

**UNIVERSIDADE FEDERAL DE PELOTAS**  
Centro de Desenvolvimento Tecnológico  
Programa de Pós-Graduação em Computação



Dissertação

**Uma extensão do algoritmo Fuzzy C-Means aplicado ao reconhecimento de spots em imagens oriundas de Eletroforese Bidimensional**

**Geancarlo Saldanha Maydana**

Pelotas, 2016

**Geancarlo Saldanha Maydana**

**Uma extensão do algoritmo Fuzzy C-Means aplicado ao reconhecimento de spots em imagens oriundas de Eletroforese Bidimensional**

Dissertação apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Mestre em Ciência da Computação

Orientador: Prof. Dr. Marilton Sanchotene de Aguiar

Pelotas, 2016

**Insira AQUI a ficha catalográfica  
(solicitada na página da biblioteca)**

**Insira AQUI a folha de aprovação  
(será entregue após a banca)**

## RESUMO

MAYDANA, Geancarlo Saldanha. **Uma extensão do algoritmo Fuzzy C-Means aplicado ao reconhecimento de spots em imagens oriundas de Eletroforese Bidimensional.** 2016. 84 f. Dissertação (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2016.

A proteômica pode ser vista como uma metodologia de seleção da biologia molecular, a qual tem como objetivo documentar a distribuição geral de proteínas da célula, identificar e caracterizar proteínas, e principalmente, elucidar as suas associações e funções.

Atualmente, uma das principais técnicas usadas na proteômica é a eletroforese bidimensional (2D), baseada na separação e na migração das moléculas carregadas, numa solução, em função da aplicação de um campo elétrico. Estas proteínas podem ser detectadas por uma variedade de reagentes de revelação, observando-se um perfil bidimensional de pontos (do inglês, *spots*).

As imagens de eletroforese bidimensional normalmente contém ruídos, tais como, partículas de poeira e até mesmo rachaduras no gel, e isso pode interferir no resultado final da análise de reconhecimento dos *spots*.

Neste contexto, este trabalho explora as características intrínsecas da lógica *fuzzy* para o processamento de dados, e provê um modelo a fim de encontrar os *spots* nas imagens oriundas de eletroforese bidimensional. O modelo proposto utiliza clusterização *fuzzy* e faz uso de informações contidas nos *pixels* da imagem para fazer a classificação dos pontos na imagem e encontrar os *spots*.

Para avaliação do modelo proposto fez-se uso de imagens de eletroforese bidimensional da base SWISS-2DPAGE contendo *spots* já verificados em laboratório. Foram utilizadas 29 imagens nos testes do modelo, com o melhor resultado de *recall* chegando a 93,30% dos *spots* encontrados. Também foram realizadas comparações do modelo proposto, de aprendizado não-supervisionado, com técnicas de aprendizado supervisionado (aprendizagem de máquina) em desenvolvimento no grupo de pesquisa.

**Palavras-chave:** Eletroforese, fuzzy, clusterização, spots.

## ABSTRACT

MAYDANA, Geancarlo Saldanha. **An Extension of Fuzzy C-Means applied to Spot Recognition in Images of Two-dimensional Electrophoresis.** 2016. 84 f. Dissertação (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2016.

Proteomics is a methodology widely used, especially by molecular biology, which main goals consist of documenting the general distribution of proteins in a cell, identifying and characterizing individual proteins of interest and, mainly, elucidating associations and functions of proteins.

Currently, one of the main techniques used in proteomics is Two-dimensional Electrophoresis (2D), which is based on separation and migration of molecules, placed in a solution under the influence of the application of an electric field. These proteins can be detected by a variety of revelation reagents, revealing a two-dimensional profile of spots.

At the end of the process, the gel is scanned and the resulting image can be processed. Two-dimensional electrophoresis images often contain noise, such as dust particles, and even cracks in the gel, and this can interfere in the the analysis results.

In this context, this work explores the application of fuzzy logic in image processing, and provides a model in order to find spots in Two-dimensional electrophoresis images. The proposed model uses fuzzy clustering and it only makes use of the pixels information to do the classification of the pixels and find the spots.

The evaluation of the proposed model was made using two-dimensional electrophoresis images of SWISS-2DPAGE database. These images contain spots already checked in the laboratory. We used 29 images in the model tests, with the best result of recall reaching 93,30% of the spots found. Also, comparisons were made of the proposed model, of unsupervised learning, with supervised learning techniques (machine learning) in development in the research group.

**Keywords:** eletrophoresis, fuzzy, clustering, spots.

## LISTA DE FIGURAS

Figura 1	Imagem monocromática (MARQUES FILHO; VIEIRA NETO, 1999).	13
Figura 2	Tipos de conectividade: conectividade B8 (a) e conectividade B4 (b) (ESQUEF, 2002).	14
Figura 3	Exemplos de histograma da imagem (ESQUEF, 2002).	16
Figura 4	Exemplo de equalização de histograma. (a) é a imagem original e seu histograma e (b) é a imagem equalizada e seu histograma.	17
Figura 5	Exemplo de aplicação de filtros de passa-baixa (b) e passa-alta (c) na imagem original (a).	19
Figura 6	Polígono representado por suas bordas (ANGEL E.; SHREINER, 2014).	20
Figura 7	Algoritmo Flood Fill (ANGEL E.; SHREINER, 2014).	20
Figura 8	Conjuntos <i>crisp</i> (CHI; YAN; PHAM, 1996).	21
Figura 9	Conjuntos <i>fuzzy</i> (CHI; YAN; PHAM, 1996).	22
Figura 10	Conjunto <i>fuzzy</i> normal (CHI; YAN; PHAM, 1996).	22
Figura 11	Conjunto <i>fuzzy</i> subnormal (CHI; YAN; PHAM, 1996).	23
Figura 12	Conjunto <i>fuzzy</i> convexo (CHI; YAN; PHAM, 1996).	23
Figura 13	Conjunto <i>fuzzy</i> não-convexo (CHI; YAN; PHAM, 1996).	24
Figura 14	Intersecção entre os conjuntos <i>fuzzy</i> A e B (área sombreada) (CHI; YAN; PHAM, 1996).	24
Figura 15	União entre os conjuntos <i>fuzzy</i> A e B (área sombreada) (CHI; YAN; PHAM, 1996).	25
Figura 16	Conjunto <i>fuzzy</i> A e seu complemento (CHI; YAN; PHAM, 1996).	25
Figura 17	(a) Função de pertinência triangular; (b) Função de pertinência trapezoidal; (c) Função de pertinência em forma de S; e (d) Função de pertinência em forma de sino. (CHI; YAN; PHAM, 1996)	31
Figura 18	Pico e larguras em uma função de pertinência triangular assimétrica (CHI; YAN; PHAM, 1996).	32
Figura 19	<i>Crosspoint</i> e nível de <i>crosspoint</i> de uma função de pertinência triangular (CHI; YAN; PHAM, 1996).	32
Figura 20	Funções de pertinência triangular (a) e trapezoidal (b) com cinco rótulos linguísticos (CHI; YAN; PHAM, 1996).	33
Figura 21	<i>Clusters</i> resultantes da aplicação de duas medidas de semelhança, (a) diferença de ângulo, e (b) distância euclidiana. $P_1$ , $P_2$ , e $P_3$ são os centróides (CHI; YAN; PHAM, 1996).	34
Figura 22	Arquitetura do AVQ (CHI; YAN; PHAM, 1996).	35

Figura 23	Mapa Auto-Organizado de Kohonen bidimensional de uma camada ( $m$ : número de nodos no mapa; $(x_1, x_2, \dots, x_n)$ : as entradas; e $W_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ : o vetor de pesos do nodo $i$ ) (CHI; YAN; PHAM, 1996). . . . .	36
Figura 24	Um mapa auto-organizado de $15 \times 15$ obtido de um conjunto de 10.426 amostras de dígitos escritos a mão (CHI; YAN; PHAM, 1996). . . . .	38
Figura 25	Demonstração do processo de combinação de funções de pertinência vizinhas (CHI; YAN; PHAM, 1996). . . . .	39
Figura 26	Uma imagem de documento e seu histograma de níveis de cinza (CHI; YAN; PHAM, 1996). . . . .	45
Figura 27	Resultado da segmentação obtida usando o algoritmo C-Means (CHI; YAN; PHAM, 1996). . . . .	46
Figura 28	Valores de pertinência <i>fuzzy</i> mostrados como uma imagem e o histograma de níveis de cinza da imagem (CHI; YAN; PHAM, 1996). . . . .	46
Figura 29	Resultado da segmentação obtida a partir da Figura 28 com um valor de limiar de 91 (CHI; YAN; PHAM, 1996). . . . .	47
Figura 30	Resultado da segmentação obtida a partir da Figura 28 com um valor de limiar de 42 (CHI; YAN; PHAM, 1996). . . . .	48
Figura 31	Uma imagem de ressonância magnética de tecidos cerebrais (esquerda) e o resultado de sua segmentação (direita)(CHI; YAN; PHAM, 1996). . . . .	48
Figura 32	Uma imagem de ressonância magnética de um cérebro (esquerda) e o resultado da segmentação (direita). A área do tumor na imagem foi isolada dos tecidos. (CHI; YAN; PHAM, 1996) . . . . .	49
Figura 33	Imagens de géis oriundas de eletroforese bidimensional. . . . .	52
Figura 34	Imagens de géis com bastante ruído. . . . .	53
Figura 35	Representação dos <i>spots</i> indicados pelo SWISS-2DPAGE. . . . .	56
Figura 36	Fluxograma do modelo <i>fuzzy</i> proposto. . . . .	58
Figura 37	Pseudocódigo da etapa de clusterização. . . . .	58
Figura 38	Pseudocódigo da etapa de classificação. . . . .	59
Figura 39	Pseudocódigo da etapa de contagem dos <i>spots</i> . . . . .	60
Figura 40	<i>Spots</i> encontrados utilizando o algoritmo de contagem. . . . .	60
Figura 41	Pseudocódigo do modelo <i>fuzzy</i> para detecção dos <i>spots</i> . . . . .	61
Figura 42	Particionamento da imagem. . . . .	61
Figura 43	Fluxograma do modelo <i>fuzzy</i> proposto para a imagem particionada em quatro quadrantes. . . . .	62
Figura 44	Clusterização resultante da Figura 42(a) . . . . .	66
Figura 45	Recorte da Figura 44. . . . .	67
Figura 46	Uma visão gráfica da Tabela 5. . . . .	71
Figura 47	Uma visão gráfica da Tabela 7. . . . .	74
Figura 48	Uma visão gráfica da Tabela 8. . . . .	76

## LISTA DE TABELAS

Tabela 1	Informações das imagens disponibilizadas pelo SWISS-2DPAGE. . . . .	57
Tabela 2	Testes executados em uma imagem fornecida pelo Laboratório de Bioinformática da Universidade Federal de Pelotas. . . . .	65
Tabela 3	Resultado da aplicação do Fuzzy C-Means nas imagens do SWISS-2DPAGE. . . . .	68
Tabela 4	Resultado da execução do Fuzzy C-Means nas imagens do SWISS-2DPAGE, utilizando a clusterização por partes. . . . .	69
Tabela 5	Comparação de <i>recall</i> nos testes do Fuzzy C-Means usando a imagem completa e dividida. . . . .	70
Tabela 6	Resultado da execução do K-Means nas imagens do SWISS-2DPAGE	72
Tabela 7	Valores de <i>Recall</i> apresentados pelas técnicas. . . . .	73
Tabela 8	Valores de $F_1$ apresentados pelas técnicas. . . . .	75
Tabela 9	Valores de $p$ no Teste T pareado para o <i>Recall</i> . . . . .	76
Tabela 10	Valores de $p$ no Teste T pareado para o $F_1$ . . . . .	76

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	10
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	12
<b>2.1</b>	<b>Processamento Digital de Imagens</b>	12
2.1.1	Fundamentos de Imagens Digitais	13
2.1.2	Histograma	14
2.1.3	Segmentação	15
2.1.4	Realce de Imagens	15
2.1.5	Preenchimento de áreas	18
<b>2.2</b>	<b>Lógica fuzzy</b>	18
2.2.1	Conjuntos Fuzzy	21
2.2.2	Funções de Pertinência	28
2.2.3	Modelos Fuzzy para o Processamento de Imagens	39
<b>2.3</b>	<b>Agrupamento</b>	41
2.3.1	Algoritmo C-Means	41
2.3.2	Algoritmo Fuzzy C-Means	43
2.3.3	Comparação entre os algoritmos C-Means e Fuzzy C-Means	45
2.3.4	Aplicação	47
<b>2.4</b>	<b>Eletroforese Bidimensional</b>	49
2.4.1	Funcionamento	50
<b>3</b>	<b>CLUSTERIZAÇÃO FUZZY APLICADO AO RECONHECIMENTO DE SPOTS</b>	54
<b>3.1</b>	<b>Trabalhos relacionados</b>	54
<b>3.2</b>	<b>Base de imagens</b>	55
<b>3.3</b>	<b>Modelo de Clusterização Fuzzy</b>	58
<b>4</b>	<b>RESULTADOS</b>	63
<b>4.1</b>	<b>Métricas</b>	63
<b>4.2</b>	<b>Definição dos Parâmetros</b>	64
<b>4.3</b>	<b>Análise do Fuzzy C-Means com a Imagem Inteira</b>	66
<b>4.4</b>	<b>Análise do Fuzzy C-Means dividindo-se a Imagem em Quadrantes</b>	69
<b>4.5</b>	<b>K-Means</b>	71
<b>4.6</b>	<b>Análise Comparativa</b>	72
<b>5</b>	<b>CONCLUSÕES</b>	77
	<b>REFERÊNCIAS</b>	79

# 1 INTRODUÇÃO

O proteoma indica as proteínas expressas em um organismo ou tecido. Enquanto o genoma representa a soma de todos os genes de um indivíduo, o proteoma não é uma característica fixa. O proteoma muda de acordo com o estado de desenvolvimento do tecido ou mesmo sob as condições nas quais o indivíduo se encontra. Portanto, investigar diretamente os produtos dos genes é uma forma de estudar doenças e qualquer problema biológico complexo.

A proteômica pode ser vista como uma metodologia de seleção da biologia molecular, a qual tem como objetivo documentar a distribuição geral de proteínas da célula, identificar e caracterizar proteínas individuais de interesse e principalmente elucidar as suas associações e funções. Além disso, a proteômica é uma abordagem para identificar, quantificar e estudar as modificações pós-traducionais das proteínas em uma célula, tecido ou, mesmo, organismos (CIERO; BELLATO, 2002).

Por exemplo, para entender no nível molecular como uma célula funciona em um indivíduo doente e em um sadio, é preciso ter conhecimento das proteínas e de outros componentes celulares que estão presentes, como eles interagem e o resultado de suas interações (CIERO; BELLATO, 2002). A análise ao nível de proteína é muito necessária. O estudo dos genes através do sequenciamento de DNA não pode adequadamente prever a estrutura dinâmica das proteínas. É justamente ao nível das proteínas que muitos dos processos de uma célula ocorrem. Por exemplo, é onde processos patológicos acontecem e aonde muitas das drogas atuam.

