, 2012. ISSN 1530-1834.
- DAYAN, P.; SAHANI, M.; DEBACK, G. Unsupervised learning. In: *In The MIT Encyclopedia of the Cognitive Sciences*. [S.l.]: The MIT Press, 1999.
- DESIMONE, R. Visual attention mediated by biased competition in extrastriate visual cortex. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, v. 353, n. 1373, p. 1245–1255, ago. 1998. ISSN 0962-8436.

- DICKINSON, S. J. Rutgers university lectures on cognitive science. In: _____. [S.I.]: Basil Blackwell, 1999. cap. Object Representatino and Recognition, p. 172–207.
- DRAPER, B. A.; BINS, J.; BAEK, K. Adore: Adaptive object recognition. In: CHRISTENSEN, H. I. (Ed.). *ICVS*. [S.I.]: Springer, 1999. (Lecture Notes in Computer Science, v. 1542), p. 522–537. ISBN 3-540-65459-3.
- DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification (2nd Edition)*. 2. ed. [S.I.]: Wiley-Interscience, 2000. Hardcover. ISBN 0471056693.
- DUNNE, J. A.; WILLIAMS, R. J.; MARTINEZ, N. D. Network structure and robustness of marine food webs. *Marine Ecology-Progress Series*, v. 273, p. 291–302, 2004.
- ERDMANN, H.; LOPES, L.; WACHS-LOPES, G.; RIBEIRO, M.; RODRIGUES, P. A study of a firefly meta-heuristics for multithreshold image segmentation. *Computational Vision and Medical Image Processing IV: VIPIMAGE 2013*, CRC Press, p. 211, 2013.
- ERDŐS, P.; RÉNYI, A. On random graphs. I. *Publ. Math. Debrecen*, v. 6, p. 290–297, 1959.
- FALOUTSOS, M.; FALOUTSOS, P.; FALOUTSOS, C. On power-law relationships of the internet topology. In: *SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*. New York, NY, USA: ACM, 1999. p. 251–262. ISBN 1-58113-135-6.
- FARKAS, I. J.; JEONG, H.; VICSEK, T.; BARABÁSI, A. L.; OLTVAI, Z. N. The topology of the transcription regulatory network in the yeast, *s. cerevisiae*. *Physica A*, v. 318, n. cond-mat/0205181, p. 3–4. 18 p, May 2002.
- FUKUSHIMA, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, v. 36, p. 193–202, 1980.
- GLATTFELDER, J. *Ownership Networks and Corporate Control: Mapping Economic Power in a Globalized World*. ETH, 2010. Disponível em: <<http://books.google.com.br/books?id=wwzUXwAACAAJ>>.
- GONZALEZ, R.; WOODS, R. *Digital Image Processing*. 2nd. ed. New Jersey: Prentice-Hall, 2002.
- GOVINDAN, R.; TANGMUNARUNKIT, H. Heuristics for internet map discovery. *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, v. 3, p. 1371–1380 vol.3, 2000.
- GUELZIM, N.; BOTTANI, S.; BOURGINE, P.; KEPES, F. Topological and causal structure of the yeast transcriptional regulatory network. *Nat Genet*, v. 31, n. 1, p. 60–3, 2002. ISSN 1061-4036.
- HADSELL, R.; SERMANET, P.; BEN, J.; ERKAN, A.; HAN, J.; MULLER, U.; LECUN, Y. Online learning for offroad robots: Spatial label propagation to learn long-range traversability. In: *Proceedings of Robotics: Science and Systems*. Atlanta, GA, USA: [s.n.], 2007.
- HARALICK, R. Statistical and structural approaches to texture. *Proceedings of the IEEE*, v. 67, n. 5, p. 786–804, May 1979. ISSN 0018-9219.