A eletroforese bidimensional é uma técnica de separação de moléculas baseada na migração delas em uma solução, quando aplicado um campo elétrico. É uma das principais técnicas usadas na proteômica. Em um experimento de eletroforese bidimensional em gel, as proteínas (ou outras moléculas) são separadas em duas dimensões, de modo que todas as proteínas/moléculas se espalhem por todo o gel. Esse gel é submetido a um reagente de revelação, que faz a coloração do gel. A partir daí se pode observar um perfil bidimensional de pontos, sendo que em cada ponto há múltiplas cópias de uma proteína. Por fim, o gel é escaneado e a imagem resultante pode ser processada (ROCHA; SILVA; SA, 2005).

Os géis de eletroforese podem conter partículas de poeira, e até mesmo rachaduras, e isto pode interferir na imagem oriunda, gerando ruídos. Isto afeta a análise do gel, dificultando a detecção dos *spots*.

A lógica *fuzzy* surge como uma alternativa para contornar essa dificuldade. A lógica “nebulosa”, ou *fuzzy*, é uma lógica multivalorada capaz de representar as informações vagas e incertas da linguagem natural. A capacidade de tomar conclusões a partir de premissas incertas é umas das principais aplicações da lógica *fuzzy*. TANSCHKEIT; GOMIDE; TEIXEIRA (2007) descreve a lógica *fuzzy* como uma ferramenta capaz de capturar informações vagas, em geral descritas em uma linguagem natural, e convertê-las para um formato numérico. Um dos campos em que a lógica *fuzzy* vem sendo aplicada é no processamento de imagens digitais. Segundo CHI; YAN; PHAM (1996), o interesse dos pesquisadores vem dos fatos que: (1) regras *fuzzy* são naturalmente efetivas para qualquer sistema de cognição humana, como a compreensão de imagens e reconhecimento de padrões, e (2) a teoria dos conjuntos *fuzzy* provê uma boa plataforma para lidar com informações ruidosas e imprecisas que são frequentemente encontradas.

Dadas as dificuldades da análise dos géis de eletroforese e as características intrínsecas da lógica *fuzzy*, os objetivos do trabalho são:

- propor um modelo *Fuzzy* a fim de detectar os *spots* nas imagens de eletroforese bidimensional.
- avaliar o modelo proposto em comparação a modelos *crisp* (não-*fuzzy*).

Para melhor compreensão, este trabalho está organizado da seguinte forma:

No Capítulo 2, é apresentado o referencial teórico deste trabalho. São abordados os conceitos referentes ao processamento digital de imagens, lógica *fuzzy*, agrupamento e eletroforese bidimensional.

O Capítulo 3 traz informações sobre a base de imagens utilizada no trabalho e também a formalização do modelo *fuzzy* utilizado para fazer a detecção dos *spots*.

O Capítulo 4 apresenta os resultados da aplicação do modelo em imagens reais de eletroforese bidimensional.

No Capítulo 5 são apresentadas as conclusões e os trabalhos futuros.

## 2 REFERENCIAL TEÓRICO

### 2.1 Processamento Digital de Imagens

As primeiras aplicações em imagens remontam do começo do século XX, onde buscavam-se formas de aprimorar a qualidade da impressão das imagens transmitidas pelo sistema Bartlane de transmissão de imagens através de cabos submarinos que ligavam Londres e Nova Iorque. Mas o grande impulso para o Processamento Digital de Imagens (PDI) foi na década de 60, com o advento dos primeiros computadores de grande porte e o início do programa espacial norte-americano. O uso de técnicas computacionais para melhoria de imagens teve início no projeto *Jet Propulsion Laboratory*<sup>1</sup> em 1964, quando imagens da Lua eram transmitidas por uma sonda Ranger eram processadas por um computador para corrigir as distorções inerentes à câmera acoplada à sonda.

De 1964 até os dias atuais, a área de processamento de imagens apresentou crescimento expressivo e suas aplicações permeiam quase todos os ramos da atividade humana. Por exemplo, na medicina é rotineiro que imagens sejam utilizadas no diagnóstico de exames, e os avanços no processamento de imagens vem permitindo tanto o desenvolvimento de novos equipamentos quanto a maior facilidade de interpretação de imagens. Na Biologia, a capacidade de processar imagens obtidas em microscópios, por exemplo, facilita a execução de tarefas laboratoriais com alto grau de precisão. Imagens captadas por satélites e interpretadas automaticamente auxiliam nas áreas de Geografia, Sensoriamento Remoto, Geoprocessamento e Meteorologia, dentre outras. Técnicas de restauração de imagens auxiliam arqueologistas na dura tarefa de recuperar fotos borradas de artefatos raros. Ainda outras áreas como Astronomia e Direito vem sendo beneficiadas com os avanços no processamento de imagens e visão computacional. (MARQUES FILHO; VIEIRA NETO, 1999)

---

<sup>1</sup><http://www.jpl.nasa.gov/>

## 2.1.1 Fundamentos de Imagens Digitais

### 2.1.1.1 Propriedades

Uma imagem monocromática pode ser matematicamente através de uma função  $f(x, y)$  da intensidade luminosa, sendo seu valor, em qualquer ponto de coordenada  $(x, y)$  proporcional ao nível de cinza da imagem naquele ponto. A Figura 1 mostra uma imagem monocromática.



Figura 1: Imagem monocromática (MARQUES FILHO; VIEIRA NETO, 1999).

No caso de uma imagem que possui informações em intervalos ou bandas distintas de frequência, é necessário uma função  $f(x, y)$  para cada banda. É este o caso de imagens no padrão RGB, que são formadas pela informação de três cores: o vermelho (R - Red), verde (G - Green) e azul (B - Blue) (MARQUES FILHO; VIEIRA NETO, 1999).

Para que uma imagem seja processada por alguma técnica, é imprescindível representar a sua informação em formato computacional adequado. É usual utilizar uma matriz de números inteiros não-negativos, cujos elementos da matriz representam o valor do *pixel* correspondente na imagem. Sendo assim, O *pixel* é o elemento básico da imagem (MASCARENHAS N.; VELASCO, 1989).

Um *pixel* não apresenta as mesmas propriedades em todas as direções, e isso faz com que um *pixel* possa ter quatro vizinhos de borda e quatro vizinhos de diagonal, como pode ser observado na Figura 2. Esta organização faz com que surjam dois problemas. O primeiro consiste em determinar qual conectividade será utilizada: vizinhos de borda (B4) ou vizinhos de borda e diagonal (B8). O segundo problema está relacionado com as distâncias entre um ponto e seus vizinhos. A distância do ponto central aos vizinhos de borda é 1 e  $\sqrt{2}$  para os vizinhos de diagonal.

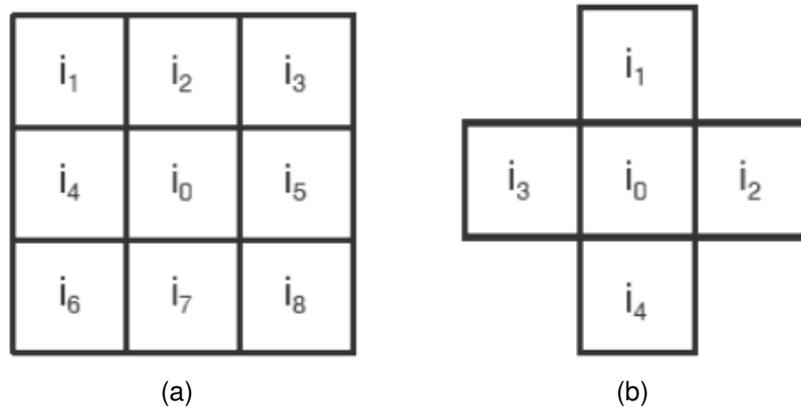


Figura 2: Tipos de conectividade: conectividade B8 (a) e conectividade B4 (b) (ESQUEF, 2002).

### 2.1.1.2 Aquisição e digitalização de imagens

Para converter uma cena real em uma imagem digitalizada, são necessárias duas etapas: a aquisição da imagem e sua digitalização.

O processo de aquisição da imagem necessita de um dispositivo físico, sensível ao espectro de energia eletromagnética. Este dispositivo deve produzir em sua saída um sinal elétrico proporcional ao nível de energia percebido por ele (GONZALEZ; WOODS, 1992).

A partir disso, para que seja feita a digitalização da imagem, necessita-se um dispositivo, chamado digitalizador, que converte o sinal elétrico analógico produzido na saída do sensor em um sinal digital (GONZALEZ; WOODS, 1992).

### 2.1.1.3 Pré-processamento

A imagem resultante do processo anterior pode apresentar diversas imperfeições, tais como: presença de *pixels* ruidosos, contraste e/ou brilho inadequado. A função da etapa de pré-processamento é aprimorar a qualidade da imagem de forma a aumentar as chances de sucesso nos processos seguintes (MARQUES FILHO; VIEIRA NETO, 1999).

As operações efetuadas nesta etapa são ditas de baixo nível porque trabalham diretamente com os valores de intensidade dos *pixels*. A imagem resultante desta etapa é uma imagem digitalizada de melhor qualidade que a original (MARQUES FILHO; VIEIRA NETO, 1999).

## 2.1.2 Histograma

O histograma de uma imagem é simplesmente um conjunto de números indicando o percentual de *pixels* naquela imagem que apresentam um determinado nível de cinza. Estes valores são normalmente representados por um gráfico de barras que fornece para cada nível de cinza o número (ou percentual) de *pixels* correspondente

na imagem (MARQUES FILHO; VIEIRA NETO, 1999).

O histograma de uma imagem pode ser definido através de uma função discreta, como pode ser observado na Equação 1, onde o argumento  $k$  representa os níveis de luminância discretos,  $n_k$  representa o número de *pixels* na imagem com intensidade  $k$  e  $n$  é o número total de *pixels* da imagem, ou seja,  $n = M \times N$  (ESQUEF, 2002).

$$p(k) = \frac{n_k}{n} \quad (1)$$

O histograma fornece, assim, uma visão estatística sobre a distribuição dos *pixels*, o contraste da imagem e os níveis de iluminação. Estes dados são úteis para obter-se melhorias na imagem e são utilizados em técnicas de compressão e segmentação de imagens (GONZALEZ; WOODS, 1992).

A Figura 4 apresenta duas imagens e seus histogramas.

### 2.1.3 Segmentação

A tarefa básica da etapa de segmentação é a de dividir uma imagem em suas unidades significativas, ou seja, nos objetos de interesse que a compõem (MARQUES FILHO; VIEIRA NETO, 1999). A segmentação é a etapa mais crítica do tratamento da informação. Nesta etapa são definidas as regiões de interesse na imagem para processamento e análise posteriores. Quaisquer erros nesta etapa serão sentidas nas demais, produzindo resultados indesejados e contribuindo de forma negativa na eficiência do processamento (ESQUEF, 2002).

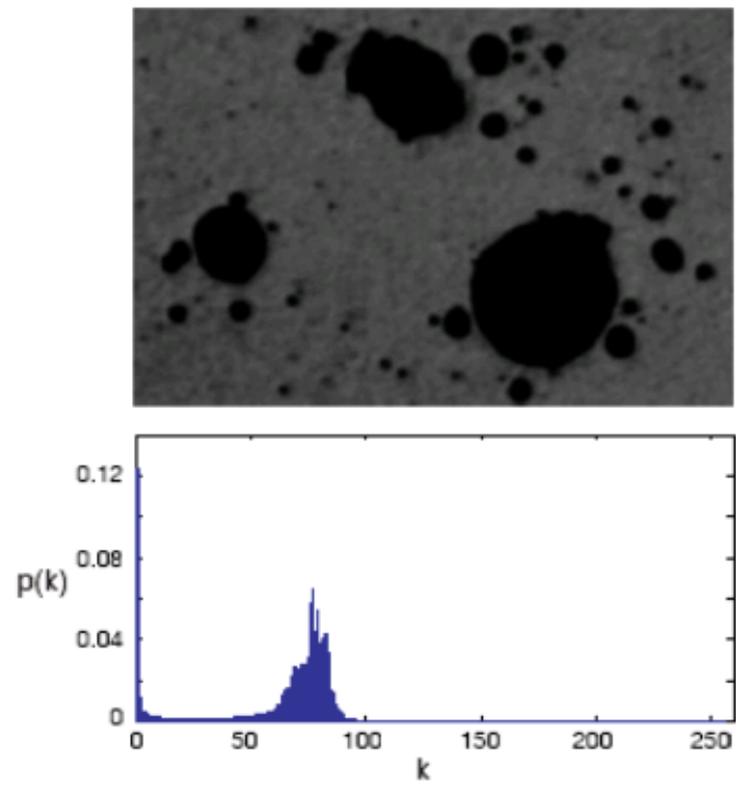
### 2.1.4 Realce de Imagens

O realce tem por objetivo acentuar aspectos da imagem para posterior análise ou visualização. Nenhuma informação é adicionada à imagem, mas sim acentuadas determinadas características (GONZALEZ; WOODS, 1992).

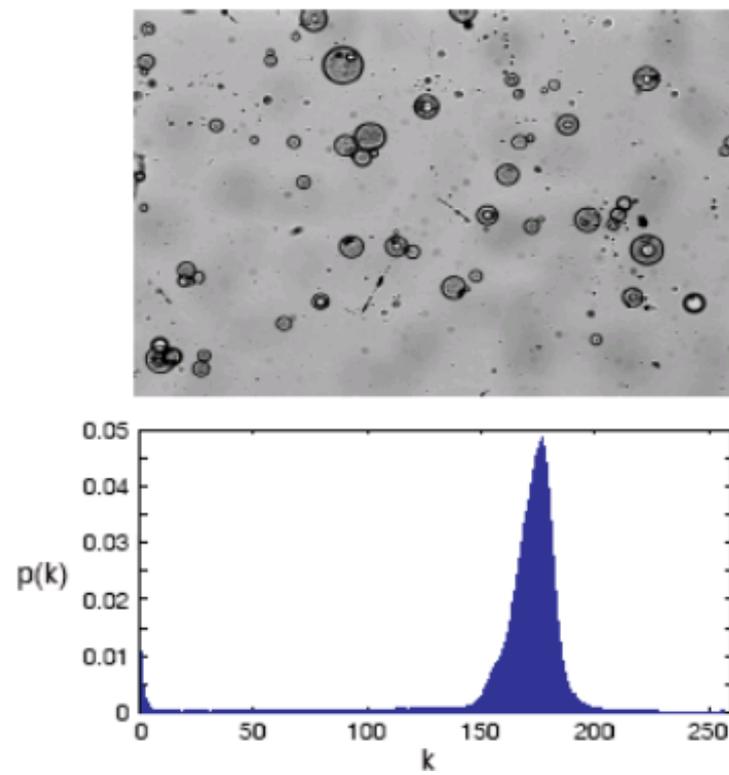
As técnicas de realce podem ser classificadas em duas categorias: métodos ponto a ponto e métodos de vizinhança. Métodos ponto a ponto modificam os níveis de cinza de uma imagem preservando os contornos. Um exemplo deste método é a equalização de histograma (GONZALEZ; WOODS, 1992).

Como visto anteriormente, o histograma é uma tabela que indica, para cada nível de cinza, a quantidade de *pixels* existentes na imagem, apresentando a frequência destes. Equalizar um histograma, é gerar uniformidade de densidade para os níveis de cinza da imagem, o que acaba por resultar em maior contraste e nitidez. A Figura 4 mostra a imagem original e seu histograma (a) e a imagem equalizada com seu histograma (b).