- HARARY, F. *Graph Theory*. [S.l.]: Addison-Wesley, 1969.
- HIDALGO, C. A.; KLINGER, B.; BARABÁSI, A.-L.; HAUSMANN, R. The product space conditions the development of nations. *Science*, v. 317, p. 482, July 2007.
- HILGETAG, C. C.; BURNS, G. A.; O'NEILL, M. A.; SCANNELL, J. W.; YOUNG, M. P. Anatomical connectivity defines the organization of clusters of cortical areas in the macaque monkey and the cat. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, Department of Psychology, University of Newcastle upon Tyne, UK. claus@bu.edu, v. 355, n. 1393, p. 91–110, January 2000. ISSN 0962-8436.
- HOWARTH, P.; RÜGER, S. Evaluation of texture features for content-based image retrieval. In: *In: Proceedings of the International Conference on Image and Video Retrieval, Springer-Verlag*. [S.l.: s.n.], 2004.
- HOWARTH, P.; RÜGER, S. M. Evaluation of texture features for content-based image retrieval. In: *CIVR*. [S.l.]: Springer, 2004. (Lecture Notes in Computer Science, v. 3115), p. 326–334. ISBN 3-540-22539-0.
- HUTTENLOCHER, D. P.; ULLMAN, S. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, v. 5, n. 2, p. 195–212, 1990.
- KHAN, F. S.; WEIJER, J. van de; VANRELL, M. Top-down color attention for object recognition. In: *ICCV*. [S.l.]: IEEE, 2009. p. 979–986.
- LAAR, P. van de; HESKES, T.; GIELEN, S. Task-dependent learning of attention. *Neural Networks*, v. 10, n. 6, p. 981 – 992, 1997. ISSN 0893-6080. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0893608097000312>>.
- LAN, T.; RAPTIS, M.; SIGAL, L.; MORI, G. From subcategories to visual composites: A multi-level framework for object detection. In: . [S.l.: s.n.], 2013.
- LAZEBNIK, S.; SCHMID, C.; PONCE, J. A sparse texture representation using local affine regions. *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, Washington, DC, USA, v. 27, n. 8, p. 1265–1278, ago. 2005. ISSN 0162-8828.
- LEE, H.; GROSSE, R.; RANGANATH, R.; NG, A. Y. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: *Proceedings of the 26th Annual International Conference on Machine Learning*. New York, NY, USA: ACM, 2009. (ICML '09), p. 609–616. ISBN 978-1-60558-516-1.
- LILJEROS, F.; EDLING, C. R.; AMARAL, L. A.; STANLEY, E. H.; ÅBERG, Y. The web of human sexual contacts. *Nature*, Nature Publishing Group, v. 411, n. 6840, p. 907–908, June 2001. ISSN 0028-0836.
- LOWE, D. G. Three-dimensional object recognition from single two-dimensional images. *Artif. Intell.*, Elsevier Science Publishers Ltd., Essex, UK, v. 31, n. 3, p. 355–395, mar. 1987. ISSN 0004-3702.
- LOWE, D. G. Object recognition from local scale-invariant features. In: *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*. Washington, DC, USA: IEEE Computer Society, 1999. (ICCV '99), p. 1150–. ISBN 0-7695-0164-8.

- LOWE, D. G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, Kluwer Academic Publishers, Hingham, MA, USA, v. 60, n. 2, p. 91–110, nov. 2004. ISSN 0920-5691.
- MARR, D. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. New York, NY, USA: Henry Holt and Co., Inc., 1982. ISBN 0716715678.
- MENG, H.; APPIAH, K.; HUNTER, A.; DICKINSON, P. Fpga implementation of naive bayes classifier for visual object recognition. In: . [S.l.: s.n.], 2011. p. 123–128.
- MESNIL, G.; RIFAI, S.; BORDES, A.; GLOROT, X.; BENGIO, Y.; VINCENT, P. Unsupervised and transfer learning under uncertainty - from object detections to scene categorization. In: MARSICO, M. D.; FRED, A. L. N. (Ed.). *ICPRAM*. [S.l.]: SciTePress, 2013. p. 345–354. ISBN 978-989-8565-41-9.
- MILGRAM, S. The small world problem. *Psychology Today*, v. 2, p. 60–67, 1967.
- MINUT, S.; MAHADEVAN, S. A reinforcement learning model of selective visual attention. In: *Proceedings of the Fifth International Conference on Autonomous Agents*. New York, NY, USA: ACM, 2001. (AGENTS '01), p. 457–464. ISBN 1-58113-326-X.
- MITCHELL, T. M. *Machine Learning*. 1. ed. New York, NY, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072.
- MORAN, J.; DESIMONE, R. Selective attention gates visual processing in the extrastriate cortex. *Science*, v. 229, p. 782–784, 1985.
- MOTTAGHI, R.; CHEN, X.; LIU, X.; CHO, N.-G.; LEE, S.-W.; FIDLER, S.; URTASUN, R.; YUILLE, A. The role of context for object detection and semantic segmentation in the wild. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2014.
- NABOUT, A. A. Object shape recognition using wavelet descriptors. *Journal of Engineering*, 2013.
- NEAPOLITAN, R. E. *Learning Bayesian Networks*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2003. ISBN 0130125342.
- NEWELL, M. A.; COOK, D.; HOFMANN, H.; JANNINK, J.-L. An algorithm for deciding the number of clusters and validation using simulated data with application to exploring crop population structure. *The Annals of Applied Statistics*, The Institute of Mathematical Statistics, v. 7, n. 4, p. 1898–1916, 12 2013.
- NEWMAN, M. *Networks: An Introduction*. New York, NY, USA: Oxford University Press, Inc., 2010. ISBN 0199206651, 9780199206650.
- NEWMAN, M.; BARABASI, A.-L.; WATTS, D. J. *The structure and dynamics of networks*. [S.l.]: Princeton University Press, 2006. ISBN 978-0-691-11356-2.
- NEWMAN, M. E. J.; GIRVAN, M. Finding and evaluating community structure in networks. *Physical Review E*, American Physical Society, v. 69, n. 2, p. 026113+, Feb 2004.
- OLIVA, A.; TORRALBA, A.; CASTELHANO, M. S.; HENDERSON, J. M. Top-down control of visual attention in object detection. In: *Proc. of the IEEE Int'l Conference on Image Processing (ICIP '03)*. [S.l.: s.n.], 2003.

- PALETTA, L.; FRITZ, G.; SEIFERT, C. Q-learning of sequential attention for visual object recognition from informative local descriptors. In: *Proceedings of the 22Nd International Conference on Machine Learning*. New York, NY, USA: ACM, 2005. (ICML '05), p. 649–656. ISBN 1-59593-180-5.
- PALETTA, L.; PINZ, A. Active object recognition by view integration and reinforcement learning. *Robotics and Autonomous Systems*, v. 31, p. 71 – 86, 2000. ISSN 0921-8890.
- PALMER, S. E. *Vision science : photons to phenomenology*. [S.l.]: MIT Press, 1999. 810 p. Stephen E. Palmer; "A Bradford book."; Bibliography: Includes bibliographical references (p. [737]-769) and indexes. ISBN 0262161834 Thanks for using Barton, the MIT Libraries' catalog <http://>.
- PIÑOL, M.; SAPPA, A.; LOPEZ, A.; TOLEDO, R. Feature selection based on reinforcement learning for object recognition. In: *Adaptive Learning Agents Workshop*. [S.l.: s.n.], 2012. p. 33–39.
- PRASAD, D. K. Survey of the problem of object detection in real images. *International Journal of Image Processing (IJIP)*, v. 6, p. 441–466, 2012.
- PRICE, D. d. S. Network of scientific papers. *Science*, n. 149, p. 510–515, 1965.
- REDNER, S. How popular is your paper? an empirical study of the citation distribution. *The European Physical Journal B - Condensed Matter and Complex Systems*, v. 4, n. 2, p. 131–134, August 1998. ISSN 1434-6028.
- REYNOLDS, J.; CHELAZZI, L.; DESIMONE, R. Competitive mechanisms subserve attention in macaque areas v2 and v4. *Journal of Neuroscience*, v. 19, p. 1736–1753, 1999.
- RIBEIRO, B. A. N.; MUNTZ, R. A belief network model for ir. In: *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, 1996. p. 253–260. ISBN 0-89791-792-8.
- RIFAI, S.; MESNIL, G.; VINCENT, P.; MULLER, X.; BENGIO, Y.; DAUPHIN, Y.; GLOROT, X. Higher order contractive auto-encoder. In: *Proceedings of the 2011 European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part II*. Berlin, Heidelberg: Springer-Verlag, 2011. (ECML PKDD'11), p. 645–660. ISBN 978-3-642-23782-9.
- RODRIGUES, P. S.; GIRALDI, G. A.; ARAUJO, A. A. Using tsallis entropy into a bayesian network for cbir. In: IEEE (Ed.). *Proceedings of International International Conference on Image Processing (ICIP'05)*. Genova, Italy: [s.n.], 2005. v. 3, p. 1028–1031.
- ROSE; GUREWITZ; FOX. Statistical mechanics and phase transitions in clustering. *Physical Review Letters*, v. 65, n. 8, p. 945–948, maio 1990.
- ROWLEY, H. A.; BALUJA, S.; KANADE, T. Rotation invariant neural network-based face detection. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 1998. (CVPR '98), p. 38–. ISBN 0-8186-8497-6.
- RUSSELL, S. J.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. 2. ed. [S.l.]: Pearson Education, 2003. ISBN 0137903952.