Métodos de vizinhança são utilizados em regiões da imagem a fim de enfatizar determinados aspectos. Uma das aplicações é a eliminação de ruídos através de



(a)



(b)

Figura 3: Exemplos de histograma da imagem (ESQUEF, 2002).

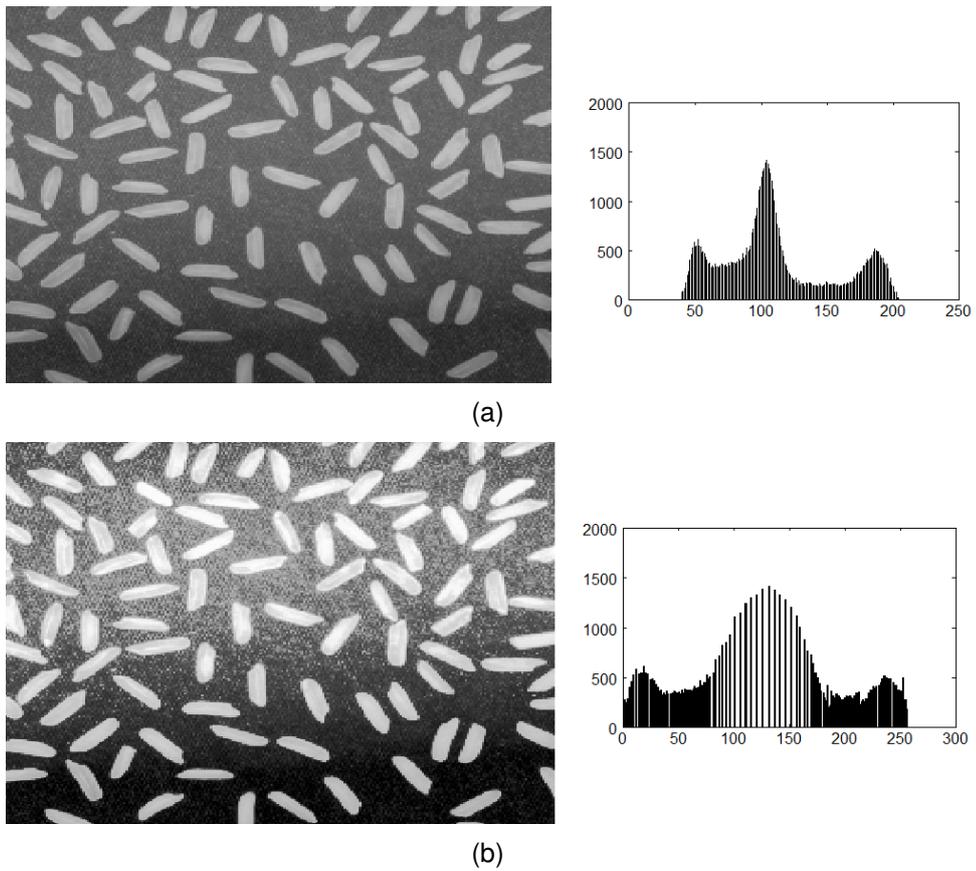


Figura 4: Exemplo de equalização de histograma. (a) é a imagem original e seu histograma e (b) é a imagem equalizada e seu histograma.

suavização. Um exemplo é a técnica de filtragem. Nesta, não somente o nível de cinza original é utilizado, mas os valores de cinza da vizinhança são considerados também (GONZALEZ; WOODS, 1992).

Existem duas classes importantes amplamente utilizadas: filtros de passa-baixa e filtros de passa-alta. Em um filtro de passa-baixa têm-se valores próximos a zero para as altas frequências. Com isso, a imagem fica “suavizada”. Um dos efeitos deste filtro é a eliminação de ruídos, porém torna a imagem menos nítida e definida (JAIN, 1989). A Figura (b) ilustra um filtro passa-baixa chamado Filtro da Média.

Um filtro passa-alta é aquele que tem valores próximos a zero para as baixas frequências. Assim, a imagem fica mais nítida. Entretanto, este filtro enfatiza o ruído presente na imagem (JAIN, 1989). A Figura (c) ilustra o filtro passa-alta Laplaciano + Original.

### 2.1.5 Preenchimento de áreas

Uma técnica que pode ser utilizada no preenchimento de áreas é o algoritmo *Flood Fill*. Através desse algoritmo é possível preencher um polígono sabendo suas bordas. Supondo que se tenha somente duas cores, a cor de fundo (branco) e o desenho em primeiro plano, com outra cor. A Figura 6 apresenta um polígono com suas bordas.

Dado um ponto inicial  $(x, y)$ , chamado de *seed point*, é possível colori-lo e, recursivamente, colorir os pontos vizinhos que sejam da mesma cor. Ou seja, colorir apenas os vizinhos que não fazem da borda.

O algoritmo pode ser expresso em pseudocódigo, na Figura 7, assumindo que temos uma função **read\_pixel** que retorna a cor do *pixel* e que a função **write\_pixel** colore um *pixel* (ANGEL E.; SHREINER, 2014):

Pode-se notar na Figura 7 que o algoritmo utiliza a vizinhança B4 (veja Figura 2 (b)), mas pode ser facilmente adaptado para a vizinhança B8 (Figura 2 (a)).

## 2.2 Lógica fuzzy

A lógica “nebulosa”, ou *fuzzy*, é uma lógica multivalorada capaz de representar as informações vagas e incertas da linguagem natural. As primeiras noções desta lógica foram dadas pelo polonês Jan Lukasiewicz, e esta foi formalizada em ZADEH (1965). A principal diferença entre a lógica clássica e a lógica *fuzzy* está na valoração do grau de pertinência; enquanto na lógica clássica os valores de pertinência são 0 e 1, na lógica *fuzzy* os valores variam entre 0 (indicando que o elemento não pertence ao conjunto) e 1 (indicando pertinência total ao conjunto).

A capacidade de tomar conclusões a partir de premissas incertas é uma das principais aplicações da lógica *fuzzy*. TANSCHKEIT; GOMIDE; TEIXEIRA (2007) descreve a lógica *fuzzy* como uma ferramenta capaz de capturar informações vagas, em geral



(a)



(b)



(c)

Figura 5: Exemplo de aplicação de filtros de passa-baixa (b) e passa-alta (c) na imagem original (a).

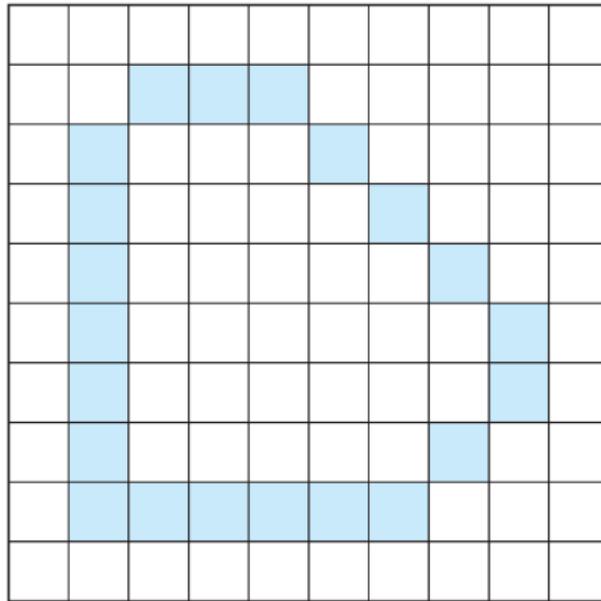


Figura 6: Polígono representado por suas bordas (ANGEL E.; SHREINER, 2014).

```

flood_fill(int x, int y)
{
    if (read_pixel(x,y) == WHITE)
    {
        write_pixel(x,y,BLACK);
        flood_fill(x-1,y);
        flood_fill(x+1,y);
        flood_fill(x,y-1);
        flood_fill(x,y+1);
    }
}

```

Figura 7: Algoritmo Flood Fill (ANGEL E.; SHREINER, 2014).

descritas em uma linguagem natural, e convertê-las para um formato numérico.

Pode-se ainda imaginar uma série de conceitos que possuem a característica de incerteza, ou vagueza. Por exemplo, conjunto das pessoas altas e o conjunto das ruas grandes de uma cidade, o conjunto dos problemas de uma determinada localidade (BARROS, 1997).

Um dos campos em que a lógica *fuzzy* vem sendo aplicada é no processamento de imagens digitais. Segundo CHI; YAN; PHAM (1996), o interesse dos pesquisadores vem dos fatos que: (1) regras *fuzzy* são naturalmente efetivas para qualquer sistema de cognição humana, como a compreensão de imagens e reconhecimento de padrões, e (2) a teoria dos conjuntos *fuzzy* provê uma boa plataforma para lidar com informações ruidosas e imprecisas que são frequentemente encontradas na nossa vida diária.

### 2.2.1 Conjuntos Fuzzy

Tratando-se de conjunto no sentido *clássico*, a relação de pertinência entre um elemento e um conjunto deve estar bastante clara. Dado um conjunto  $A$  e um elemento  $x$  tem-se que  $x$  *pertence* a  $A$  ( $x \in A$ ) ou  $x$  *não pertence* a  $A$  ( $x \notin A$ ). A noção de conjunto *fuzzy* dada por Zadeh (ZADEH, 1965) generaliza a ideia de conjunto.

As Figuras 8 e 9 mostram os diferentes conceitos de descrever “verdadeiro” e “falso” em conjuntos *crisp* e *fuzzy*, respectivamente. O que realmente importa em um conjunto *fuzzy* é o grau de pertinência atribuído a cada membro. Esta percepção de imprecisão acaba por ser natural e mais atraente para os conceitos cotidianos. Porém, é importante deixar claro que a teoria dos conjuntos *fuzzy* não é uma teoria que permite imprecisão, mas sim que consegue lidar com informações imprecisas em situações complexas (CHI; YAN; PHAM, 1996).

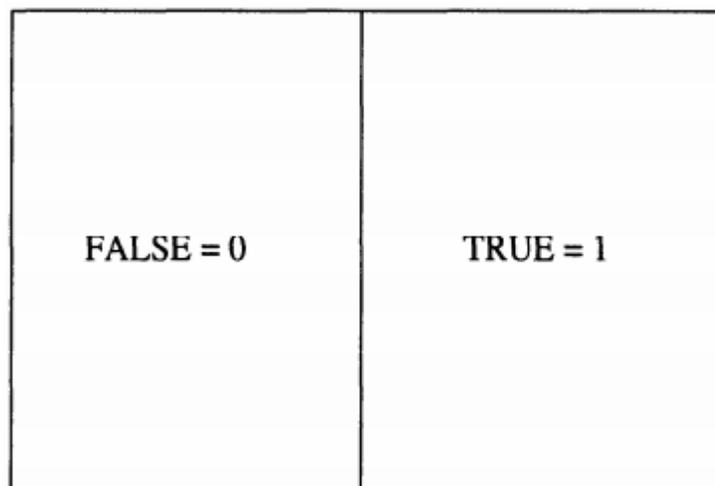


Figura 8: Conjuntos *crisp* (CHI; YAN; PHAM, 1996).

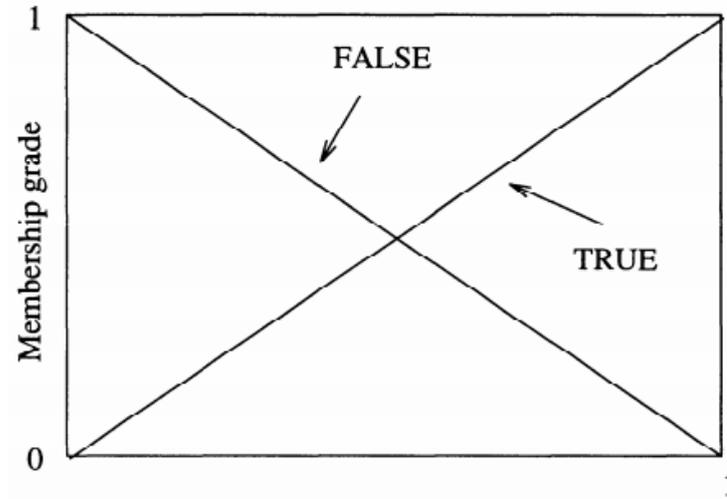


Figura 9: Conjuntos *fuzzy* (CHI; YAN; PHAM, 1996).

### 2.2.1.1 Propriedades dos Conjuntos Fuzzy

Abaixo as propriedades dos conjuntos *fuzzy* dadas em (CHI; YAN; PHAM, 1996):

- *Normalidade*: Um conjunto *fuzzy* é dito normal se o maior valor de sua função de pertinência é igual a 1. Do contrário, o conjunto é subnormal. As Figuras 10 e 11 mostram conjuntos *fuzzy* normal e subnormal, respectivamente.

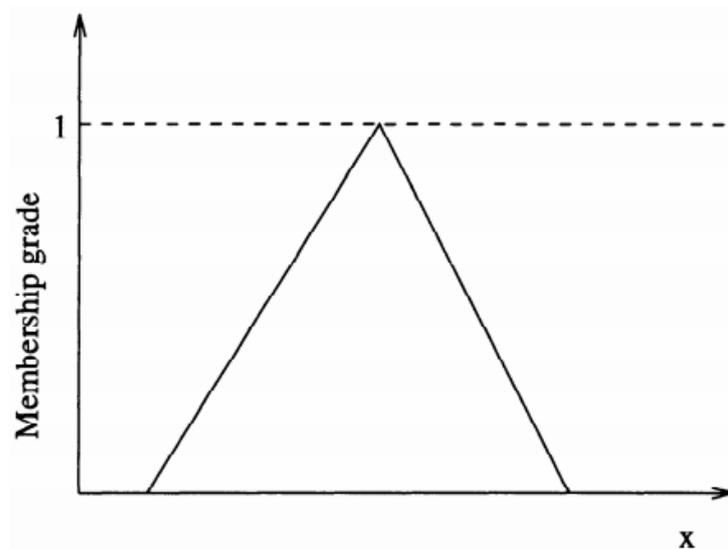


Figura 10: Conjunto *fuzzy* normal (CHI; YAN; PHAM, 1996).

- *Convexidade*: Um conjunto *fuzzy* é convexo se, e somente se, os conjuntos  $A_\alpha$  definidos como

$$A_\alpha = \{x/\mu_A(x) \geq \alpha\} \quad (2)$$

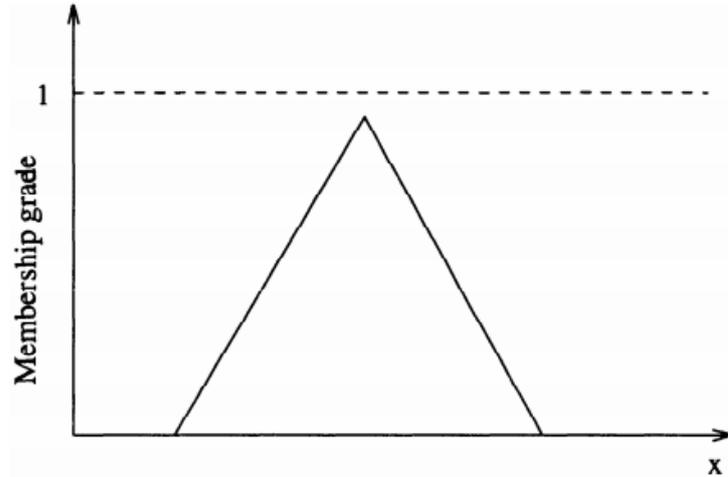


Figura 11: Conjunto *fuzzy* subnormal (CHI; YAN; PHAM, 1996).

são convexas para todo  $\alpha \in [0, 1]$ . As Figuras 12 e 13 mostram conjuntos *fuzzy* convexo e não-convexo, respectivamente.