- SARFRAZ, M.; HASSAN, M. ul; IQBAL, M. Object recognition using fourier descriptors and genetic algorithm. In: ABRAHAM, A.; MUDA, A. K.; HERMAN, N. S.; SHAMSUDDIN, S. M.; CHOO, Y.-H. (Ed.). *SoCPaR*. [S.l.]: IEEE, 2009. p. 318–323. ISBN 978-1-4244-5330-6.
- SAUNDERS, B. *Ivan Pavlov: Exploring the Mysteries of Behavior*. [S.l.]: Enslow Publishers, 2006. (Great minds of science). ISBN 9780766025066.
- SCHAEFFER, S. E. Graph clustering. *Computer Science Review*, v. 1, n. 1, p. 27 – 64, 2007. ISSN 1574-0137.
- SHEN-ORR, S. S.; MILO, R.; MANGAN, S.; ALON, U. Network motifs in the transcriptional regulation network of escherichia coli. *Nature genetics*, Nature Publishing Group, v. 31, n. 1, p. 64–68, May 2002. ISSN 1061-4036.
- SOLOMONOFF, R.; RAPOPORT, A. Connectivity of random nets. *Bulletin of Mathematical Biophysics*, n. 13, p. 107–117, 1951.
- SPORNS, O. Graph theory methods for the analysis of neural connectivity patterns. *Complex.*, v. 8, n. 1, p. 56–60, 2002. ISSN 1076-2787.
- STARZYK, J. A.; PRASAD, D. K. A computational model of machine consciousness. *International Journal of Machine Consciousness*, v. 03, n. 02, p. 255–281, 2011.
- STAUFFER, D.; AHARONY, A. Book. *Introduction to percolation theory / Dietrich Stauffer and Amnon Aharony*. 2nd ed.. ed. [S.l.]: Taylor and Francis, London, 1992. x, 181 p. : p. ISBN 0748400273.
- SUTTON, R. S.; BARTO, A. G. *Reinforcement Learning: An Introduction*. [S.l.]: The MIT Press, 1998. Hardcover. ISBN 0262193981.
- SZELISKI, R. *Computer Vision: Algorithms and Applications*. 1st. ed. New York, NY, USA: Springer-Verlag New York, Inc., 2010. ISBN 1848829345, 9781848829343.
- TAMURA, H.; MORI, S.; YAMAWAKI, T. Texture features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics*, v. 8, n. 6, 1978.
- TANG, J.; JIANG, B.; CHANG, C.-C.; LUO, B. Graph structure analysis based on complex network. *Digital Signal Processing*, v. 22, n. 5, p. 713–725, 2012.
- TEMEL, T. Finding number of clusters in single-step with similarity-based information-theoretic algorithm. *Electronics Letters*, Institution of Engineering and Technology, v. 50, p. 29–30(1), January 2014. ISSN 0013-5194.
- VAINA, L. (Ed.). *From Retina to Neocortex: Selected Papers of David Marr*. Boston, MA: Birkhauser, 1991.
- VASCONCELOS, N.; LIPPMAN, A. A bayesian framework for content-based indexing and retrieval. In: *Data Compression Conference*. [S.l.]: IEEE Computer Society, 1998. p. 580. ISBN 0-8186-8406-2.
- VINCENT, P.; LAROCHELLE, H.; BENGIO, Y.; MANZAGOL, P.-A. Extracting and composing robust features with denoising autoencoders. In: *Proceedings of the 25th International Conference on Machine Learning*. New York, NY, USA: ACM, 2008. (ICML '08), p. 1096–1103. ISBN 978-1-60558-205-4.