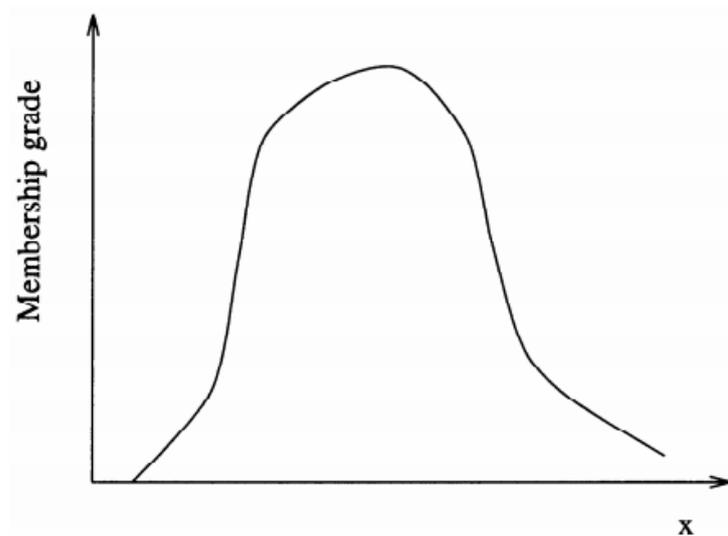


Figura 12: Conjunto *fuzzy* convexo (CHI; YAN; PHAM, 1996).

- *Ponto de cruzamento*: O Ponto de cruzamento em um conjunto *fuzzy*  $A$  é um elemento que tem seu grau de pertinência em  $A$  igual a 0,5.
- *Fuzzy Singleton*: Um *fuzzy singleton* é um conjunto *fuzzy* que só tem um valor de pertinência para um único valor.

#### 2.2.1.2 Operações em Conjuntos Fuzzy

As operações em conjuntos *fuzzy* a seguir, dadas em (CHI; YAN; PHAM, 1996), baseiam-se nos trabalhos de Zadeh. Os trabalhos originais de Zadeh foram editados em (YAGER et al., 1987).

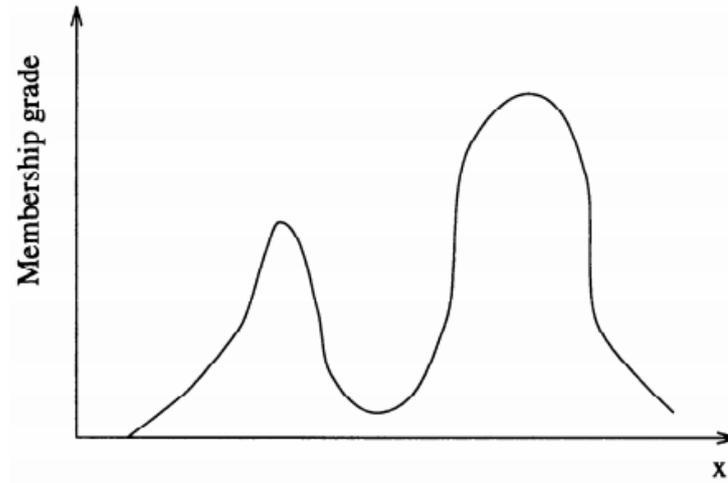


Figura 13: Conjunto *fuzzy* não-convexo (CHI; YAN; PHAM, 1996).

- *Intersecção*: A Figura 14 mostra a intersecção entre dois conjuntos *fuzzy* A e B. A intersecção *fuzzy* é interpretada como "A e B", que assume o mínimo valor entre dois graus de pertinência.

$$A(x) \cap B(x) = \sum \mu_A(x) \wedge \mu_B(x) \quad (3)$$

onde  $\wedge$  é o operador mínimo.

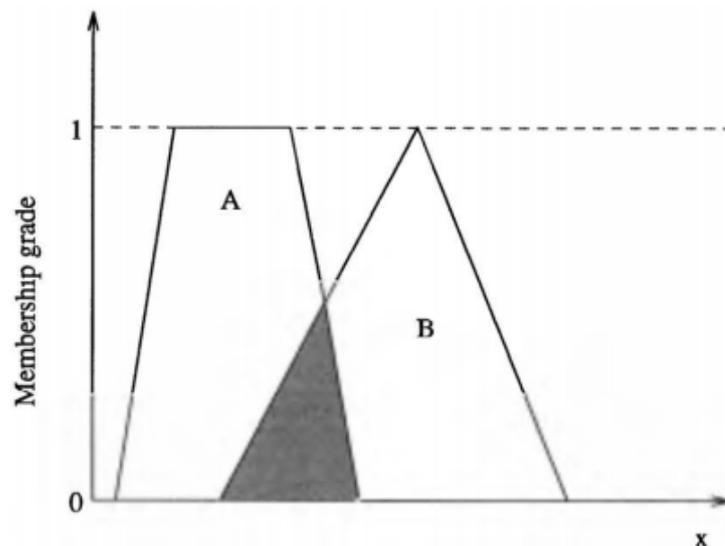


Figura 14: Intersecção entre os conjuntos *fuzzy* A e B (área sombreada) (CHI; YAN; PHAM, 1996).

- *União*: A Figura 15 mostra a união entre dois conjuntos *fuzzy* A e B. A união *fuzzy* é interpretada como "A ou B", que assume o valor dos dois graus de pertinência.

$$A(x) \cup B(x) = \sum \mu_A(x) \vee \mu_B(x) \quad (4)$$

onde  $\vee$  é o operador máximo.

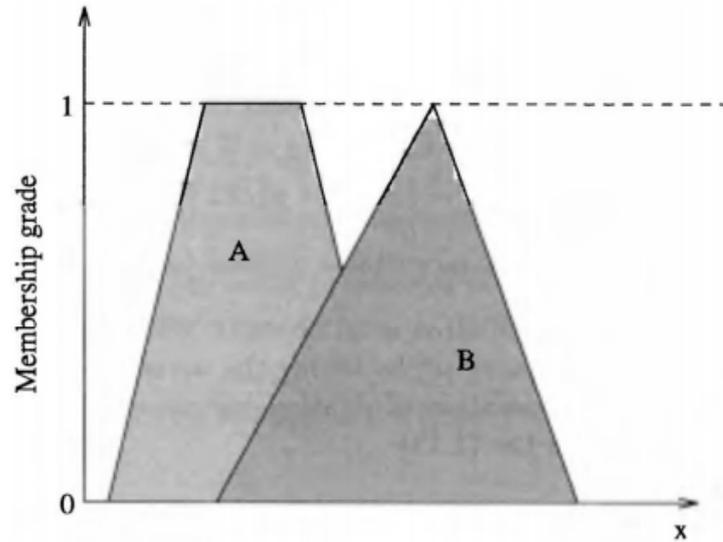


Figura 15: União entre os conjuntos *fuzzy* A e B (área sombreada) (CHI; YAN; PHAM, 1996).

- *Complemento*: O complemento de um conjunto *fuzzy* A, que pode ser entendido como “NÃO (A)”, é definido por

$$\bar{A} = \sum 1 - \mu_A(x) \quad (5)$$

onde  $\bar{A}$  representa o complemento de A. A Figura 16 demonstra o complemento em conjuntos *fuzzy*.

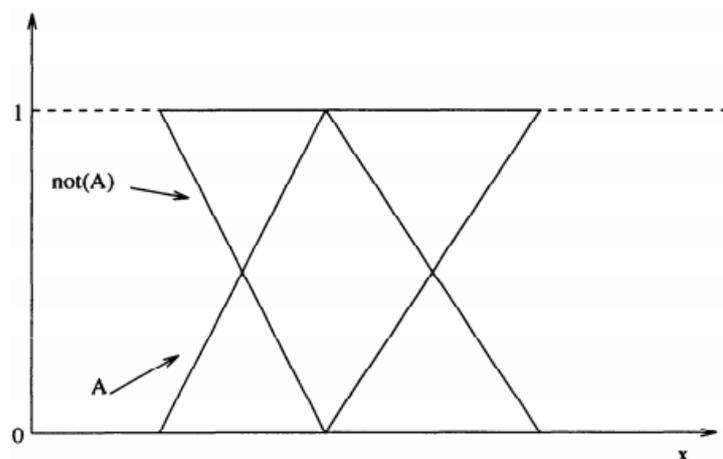


Figura 16: Conjunto *fuzzy* A e seu complemento (CHI; YAN; PHAM, 1996).

- *Combinação convexa*: A combinação convexa é um operador que combina diferentes conjuntos *fuzzy* em um único usando os pesos atribuídos a cada conjunto

*fuzzy*. A função de pertinência total  $\mu_T(x)$  como resultado da combinação con-  
vexa das funções de pertinência  $\mu_{A_1}, \dots, \mu_{A_n}$  é definida por

$$\mu_T(x) = w_1(x)\mu_{A_1}(x) + w_2(x)\mu_{A_2}(x) + \dots + w_n(x)\mu_{A_n}(x) \quad (6)$$

onde  $w_1, w_2, \dots, w_n$  são os pesos dos conjuntos *fuzzy*  $A_1, A_2, \dots, A_n$ , respectiva-  
mente, tal que:

$$w_1(x) + w_2(x) + w_n(x) = 1 \quad (7)$$

onde o sinal “+” nas Equações 6 e 7 denota uma adição aritmética.

- **Concentração:** A concentração de um conjunto *fuzzy* produz uma redução por elevar à segunda potência a função de pertinência desse conjunto *fuzzy*. Se um conjunto *fuzzy*  $A$  é descrito como

$$A = \{\mu_1/x_1 + \mu_2/x_2 + \dots + \mu_n/x_n\}$$

então o concentrador *fuzzy* aplicado ao conjunto *fuzzy*  $A$  é definido da seguinte  
maneira:

$$CON(A) = A^2 = \{\mu_1^2/x_1 + \mu_2^2/x_2 + \dots + \mu_n^2/x_n\} \quad (8)$$

onde  $CON(A)$  é o concentrador aplicado a  $A$ .

- **Dilatação:** Dilatação *fuzzy* é um operador que incrementa o grau de crença em cada objeto de um conjunto *fuzzy* tomando a raiz da função de pertinência. Claramente, o operador de dilatação tem o efeito oposto ao operador de concentração definido na Equação 8:

$$DIL(A) = A^{0.5} = \{\mu_1^{0.5}/x_1 + \mu_2^{0.5}/x_2 + \dots + \mu_n^{0.5}/x_n\} \quad (9)$$

onde  $DIL(A)$  é o dilatador aplicado ao conjunto *fuzzy*  $A$ .

- **Soma e subtração fuzzy:** As operações de soma e subtração aplicadas a um conjunto *fuzzy* dão um efeito intermediário de concentração e dilatação, respectivamente. Se  $A$  é um conjunto *fuzzy*, então a soma e a subtração podem ser definidas, por exemplo, como

$$Plus(A) = A^{1.25} \quad (10)$$

$$Minus(A) = A^{0.75} \quad (11)$$

- **Intensificação fuzzy:** Intensificação é o operador que incrementa a função de

pertinência de um conjunto *fuzzy* acima do ponto de cruzamento ( $\mu = 0.5$ ), e decrementa o valor da função de pertinência abaixo do ponto de cruzamento. Se  $A$  é um conjunto *fuzzy*, então a intensificação aplicada a  $A$  é definida como

$$\begin{aligned}\mu_{INT(A)}(x) &\geq \mu_A(x), \mu_A(x) \geq 0.5 \\ \mu_{INT(A)}(x) &\leq \mu_A(x), \mu_A(x) \leq 0.5\end{aligned}\tag{12}$$

- *Soma limitada*: A soma limitada de dois conjuntos *fuzzy*  $A$  e  $B$  nos universos  $X$  e  $Y$  com funções de pertinência  $\mu_A(x)$  e  $\mu_B(x)$  respectivamente, é definida por

$$A \oplus B = \mu_{A \oplus B} = 1 \wedge (\mu_A(x) + \mu_B(x))\tag{13}$$

onde o sinal “+” é o operador aritmético.

- *Produto limitado*: O produto limitado de dois conjuntos *fuzzy*  $A$  e  $B$  assim como descrita na soma é definida por

$$A \otimes B = \mu_{A \otimes B} = 0 \vee (\mu_A(x) + \mu_B(x) - 1)\tag{14}$$

### 2.2.1.3 Relações Fuzzy

Relações *fuzzy* têm sido largamente investigadas na literatura devido às suas importantes aplicações inerentes. O uso de relações *fuzzy* provê uma forma natural de inferir uma conclusão a partir de expressões humanas que incluem imprecisão (CHI; YAN; PHAM, 1996).

Baseado no conceito de relação *fuzzy*, algumas regras para inferência *fuzzy* foram propostas por Zadeh (ZADEH, 1975, 1988).

- *Regra de vinculação* (do inglês, *Entailment Rule*): O princípio de vinculação permite inferir a partir de uma proposição *fuzzy* “ $X$  é  $A$ ” uma proposição menos específica “ $X$  é  $B$ ”. Uma regra de vinculação é expressa na forma canônica como

$$\frac{X \text{ é } A}{A \subset B \rightarrow \mu_A(x) \leq \mu_B(x), x \in X} \quad (15)$$

(portanto)  $X$  é  $B$

Pode-se notar que a linha é usada para separar as premissas da conclusão.

- *Regra conjuntiva*: Se as premissas *fuzzy* são expressas na forma “ $X$  é  $A_1$ ,  $X$  é  $A_2$ , ...,  $X$  é  $A_n$ .”, então a conclusão induzida pela regra conjuntiva é a intersecção entre todas as premissas *fuzzy* – todas as premissas são conectadas pelo ope-

rador AND ( $\wedge$ ).

$$\frac{\begin{array}{l} X \text{ é } A_1(16) \\ X \text{ é } A_2 \\ \dots \\ X \text{ é } A_n \end{array}}{\text{(portanto) } X \text{ é } A_1 \cup A_2 \cup \dots \cup A_n \rightarrow \mu_{A_1 \cup A_2 \cup \dots \cup A_n}(x) = \mu_{A_1}(x) \wedge \mu_{A_2}(x) \wedge \dots \wedge \mu_{A_n}(x)}$$

- *Regra de projeção*: A projeção de uma relação binária R de X e Y no domínio X é definida como

$$\frac{(X, Y) \text{ é } R}{\text{(portanto) } X \text{ é } XR} \quad (17)$$

onde XR é a projeção de R em X, e a função de pertinência é dada por

$$\mu_{XR}(x) = \bigvee_y \mu_R(x, y), \quad x \in X, y \in Y$$

- *Regra composicional*: A regra composicional de inferência pode ser aplicada para inferir um novo subconjunto *fuzzy*  $B'(Y)$  através de  $A'(X) \circ R_{A \times B}(X, Y)$  em que a função de pertinência de  $B'$  é construída pelo produto max-min do vetor de  $\mu_{A'}(x)$  e a matriz de relação de  $\mu_{A \times B}(x, y)$ , para  $x \in X$ , e  $y \in Y$ , isto é,

$$\frac{\begin{array}{l} A \text{ é } X \\ B \text{ é } Y \\ (X, Y) \text{ é } R \\ A' \text{ é } X' \end{array}}{\text{(portanto) } B' \text{ é } A' \circ R.} \quad (18)$$

### 2.2.2 Funções de Pertinência

A escolha da função de pertinência é o primeiro e essencial passo em aplicações que utilizam lógica *fuzzy*, e esta tem grande impacto no desempenho do sistema. A função de pertinência estabelece uma correspondência entre um elemento e o valor que indica o grau de pertinência dele no conjunto (CHI; YAN; PHAM, 1996).

Além da seleção heurística de funções de pertinência, como geralmente é feito no desenvolvimento de sistemas de controle *fuzzy*, muitas outras técnicas têm sido propostas para produzir funções de pertinência que reflitam a distribuição dos dados; estas usam algoritmos de aprendizado supervisionado e não-supervisionado. Funções de pertinência que utilizam aprendizado não são apenas importantes, mas também viáveis no reconhecimento de padrões em imagens por causa do grande número de padrões de treinamento disponíveis (CHI; YAN; PHAM, 1996).

De acordo com a intenção original na introdução dos sistemas *fuzzy*, uma função de pertinência deve ser subjetiva. Entretanto, um conjunto de funções de pertinência subjetivas ou são muito difíceis de escolher, devido a falta de compreensão da abordagem humana a determinado problema, ou não pode produzir resultados satisfatórios. Portanto, muitos pesquisadores propuseram combinar métodos estatísticos e sistemas *fuzzy* na resolução de problemas. Usar funções de pertinência que são geradas a partir de dados de treinamento por uma das várias técnicas de agrupamento é uma maneira de alcançar esta combinação (CHI; YAN; PHAM, 1996).

Um algoritmo de agrupamento pode ser aplicado para estimar a atual distribuição dos dados e os *clusters* resultantes podem ser usados para produzir uma função de pertinência que interpretará melhor os dados. DICKERSON; KOSKO (1993) propuseram uma técnica de aprendizado para construir funções de pertinência por usar AVQ (*Adaptive Vector Quantization*). Além do AVQ, outras técnicas que podem ser utilizadas para gerar funções de pertinência são o algoritmo de agrupamento C-Means e os Mapa Auto-Organizados (SOM) (CHI; YAN, 1995, 1994; CHI; WU; H., 1995a,b).

### 2.2.2.1 Seleção heurística

A escolha de funções de pertinência é frequentemente determinada de maneira heurística e subjetiva (BEZDEK; PAL, 1992; DRIANKOV; HELLENDORRN; REINFRANK, 1993). As funções de pertinência mais comumente usadas são nas formas de triângulo, trapézio e sino, como mostra a Figura 17, que são denotadas como  $\Lambda(x; a, b, c)$ ,  $\Pi(x; a, c, b, d)$  e  $\pi(x; b, c)$ , respectivamente. Elas são definidas como

$$\Lambda(x; a, b, c) = \begin{cases} 0 & x \leq a, \\ (x - a)/(b - a) & a < x \leq b, \\ (c - x)/(c - b) & b < x \leq c, \\ 0 & x > c \end{cases} \quad (19)$$

$$\Pi(x; a, c, b, d) = \begin{cases} 0 & x \leq a, \\ (x - a)/(b - a) & a < x \leq b, \\ 1 & b < x \leq c, \\ (d - x)/(d - c) & c < x \leq d, \\ 0 & x > d \end{cases} \quad (20)$$

e

$$\pi(x; b, c) = \begin{cases} S(x; c - b, c - b/2, c) & x \leq c, \\ 1 - S(x; c, c + b/2, c + b) & x > c \end{cases} \quad (21)$$

onde a função S de Zadeh é definida como

$$S(x; a, b, c) = \begin{cases} 0 & x \leq a, \\ 2\left(\frac{x-a}{c-a}\right)^2 & a < x \leq b, \\ 1 - 2\left(\frac{x-c}{c-a}\right)^2 & b < x \leq c, \\ 1 & x > c \end{cases} \quad (22)$$

Além de escolher a forma da função de pertinência, deve-se construir um conjunto de funções de pertinência para todos os subconjuntos *fuzzy* (termos linguísticos) para cada entrada e cada saída. Algumas propriedades das funções de pertinência pico, função de pertinência simétrica ou assimétrica, largura esquerda e direita, *crosspoint*, e nível de *crosspoint* (CHI; YAN; PHAM, 1996):

- *Pico*: Assumindo que  $\mu_A : \mathbf{X} \rightarrow [0, 1]$ , o ponto de pico é definido como o valor  $x_{peak}$  do domínio  $\mathbf{X}$  que torna  $\mu_A(x_{peak}) = 1$ . A Figura 18 ilustra o ponto de pico em uma função de pertinência triangular.
- *Larguras direita e esquerda*: A largura esquerda ( $w_l$ ) é o tamanho do intervalo entre o ponto de pico e o ponto mais a esquerda  $x_{left}$  que tem  $\mu_A(x_{left}) = 0$ , isto é,

$$w_l = x_{peak} - x_{left}. \quad (23)$$

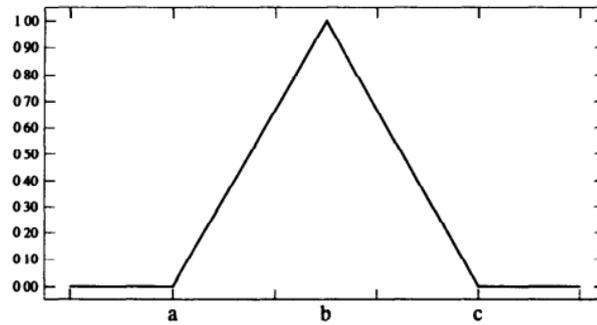
Similarmente, a largura direita ( $w_r$ ) é o tamanho do intervalo entre o ponto de pico e o ponto mais a esquerda  $x_{right}$  que tem  $\mu_A(x_{right}) = 0$ , isto é,

$$w_r = x_{peak} - x_{right}. \quad (24)$$

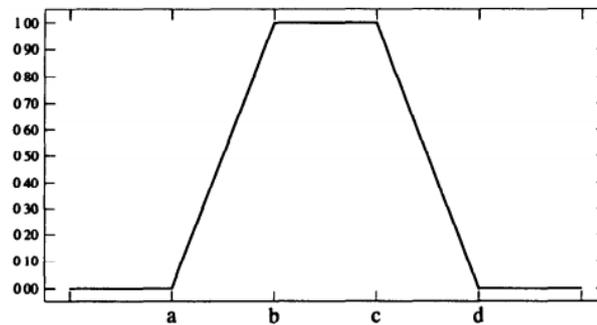
Se  $w_l = w_r$ , então a função de pertinência é simétrica. Do contrário é assimétrica. A Figura 18 ilustra uma função de pertinência assimétrica.

- *Crosspoint e Nível de crosspoint*: Um *crosspoint* é o valor  $x_{cross}$  em  $\mathbf{X}$  onde  $\mu_A(x_{cross}) = \mu_B(x_{cross}) > 0$ . Um nível de *crosspoint* pode então ser definido como o grau de pertinência de  $x_{cross}$  em qualquer subconjunto *fuzzy*  $A$  ou subconjunto *fuzzy*  $B$ . Note que duas funções de pertinência vizinhas podem ter mais que um *crosspoint*. A Figura 19 mostra duas funções de pertinência representando dois termos linguísticos diferentes,  $A$  e  $B$  de  $\mathbf{X}$ . Ainda na Figura 19, o nível de *crosspoint* é 0.67.

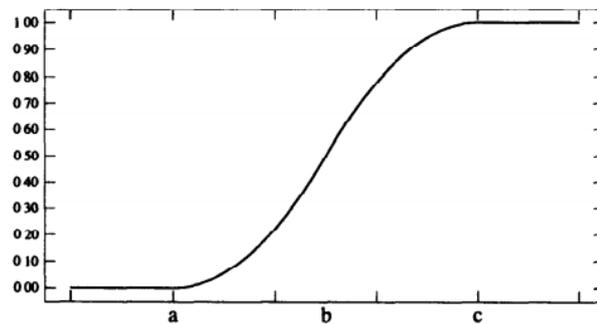
Em aplicações de controle *fuzzy*, um conjunto de funções de pertinência são escolhidas baseadas nos seguintes princípios (CHI; YAN; PHAM, 1996):



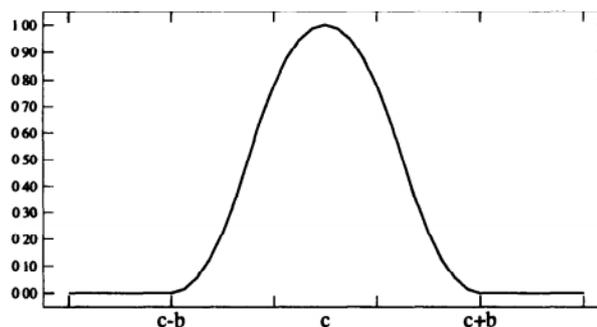
(a)



(b)



(c)



(d)

Figura 17: (a) Função de pertinência triangular; (b) Função de pertinência trapezoidal; (c) Função de pertinência em forma de S; e (d) Função de pertinência em forma de seno. (CHI; YAN; PHAM, 1996)

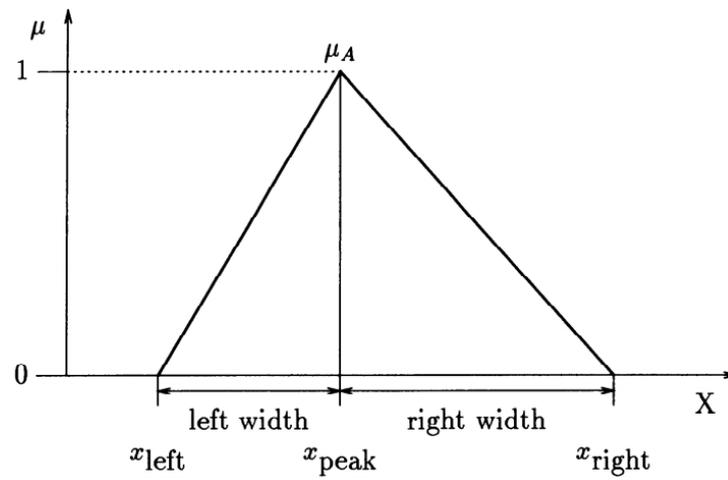


Figura 18: Pico e larguras em uma função de pertinência triangular assimétrica (CHI; YAN; PHAM, 1996).

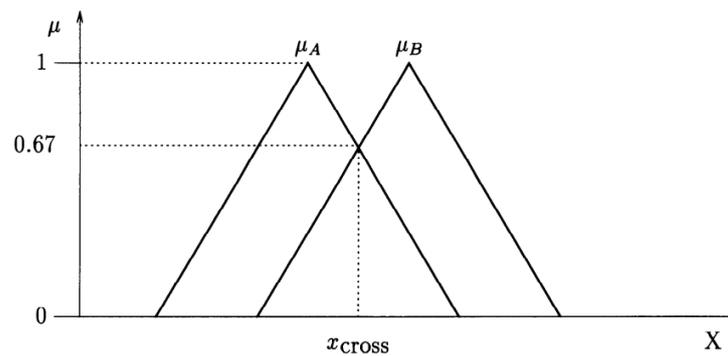


Figura 19: *Crosspoint* e nível de *crosspoint* de uma função de pertinência triangular (CHI; YAN; PHAM, 1996).

- Uma função de pertinência simétrica é geralmente aplicada a cada subconjunto *fuzzy* de entrada ou saída.
- Estas funções de pertinência são uniformemente distribuídas na faixa de valores de cada entrada ou saída. Geralmente o *crosspoint* delas é 0,5.
- Diferentes entradas e saídas podem ter a mesma forma da função de pertinência ou não, e o mesmo número de termos linguísticos ou não.

O valor de cada variável é geralmente normalizado no intervalo  $[-1, 1]$  ou  $[0, 1]$ . A Figura 20 mostra funções de pertinência triangular 20(a) e trapezoidal 20(b) com cinco rótulos linguísticos, L (baixo), SL (quase baixo), M (médio), SH (quase alto), e H (alto).

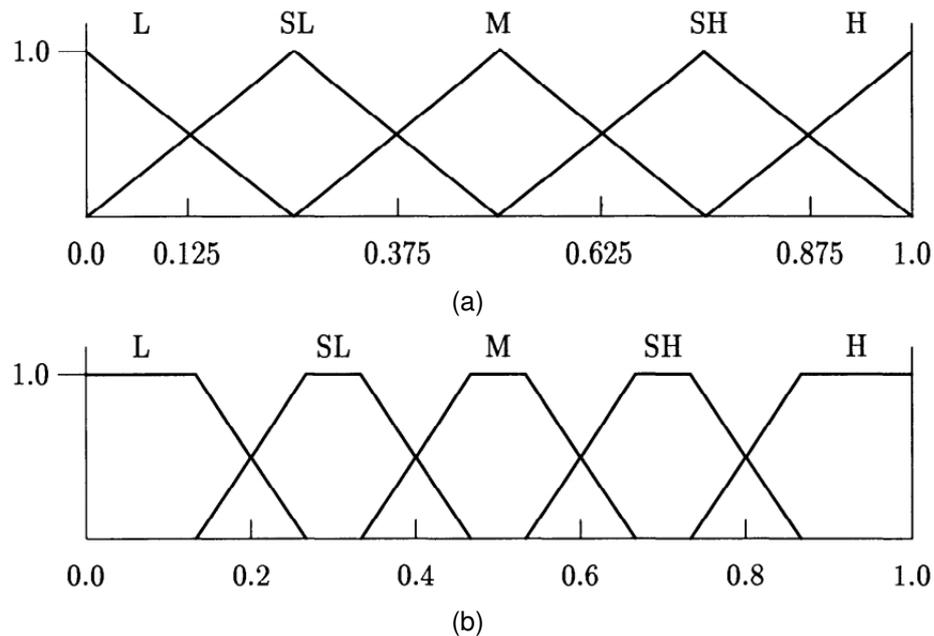


Figura 20: Funções de pertinência triangular (a) e trapezoidal (b) com cinco rótulos linguísticos (CHI; YAN; PHAM, 1996).

#### 2.2.2.2 Abordagens de agrupamento

Funções de pertinência escolhidas heurísticamente não refletem a atual distribuição dos dados nos espaços de entrada e saída. Elas não são adequadas no reconhecimento de padrões *fuzzy*. Para construir funções de pertinência a partir dos dados disponíveis, pode-se usar uma técnica de agrupamento para particionar os dados e então produzir funções para os *clusters* resultantes (CHI; YAN; PHAM, 1996).