Wachs-Lopes, G. A.; FUKUMA, W.; RODRIGUES, P. S. Detecção de tipos de tomadas de vídeos de futebol utilizando a divergência de Kullback-Liebler. In: NEVES, L. A. P.; Vieira Neto, H.; GONZAGA, A. (Ed.). *Avanços em Visão Computacional*. 1. ed. Curitiba, PR: Omnipax, 2012. cap. 11, p. 201–218. ISBN 978-85-64619-09-8.

WALSH, T. Search in a small world. In: *IJCAI '99: Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999. p. 1172–1177. ISBN 1-5860-613-0.

WATTS, D. J. *Small Worlds: The Dynamics of Networks between Order and Randomness (Princeton Studies in Complexity)*. [S.l.]: Princeton University Press, 2004.

WATTS, D. J.; STROGATZ, S. H. Collective dynamics of /‘small-world’/ networks. *Nature*, v. 393, n. 6684, p. 440–442, 1998. ISSN 00280836.

WERNICK, M.; YANG, Y.; BRANKOV, J.; YOURGANOV, G.; STROTHER, S. Machine learning in medical imaging. *Signal Processing Magazine, IEEE*, IEEE, v. 27, n. 4, p. 25–38, jul. 2010. ISSN 1053-5888.

WOLFE, J. M.; CAVE, K. R.; FRANZEL, S. L. Guided search: An alternative to the feature integration model for visual search. *Journal of Experimental Psychology: Human Perception & Performance*, v. 15(3), p. 419–433, 1989.

XIAO, J.; HAYS, J.; EHINGER, K. A.; OLIVA, A.; TORRALBA, A. Sun database: Large-scale scene recognition from abbey to zoo. In: *CVPR*. [S.l.]: IEEE, 2010. p. 3485–3492.

YANG, J.; YU, K.; GONG, Y.; HUANG, T. Linear spatial pyramid matching using sparse coding for image classification. In: *in IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*. [S.l.: s.n.], 2009.

YANG, Y.; SHU, G.; SHAH, M. Semi-supervised learning of feature hierarchies for object detection in a video. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 1650–1657, 2013. ISSN 1063-6919.

YU, H.; LIU, Z.; WANG, G. An automatic method to determine the number of clusters using decision-theoretic rough set. *International Journal of Approximate Reasoning*, v. 55, n. 1, Part 2, p. 101 – 115, 2014. ISSN 0888-613X. Special issue on Decision-Theoretic Rough Sets.

ZHANG, D.; LU, G. Review of shape representation and description techniques. *Pattern Recognition*, v. 37, p. 1–19, 1 2004.

ZHANG, Y.; DONG, L.; LI, S.; LI, J. A complex network-based approach for interest point detection in images. In: *2012 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. [S.l.]: IEEE Press, 2012. p. 1–4. ISBN 978-1-4673-0293-7.