Agrupamento, ou *clustering*, é um processo para obter uma partição  $P$  de um conjunto  $E$  de  $N$  objetos  $X_i (i = 1, 2, \dots, N)$ , por usar uma medida de semelhança (ou diferença)  $d$ . Uma *partição*  $P$  é um conjunto de subconjuntos disjuntos de  $E$  e um elemento  $P_s$  de  $P$  é chamado *cluster*. Os centros dos *clusters* são chamados centróides ou protótipos (CHI; YAN; PHAM, 1996).

Baseado em diferentes medidas de similaridade, diferentes *clusters* são obtidos. A Figura 21 mostra os *clusters* resultantes de entradas 2D onde em (a) foi aplicada a diferença de ângulo como medida de semelhança, e em (b) foi usada a distância euclidiana.

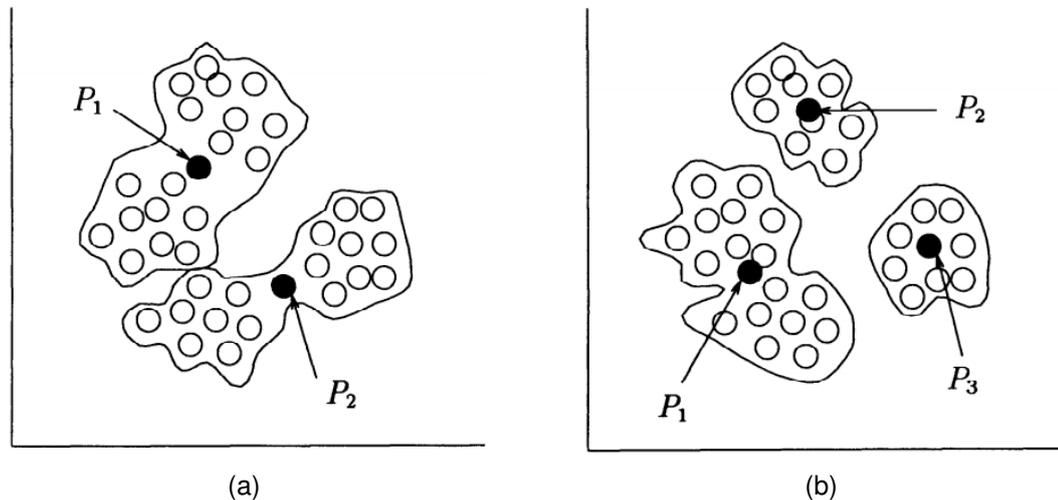


Figura 21: *Clusters* resultantes da aplicação de duas medidas de semelhança, (a) diferença de ângulo, e (b) distância euclidiana.  $P_1$ ,  $P_2$ , e  $P_3$  são os centróides (CHI; YAN; PHAM, 1996).

Muitas técnicas de agrupamento têm sido propostas. Nas Seções a seguir serão detalhadas as técnicas de agrupamento Aprendizado com *Adaptive Vector Quantization* (AVQ) e Abordagem de Mapas Auto-Organizados (SOM) e como se aplicam ao gerar funções de pertinência.

#### 2.2.2.2.1 Aprendizado com *Adaptive Vector Quantization* (AVQ)

DICKERSON; KOSKO (1993) propuseram uma técnica para aprendizado de funções *fuzzy* por usar AVQ. Nesta abordagem, o AVQ não-supervisionado foi usado para estimar a localização dos centróides e as matrizes de covariância dos *clusters* no espaço de entrada-saída. A matriz de covariância de cada vetor quantização define um elipsóide com seu centro sendo o centróide do *cluster*. Regras *fuzzy* são aprendidas dos elipsóides resultantes.

O AVQ é uma sistema neural mostrado na Figura 22 que usa aprendizado competitivo não-supervisionado. O AVQ aprende a matriz de conexão sináptica  $M = [m_1, m_2, \dots, m_p]$  ( $p$  é o número de neurônios) entre os neurônios de entrada na camada  $F_X$  e os neurônios de saída na camada  $F_Y$  que será uma estimativa inicial da função densidade de probabilidade desconhecida dos dados. A seguinte Equação de diferença estocástica é usada para estimar  $m_i = [m_{i1}, m_{i2}, \dots, m_{in}]^T$  ( $n$  é o número de

entradas):

$$m_i(k+1) = \begin{cases} m_i(k) + \eta_k [x_k - m_i(k)] & \text{se o } i\text{-ésimo neurônio vence,} \\ m_i(k) & \text{se o } i\text{-ésimo neurônio perde} \end{cases} \quad (25)$$

onde  $\eta_k$  é um coeficiente de aprendizado dado por

$$\eta_k = 0,1 \left(1 - \frac{k}{k_{max}}\right). \quad (26)$$

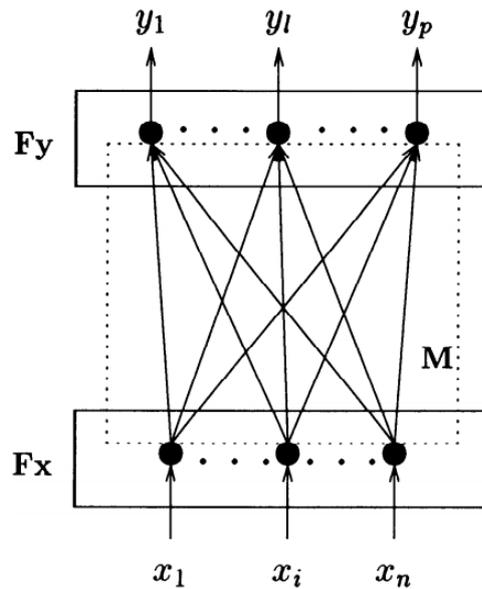


Figura 22: Arquitetura do AVQ (CHI; YAN; PHAM, 1996).

O vetor converge aos centróides padrão. Por outro lado, a matriz de covariância local ( $V$ ) que dá uma segunda estimativa é aprendida por

$$V_i(k+1) = \begin{cases} V_i(k) + \beta_k [(x_k - m_i(k))(x_k - m_i(k))^T - V_i(k)] & \text{se o } i\text{-ésimo neurônio vence,} \\ V_i(k) & \text{se o } i\text{-ésimo neurônio perde} \end{cases} \quad (27)$$

onde  $\beta_k$  é um coeficiente de aprendizado similar a  $\eta_k$ . Ambos coeficientes de aprendizado diminuem em função do tempo.

O algoritmo AVQ competitivo consiste dos seguintes passos (DICKERSON; KOSKO, 1993):

1. Inicializa-se  $m_i(0)$  ( $i = 1, \dots, p$ ) arbitrariamente com os  $p$  vetores de entrada.
2. Para uma amostra  $x_k$  encontra o vetor sináptico mais próximo ou “vencedor”  $m_i(k)$  baseado na medida de distância Euclidiana:

$$\| m_k - m_i(k) \| = \min_j \| x_k - m_j(k) \| . \quad (28)$$

3. Atualiza-se o vetor sináptico vencedor  $m_i(k)$  de acordo com a Equação 25 e sua matriz de covariância local  $V_i(k)$  de acordo com a Equação 27.

Os dados são estatisticamente normalizados tal que todas as entradas tem peso igual na medida de distância. Funções aprendidas pelo algoritmo AVQ tem sido aplicadas a segmentação de imagens em mapas geográficos (CHI; YAN, 1994).

#### 2.2.2.2.2 Mapas Auto-Organizados de Kohonen (SOM)

Os Mapas Auto-Organizados de Kohonen (SOM) é um algoritmo de aprendizado não-supervisionado que modifica os vetores de pesos ( $W_i, i = 1, 2, \dots, m$ , e  $m$  é o número de nodos no mapa) de uma rede para modelar as características encontradas nos dados de treinamento (KOHONEN, 1990) (ver Figura 23).

Um mapa topológico é automaticamente organizado por um processo cíclico de comparar padrões de entrada para vetores de pesos para cada nodo. O vetor de pesos para os dados introduzidos é seletivamente otimizados para representar uma média dos dados de treinamento. Como resultado, todos os dados de treinamento são representados pelos vetores de pesos do mapa que podem ser considerados centróides dos padrões de entrada.

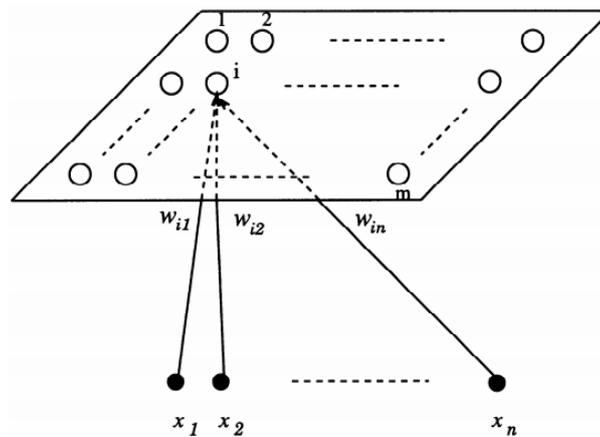


Figura 23: Mapa Auto-Organizado de Kohonen bidimensional de uma camada ( $m$ : número de nodos no mapa;  $(x_1, x_2, \dots, x_n)$ : as entradas; e  $W_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ : o vetor de pesos do nodo  $i$ ) (CHI; YAN; PHAM, 1996).

Começando com um conjunto de nodos organizados aleatoriamente, e procedendo a criação de um mapa de características representando os protótipos dos dados de entrada, o procedimento de treinamento é como se segue (CHI; YAN; PHAM, 1996):

1. Inicializa-se os pesos  $w_{ij} (1 \leq i \leq m, 1 \leq j \leq n)$  com valores aleatórios pequenos, onde  $n$  é o número de entradas. Defina o raio inicial da vizinha em torno do nodo  $i$  como  $N_i(0)$ .

2. Presentes as entradas  $x_1(t), x_2(t), \dots, x_n(t)$ , onde  $x_j(t)$  é a  $j$ -ésima entrada no tempo  $t$ .
3. Calcula-se a distância  $d_i$  entre as entradas e o nodo  $i$  pela Equação

$$d_i = \sum_{j=1}^n (x_j(t) - w_{ij}(t))^2. \quad (29)$$

4. Determina-se  $i_*$  que minimiza  $d_i$ .
5. Atualiza-se os pesos para o nó  $i_*$  e seus vizinhos em  $N_{i_*}(t)$ . Os novos pesos para  $i$  em  $N_{i_*}(t)$  são

$$w_{ij}(t+1) = w_{ij}(t) + a(t)(x_j(t) - w_{ij}(t)) \quad (30)$$

onde  $a(t)$  é a taxa de aprendizagem. Ambos  $a(t)$  e  $N_{i_*}(t)$  decrescem ao passo que  $t$  aumenta.

6. Se o processo atingiu o número máximo de iterações, para; do contrário, vai para o passo 2.

O algoritmo de auto-organização tem características de preservar a topologia no sentido que os nodos fisicamente localizados próximos um dos outros estarão em classes de vetores de entrada que estarão próximas umas das outras.

Isto é uma vantagem para usar-se mapas 2-D em que pode-se visualizar facilmente o relacionamento entre os nodos. Entretanto, mapas 1-D ou de mais dimensões podem ser utilizados por diversas aplicações. Um exemplo de uso pode ser visto na Figura 24, onde mostra uma mapa auto-organizado de  $15 \times 15$  obtido de um conjunto de 10.426 amostras de dígitos escritos a mão.

Tal como pode-se ver no mapa, alguns dígitos escritos a mão que são bastante parecidos estão próximos uns dos outros. Por exemplo, alguns dos protótipos de “7” são muito similares aos de “9” ou “1”, e isto é verdadeiramente indicado no mapa. O mesmo pode ser observado em alguns dos protótipos de “2” e “3”. Esta propriedade dos mapa auto-organizados, de preservar a topologia, é bastante demandada para gerar funções de pertinência mais robustas, particularmente, quando é feita uma fusão entre duas funções de pertinência (CHI; YAN; PHAM, 1996).

Pode-se usar protótipos SOM e variâncias para determinar funções de pertinência para cada entrada. Cada protótipo gera uma função de pertinência triangular para cada variável de entrada com o pico da função localizado em cada componente do protótipo. As larguras dos subconjuntos *fuzzy* são determinadas pelas variâncias correspondentes.

4	9	4	4	7	7	7	9	9	9	8	8	8	4	4
4	4	4	4	9	7	7	9	9	9	8	8	1	1	4
4	4	4	4	9	7	7	7	9	7	9	1	1	1	7
4	4	4	4	4	2	7	7	9	7	1	1	1	4	2
5	5	4	4	4	9	7	7	7	1	1	1	8	1	5
5	5	8	9	8	8	7	7	7	1	1	1	9	8	0
5	5	5	8	8	8	8	7	1	1	1	6	5	6	0
0	5	8	8	8	8	8	2	2	8	6	6	6	0	0
0	0	0	8	8	8	2	2	2	6	6	6	0	0	0
0	0	5	5	8	8	2	2	2	2	6	6	0	0	0
0	0	0	5	5	3	3	2	2	2	6	6	6	0	0
0	0	0	5	3	3	3	3	2	2	2	6	6	6	6
6	0	5	3	3	3	3	3	2	2	2	2	6	6	6
6	8	9	3	3	3	3	3	3	2	2	2	2	6	4
4	9	9	7	7	3	3	3	8	2	2	2	2	2	4

Figura 24: Um mapa auto-organizado de  $15 \times 15$  obtido de um conjunto de 10.426 amostras de dígitos escritos a mão (CHI; YAN; PHAM, 1996).

Supondo-se  $m$  protótipos  $W_i (i = 1, 2, \dots, m)$  com as variâncias correspondentes  $V_i (i = 1, 2, \dots, m)$ . Assuma  $W_i [w_{i1}, w_{i2}, \dots, w_{in}]^T$  e  $V_i [v_{i1}, v_{i2}, \dots, v_{in}]^T$ . Os centros das funções de pertinência triangulares no  $j$ -ésimo eixo são  $w_{1j}, w_{2j}, \dots, w_{mj}$ . As regiões correspondentes são definidas para

$$[w_{1j} - av_{1j}, w_{1j} + av_{1j}],$$

$$[w_{2j} - av_{2j}, w_{2j} + av_{2j}],$$

.....,

$$[w_{mj} - av_{mj}, w_{mj} + av_{mj}].$$

O parâmetro  $a (1 \leq a \leq 4)$  é um valor escalar que reflete a imprecisão dos dados. Um  $a$  grande é usado para dados mais imprecisos. Funções de pertinência gerados a partir de mapas SOM encontraram aplicação em reconhecimento de dígitos escritos a mão.

### 2.2.2.3 Funções de pertinência vizinhas

Ainda, pode-se combinar funções de pertinência vizinhas a fim de gerar funções de pertinência mais robustas. Em aplicações de reconhecimento de padrões, pode-se facilmente ter um grande número de exemplos de treinamento tal que o número de regras *fuzzy* produzidas pode ser  $(\prod_{k=1}^n l_k)$ , onde a entrada  $k$  tem  $l_k$  variáveis linguísticas. Por reduzir o número de variáveis linguísticas de cada entrada, pode-se reduzir o

número de regras *fuzzy* produzidas.

Uma forma de fazer isso é combinar duas funções de pertinência com alto grau de sobreposição. Se duas funções de pertinência são  $\Pi(x_j; a_{i-1}, b_{i-1}, c_{i-1}, d_{i-1})$  e  $\Pi(x_j; a_i, b_i, c_i, d_i)$ , elas serão combinadas se a seguinte condição é satisfeita (CHI; YAN; PHAM, 1996):

$$\left| \frac{b_i + c_i}{2} - \frac{b_{i-1} + c_{i-1}}{2} \right| \leq l_T \quad (31)$$

onde  $l_T$  é um limiar pré-estabelecido. A nova função de pertinência após a combinação é  $\Pi(x_j; a_{i*}, b_{i*}, c_{i*}, d_{i*})$ , com os seguintes parâmetros:

$$\begin{aligned} a_{i*} &= \min(a_i, a_{i-1}), \\ b_{i*} &= \min(b_i, b_{i-1}), \\ c_{i*} &= \min(c_i, c_{i-1}), \\ d_{i*} &= \min(d_i, d_{i-1}) \end{aligned} \quad (32)$$

A Figura 25 ilustra o resultado do processo de combinação de funções de pertinência vizinhas.

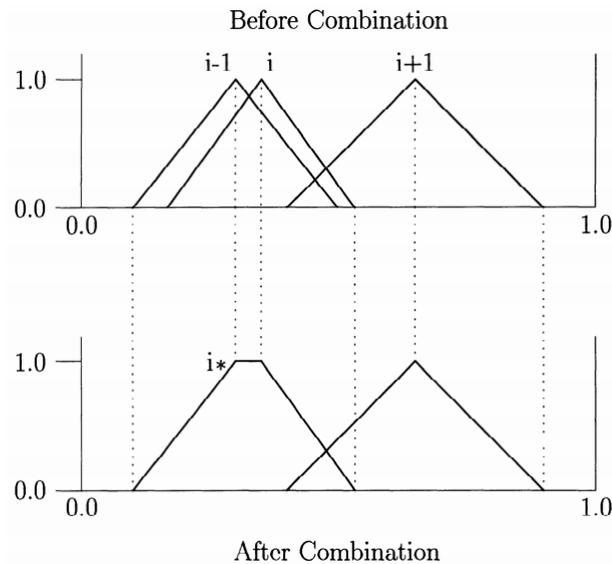


Figura 25: Demonstração do processo de combinação de funções de pertinência vizinhas (CHI; YAN; PHAM, 1996).

### 2.2.3 Modelos Fuzzy para o Processamento de Imagens

Não demorou muito para que a teoria dos conjuntos *fuzzy* tivesse seu potencial para aplicações no campo do processamento de imagens encontrado. BEZDEK; PAL (1992) compilaram uma série de artigos sobre a aplicação de modelos *fuzzy* no reconhecimento de padrões. Este trabalho apresentou contribuições significativas na solução de problemas da vida real a partir de modelos *fuzzy*.

Existem diversos modelos de conjuntos *fuzzy* para resolver problemas de reconhecimento de padrões e processamento de imagens. Dentre estes, pode-se citar: o uso de funções de pertinência, agrupamento *fuzzy*, sistemas baseados em regras *fuzzy*, entropia *fuzzy*, medida *fuzzy* e integral *fuzzy*.

Abaixo uma pequena introdução a alguns destes modelos (CHI; YAN; PHAM, 1996):

- *Funções de pertinência fuzzy*: A função de pertinência é a força fundamental de cada modelo *fuzzy*, pois é capaz de modelar a transição gradual de uma região menos distinta para outra de maneira sutil. O uso direto de funções de pertinência é eficaz no melhoramento de imagens, onde diferentes tipos de função de pertinência são usadas para reduzir a quantidade de iterações realizadas através de uma técnica de relaxamento e fornece uma maneira melhor de lidar com o histograma da imagem.
- *Agrupamento fuzzy*: Métodos de agrupamento têm sido o método de solução dominante para problemas de classificação de padrão. Na natureza, existem apenas alguns casos em que as diferenças entre padrões são claras. Portanto, a implementação de conjuntos *fuzzy* em algoritmos de agrupamento convencionais se torna naturalmente essencial. Muitos problemas envolvendo classificação em análise de imagens e visão computacional têm sido resolvidos ao usar técnicas de agrupamento *fuzzy*.
- *Correspondência de padrão fuzzy*: Muitos problemas de reconhecimento de padrões podem ser simplificados como uma tarefa de correspondência de padrão por linha ou ponto. Algoritmos *fuzzy*, que podem lidar com ambiguidade ou características *fuzzy* de padrões de linha ou pontos ruidosos, têm sido particularmente úteis na correspondência de padrão. Diversos algoritmos, incluindo várias técnicas de relaxamento *fuzzy* (OGAWA, 1984, 1994), têm sido desenvolvidas e aplicadas a um grande número de aplicações de reconhecimento de padrões.
- *Sistemas baseados em regras fuzzy*: Sistema baseado em regras é uma das áreas de pesquisa mais ativas na inteligência artificial. É amplamente utilizado em sistemas especialistas. Acredita-se que levar em conta a forma com que seres humanos percebem o mundo em sua volta seja uma abordagem robusta para lidar com problemas. Na vida real, quase tudo é incerteza, portanto, espera-se que um sistema baseado em regras *fuzzy* tenha melhor performance que um sistema baseado em regras *crisp* ao lidar com ambiguidade, incerteza e dados *fuzzy*. Sistemas de regras *fuzzy* têm encontrado aplicações em muitos campos incluindo controle, tomada de decisão, e reconhecimento de padrões (BEZDEK;

PAL, 1992; LEE, 1990a,b; KRISHAPURAM; LEE, 1992). Uma base de regras *fuzzy* consiste de um conjunto de regras *fuzzy* do tipo SE-ENTÃO que juntos a um mecanismo de inferência, um *fuzzificador*, e um *defuzzificador* formam um sistema baseado em regras *fuzzy*. O papel do *fuzzificador* é mapear pontos *crisp* no espaço de entrada para subconjuntos *fuzzy* por aplicar funções de pertinência às entradas. Portanto, a escolha das funções de pertinência é essencial para o desempenho do sistema.

## 2.3 Agrupamento

Em geral existem duas classes de técnicas de reconhecimento de padrões: os métodos supervisionados e os não-supervisionados. Em um método supervisionado, tem-se um conjunto de amostras de treinamento em diferentes classes. Para estes dados de treinamento, precisa-se encontrar uma função de mapeamento, ou construir um classificador, que pode ser um conjunto de regras *fuzzy*, uma rede neural, uma árvore de decisão, ou simplesmente um conjunto de Equações matemáticas, tal que as amostras sejam classificadas corretamente em suas classes.

Em métodos não supervisionados, não tem-se amostras de treinamento, e em alguns casos, não se sabe o número exato de classes. Neste caso tem-se um conjunto de dados não-rotulados:

$$x_1, x_2, \dots, x_n. \quad (33)$$

A tarefa é dividir estes dados em vários grupos de acordo com uma medida de similaridade ou estrutura inerente dos dados. Isto pode ser feito através de um procedimento de agrupamento. Em um procedimento de agrupamento *hard*, uma amostra é atribuída a um grupo ou não. Já em um procedimento de agrupamento *fuzzy*, uma amostra é atribuída a cada grupo de acordo com um valor de pertinência. Os valores de pertinência desempenham um importante papel no processo de agrupamento e tornam o processo de classificação mais flexíveis e robustos ao lidar com dados incertos e ruidosos.

### 2.3.1 Algoritmo C-Means

O algoritmo C-Means foi brevemente discutido anteriormente onde foi usado para agrupar dados a fim de obter os centróides e as correspondentes variâncias para gerar funções de pertinência. Agora, o foco é utilizar esta técnica como um procedimento de otimização iterativa para ser usado para classificação.

Seja

$$X = \{x_1, x_2, \dots, x_n\} \quad (34)$$

um conjunto de amostras divididas (agrupadas) em  $c$  classes. O processo de agrupamento pode ser considerado como um procedimento de otimização iterativa. Supondo que estas amostras tenham sido atribuídas a cada classe aleatoriamente ou não, a tarefa é ajustar as partições tal que algum tipo de medida de similaridade seja otimizada.

No algoritmo C-Means, a medida de similaridade é calculada em função da distância Euclidiana entre cada amostra e o centro da classe a qual pertence. A função critério usada para o processo de agrupamento é (THERRIEN, 1989):

$$J(V) = \sum_{k=1}^n \sum_{x_k \in C_i} |x_k - v_i|^2 \quad (35)$$

onde  $v_i$  é amostra média ou o centro das amostras do *cluster*  $i$ , e  $V = \{v_1, \dots, v_c\}$ .

Para aprimorar a similaridade entre amostras em cada *cluster*, pode-se minimizar a função critério, tal que todas as amostras estejam distribuídas mais compactamente em torno dos centros dos seus *clusters*. Definindo a derivada de  $J(V)$  com respeito a  $v_i$  para zero, obtém-se

$$\frac{\delta J(V)}{\delta v_i} = \sum_{k=1}^n \sum_{x_k \in C_i} (x_k - v_i) = 0. \quad (36)$$

Portanto, o local ótimo do centro de *cluster*  $v_i$  é

$$v_i = \frac{1}{n_i} \sum_{x_k \in C_i} x_k \quad (37)$$

onde  $n_i$  é o número de amostras na classe  $i$  e  $C_i$  contém todas as amostras da classe  $i$ .

Baseados na informação dos *clusters* iniciais e nas posições de seus centros, pode-se reagrupar as amostras de forma que a função critério seja minimizada. Na Equação 35, cada amostra  $x_k$  aparece apenas uma vez, isto é, está associada a um único centro de *cluster*. Obviamente, a função critério é minimizada se ela está associada ao centróide mais próximo. Então realoca-se  $x_k$  a classe  $i$  tal que  $(x_k - v_j)^2$  é mínimo quando  $j = i$ .

Tendo reagrupado as amostras, os centros dos *clusters* precisam ser recalculados para minimizar  $J(V)$ . Para os novos centros dos *clusters*, pode-se ainda reagrupar as amostras para reduzir  $J(V)$ . O processo pode ser repetido até que  $J$  não possa mais ser reduzido.

Em síntese, o procedimento de agrupamento C-Means consiste dos seguintes passos (CHI; YAN; PHAM, 1996):

1. Determine o número de *clusters*  $c$ .

2. Particione as amostras de entrada nos  $c$  *clusters* com base numa aproximação. Se nenhuma regra de aproximação existe, as amostras podem ser particionadas aleatoriamente.
3. Calcule os centros dos *clusters*.
4. Atribua cada amostra de entrada a classe de centro mais próximo.
5. Repita os passos 3 e 4 até que nenhuma mudança possa ser feita em  $J$ .

### 2.3.2 Algoritmo Fuzzy C-Means

No processo *hard*, cada amostra é atribuída a somente um *cluster* e todos os *clusters* são conjuntos disjuntos. Na prática, entretanto, existem muitos casos em que os *clusters* não são completamente disjuntos e os dados podem ser classificados como pertencendo mais a um *cluster* que a outro.

Tal situação não pode ser atendida por um processo de seleção *crisp*. Portanto, a separação dos *clusters* traz uma noção *fuzzy*, e as representações de estruturas de dados reais podem então serem tratadas com mais precisão por métodos de agrupamento *fuzzy*. Nestes casos, é necessário descrever a estrutura de dados em termos dos *clusters fuzzy*.

O algoritmo *Fuzzy C-Means* (FCM) é o mais conhecido e a técnica de agrupamento *fuzzy* mais amplamente usada. Este algoritmo é desenvolvido baseado na minimização iterativa da seguinte função critério (BEZDEK; PAL, 1992; ZIMMERMANN, 1985; BEZDEK, 1981):

$$J(U, V) = \sum_{i=1}^c \sum_{k=1}^n u_{ik}^m |x_k - v_i|^2 \quad (38)$$

onde

- $x_1, \dots, x_n$  são  $n$  vetores de dados de amostra;
- $V = \{v_1, \dots, v_c\}$  são os centros dos *clusters*;
- $U = [u_{ik}]$  é uma matriz  $c \times n$ , onde  $u_{ik}$  é o  $i$ -ésimo valor de pertinência da  $k$ -ésima amostra de entrada  $x_k$ , e os valores de pertinência satisfazem as seguintes condições

$$0 \leq u_{ik} \leq 1 \quad i = 1, 2, \dots, c; k = 1, 2, \dots, n \quad (39)$$

$$\sum_{i=1}^c u_{ik} = 1 \quad k = 1, 2, \dots, n \quad (40)$$

$$0 < \sum_{k=1}^n u_{ik} < n \quad i = 1, 2, \dots, n \quad (41)$$

- $m \in [1, \infty)$  é um fator de peso expoente.

A função objetivo é a soma das distâncias Euclidianas entre cada amostra e seu centro de *cluster* correspondente elevada ao quadrado, com as distâncias sendo ponderadas pela pertinência *fuzzy*.

O algoritmo é iterativo e faz uso das seguintes Equações (CHI; YAN; PHAM, 1996):

$$v_i = \frac{1}{\sum_{k=1}^n u_{ik}^m} \sum_{k=1}^n u_{ik}^m x_{ik} \quad i = 1, 2, \dots, c \quad (42)$$

$$u_{ik} = \frac{\left[ \frac{1}{|x_k - v_i|^2} \right]^{\left(\frac{1}{m-1}\right)}}{\sum_{j=1}^c \left[ \frac{1}{|x_k - v_j|^2} \right]^{\left(\frac{1}{m-1}\right)}} \quad i = 1, 2, \dots, c; k = 1, 2, \dots, n \quad (43)$$

Para calcular um centro de *cluster*, todas as amostras de entrada são consideradas e as contribuições das amostras são ponderadas pelos valores de pertinência. Para cada amostra, seu valor de pertinência em cada classe depende de sua distância ao centro de *cluster* correspondente. O fator  $m$  reduz a influência de pequenos valores de pertinência. Quanto maior o valor de  $m$ , menor é a influência de amostras com menor valor de pertinência.

O FCM consiste dos seguintes passos (CHI; YAN; PHAM, 1996):

1. Inicialize  $U^{(0)}$  aleatoriamente ou baseado em uma aproximação; inicialize  $V^{(0)}$  e calcule  $U^{(0)}$ . Defina o contador de iteração  $\alpha = 1$ . Selecione o número de centro de classes  $c$  e escolha o peso  $m$ .
2. Calcule os centros dos *clusters*. Tendo  $U^{(\alpha)}$ , calcule  $V^{(\alpha)}$  de acordo com a Equação 42.
3. Atualize os valores de pertinência. Tendo  $V^{(\alpha)}$ , calcule  $U^{(\alpha)}$  de acordo com a Equação 43.
4. Pare se

$$\max |u_{ik}^{(\alpha)} - u_{ik}^{(\alpha-1)}| \leq \varepsilon \quad (44)$$

senão faça  $\alpha = \alpha + 1$  e vá para o passo 2, onde  $\varepsilon$  é um número pequeno pré-especificado representando a menor mudança aceitável em  $U$ .

### 2.3.3 Comparação entre os algoritmos C-Means e Fuzzy C-Means

O algoritmo *hard* C-Means pode ser considerado como um caso especial do algoritmo FCM. Na Equação 38, se  $u_{ik}$  é 1 para apenas uma classe e zero para todas as outras, então a função critério  $J(U, V)$  usada no FCM é a mesma função critério  $J(V)$  usada no *hard* C-Means.

O exemplo abaixo mostra uma comparação entre os dois algoritmos aplicados ao problema de segmentação de uma imagem de documento. A Figura 26 foi usada na comparação entre os algoritmos (CHI; YAN; PHAM, 1996). A tarefa é segmentar a imagem, isto é, separar o fundo do primeiro plano da imagem. Esta imagem é difícil de lidar pois o fundo dela não é uniforme. Embora exista um vale no histograma da imagem, o que indica que existem dois grupos de *pixels* na imagem, o vale não é tão fundo, isto é, os dois grupos têm alto grau de sobreposição. Portanto, a segmentação é bastante sensível ao valor de limiar se um método de *thresholding* é usado para segmentar a imagem.

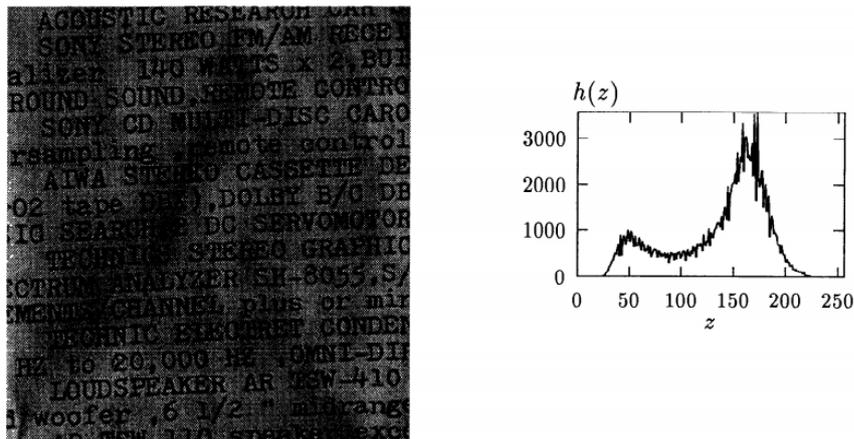


Figura 26: Uma imagem de documento e seu histograma de níveis de cinza (CHI; YAN; PHAM, 1996).

O método de *thresholding* tem algumas limitações devido ao fato de lidar apenas com o nível de cinza dos *pixels*. Para levar em conta propriedades geométricas da imagem, pode-se fazer uso de algumas características espaciais. Para fazer isso, no entanto, é preciso resolver um problema de classificação em um espaço de características multi-dimensional e o método de *thresholding* não pode ser usado.

Os métodos de agrupamento descritos neste Capítulo podem ser usados para resolver esse problema. Nos experimentos descritos em (CHI; YAN; PHAM, 1996), foram usadas sete características de cada *pixel*, o que inclui o nível de cinza do *pixel*, a média e o desvio padrão dos níveis de cinza em uma vizinhança 5 por 5 em torno do *pixel*, e as intensidades horizontal, vertical, e diagonal.

A Figura 27 mostra imagem resultado da segmentação usando o algoritmo C-Means. Geralmente, o algoritmo C-Means é melhor que o método de *thresholding*

e o resultado da segmentação é menos sensível a variações no fundo da imagem. Entretanto, os caracteres extraídos usando o algoritmo C-Means parecem estar borrados. Este problema pode ser resolvido se for utilizado o algoritmo FCM.

ACOUSTIC RESEARCH CAR  
 SONY STEREO FM/AM RECEI  
 alizer 140 WATTS x 2, BUI  
 ROUND SOUND, REMOTE CONTRO  
 SONY CD MULTI-DISC CARO  
 rsampling ,remote control  
 AIWA STEREO CASSETTE DE  
 02 tape DBX), DOLBY B/C DE  
 IC SEARCH, 2 DC SERVOMOTO  
 TECHNICS STEREO GRAPHIC  
 ISTRUM ANALYZER SH-8055, S/  
 MENTS/CHANNEL plus or mi  
 TECHNIC ELECTRET CONDEI  
 HZ to 20,000 HZ ,OMNI-DII  
 LOUDSPEAKER AR TSW-410  
 woofers ,6 1/2 " midrang  
 100 mw 110 speaker exci

Figura 27: Resultado da segmentação obtida usando o algoritmo C-Means (CHI; YAN; PHAM, 1996).

No FCM, cada *pixel* tem dois valores de pertinência, um representando o grau de certeza de um *pixel* pertencer ao fundo da imagem e o outro representando o grau de certeza de um *pixel* pertencer ao primeiro plano. Para a imagem de documento apenas o valor de pertinência sobre o fundo é utilizado, como mostra a Figura 28. O histograma da imagem também é mostrado na Figura 28. O histograma é bem similar ao da Figura 26 exceto pelo fato de que o vale é muito fundo e amplo.

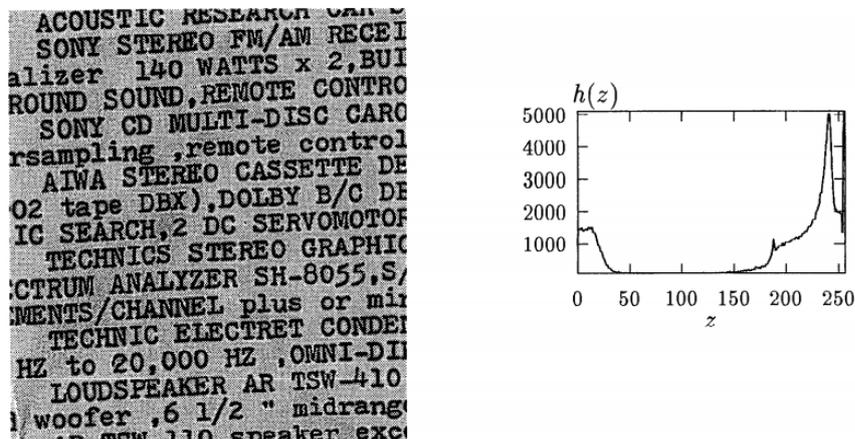


Figura 28: Valores de pertinência *fuzzy* mostrados como uma imagem e o histograma de níveis de cinza da imagem (CHI; YAN; PHAM, 1996).

Assim, segmenta-se a imagem de maneira mais confiável utilizando um método de *thresholding*. Com um limiar de 91, o resultado obtido é mostrado na Figura 29. Comparado com a imagem na Figura 27 obtida usando o algoritmo *hard C-Means*, a imagem segmentada na Figura 29 é levemente melhor e não tem *pixels* ruidosos isolados no fundo da imagem.

ACOUSTIC RESEARCH CAR  
 SONY STEREO FM/AM RECEI  
 alizer 140 WATTS x 2,BUI  
 ROUND SOUND,REMOTE CONTRO  
 SONY CD MULTI-DISC CARO  
 rsampling ,remote control  
 AIWA STEREO CASSETTE DE  
 02 tape DBX),DOLBY B/C DE  
 IC SEARCH,2 DC SERVOMOTO  
 TECHNICS STEREO GRAPHIC  
 ICTRUM ANALYZER SH-8055,S/  
 MENTS/CHANNEL plus or mix  
 TECHNIC ELECTRET CONDEI  
 HZ to 20,000 HZ ,OMNI-DI  
 LOUDSPEAKER AR TSW-410  
 i woofer ,6 1/2 " midrang  
 100 msw 110 speaker exci

Figura 29: Resultado da segmentação obtida a partir da Figura 28 com um valor de limiar de 91 (CHI; YAN; PHAM, 1996).

Quando o valor do limiar é diminuído para 42, o resultado é diferente e a imagem é demonstrada na Figura 30. Pode-se notar que a imagem não está mais borrada. Geralmente, pode-se mudar o valor do limiar para alterar a largura dos traços dos caracteres na imagem segmentada. Isto pode ser importante para alguns sistemas de reconhecimento óptico de caracteres (OCR).

## 2.3.4 Aplicação

### 2.3.4.1 Caracterização de tecidos em imagens de ressonância magnética

A caracterização de tecidos cerebrais em imagens de ressonância magnética é útil para diversos procedimentos de diagnóstico médico. A tarefa aqui é classificar os *pixels* da imagem em algumas categorias significativas. Isto pode ser feito usando o FCM.

No exemplo de aplicação descrito em CHI; YAN; PHAM (1996) foram consideradas duas imagens de ressonância magnética, mostradas à esquerda nas Figuras 31 e 32. Para cada *pixel*, foram extraídas sete características, que incluem a intensidade do *pixel*, a média local e a variância calculadas em uma vizinhança de 5 por 5 *pixels*,

ACOUSTIC RESEARCH CARBON  
 SONY STEREO FM/AM RECEI  
 alizer 140 WATTS x 2, BUI  
 ROUND SOUND, REMOTE CONTRO  
 SONY CD MULTI-DISC CARC  
 rsampling , remote control  
 AIWA STEREO CASSETTE DE  
 02 tape DBX), DOLBY B/C DE  
 IC SEARCH, 2 DC SERVOMOTO  
 TECHNICS STEREO GRAPHIC  
 CTROM ANALYZER SH-8055, S/  
 MENTS/CHANNEL plus or mix  
 TECHNIC ELECTRET CONDEI  
 HZ to 20,000 HZ , OMNI-DII  
 LOUDSPEAKER AR TSW-410  
 i woofer , 6 1/2 " midrang  
 AN MSW 110 speaker exci

Figura 30: Resultado da segmentação obtida a partir da Figura 28 com um valor de limiar de 42 (CHI; YAN; PHAM, 1996).

e valores de borda (*edge*) dados pelos operadores de Sobel nas quatro diferentes direções.

Para cada imagem o número de classes escolhido foi  $c = 4$ , correspondendo ao fundo da imagem, e três tipos de tecidos cerebrais, substância branca, massa cinzenta e líquido cefalorraquidiano. As imagens segmentadas são mostradas à direita nas Figuras 31 e 32, respectivamente. Na Figura 31, três tipos de tecidos cerebrais são separados. Na Figura 32, a área do tumor foi isolada dos tecidos a sua volta.

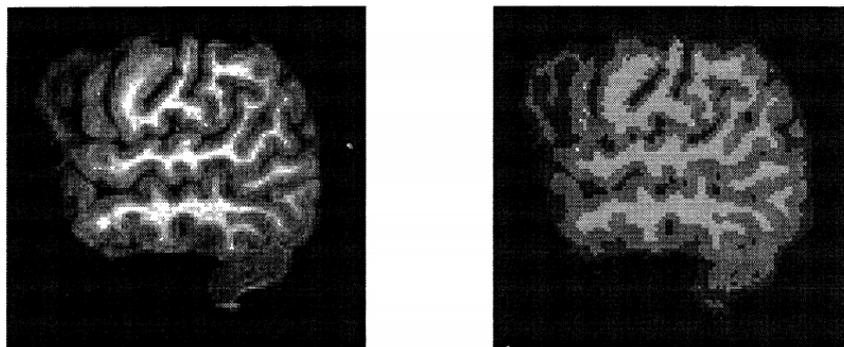


Figura 31: Uma imagem de ressonância magnética de tecidos cerebrais (esquerda) e o resultado de sua segmentação (direita) (CHI; YAN; PHAM, 1996).

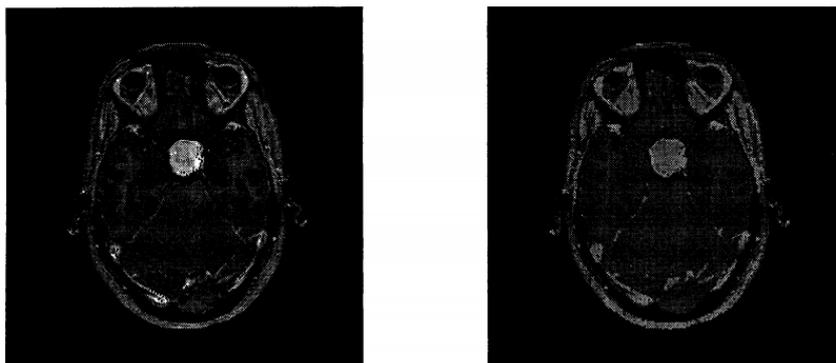


Figura 32: Uma imagem de ressonância magnética de um cérebro (esquerda) e o resultado da segmentação (direita). A área do tumor na imagem foi isolada dos tecidos. (CHI; YAN; PHAM, 1996)

## 2.4 Eletroforese Bidimensional

O proteoma indica as proteínas expressas em um organismo ou tecido. Enquanto o genoma representa a soma de todos os genes de um indivíduo, o proteoma não é uma característica fixa de um organismo. O proteoma altera com o estado de desenvolvimento do tecido ou mesmo sob as condições nas quais o indivíduo se encontra. Portanto, investigar diretamente os produtos dos genes é uma forma de estudar doenças e qualquer problema biológico complexo.

O termo proteoma foi introduzido em 1996 para descrever as proteínas expressas em um genoma (WILKINS et al., 1996; ANDERSON; ANDERSON, 1996). A proteômica pode ser vista como uma metodologia de seleção da biologia molecular, a qual tem como objetivo documentar a distribuição geral de proteínas da célula, identificar e caracterizar proteínas individuais de interesse e principalmente elucidar as suas associações e funções. Além disso, a proteômica é uma abordagem para identificar, quantificar e estudar as modificações pós-traducionais das proteínas em uma célula, tecido ou, mesmo, organismos (CIERO; BELLATO, 2002).

Por exemplo, para entender no nível molecular como uma célula funciona em um indivíduo doente e em um sadio é preciso ter conhecimento das proteínas e de outros componentes celulares que estão presentes, como eles interagem e o resultado de suas interações (CIERO; BELLATO, 2002). A análise ao nível de proteína é muito necessária, pois o estudo dos genes através do sequenciamento de DNA não pode adequadamente prever a estrutura dinâmica das proteínas, uma vez que é ao nível das proteínas que muitos dos processos de uma célula ocorrem, onde processos patológicos acontecem e aonde muitas das drogas atuam.

Proteínas de uma forma geral controlam a vida e a saúde, ou seja, é preciso entender as proteínas e como elas operam celularmente para assim entender como regular os mecanismos da doença para um posterior tratamento.

Em termos de aplicação, a proteômica está presente em diversas áreas de interesse, por exemplo, na investigação de marcadores moleculares em determinadas doenças indicando a resposta da célula ou tecido a estresses externos. Através da proteômica pode-se fazer uma comparação do perfil proteico de uma célula cancerosa com o perfil de uma célula sadia, ou de uma célula cancerosa cujo portador está sob tratamento médico, por exemplo, para entender a evolução deste tratamento (CIERO; BELLATO, 2002).

### **2.4.1 Funcionamento**

Embora o conceito de proteoma seja recente, utiliza técnicas de separação de proteínas desenvolvidas a bastante tempo, como a eletroforese bidimensional (O'FARRELL, 1975). Contudo, o uso da técnica era limitado devido à baixa reprodutibilidade dos géis e a falta de métodos sensíveis o bastante para identificar as proteínas separadas.

A eletroforese bidimensional baseia-se na migração de íons submetidos à corrente elétrica, de tal forma que moléculas com carga positiva migrem para o polo negativo, e moléculas com carga negativa, migrem para o polo positivo (O'FARRELL, 1975). A velocidade de migração depende de fatores como: tamanho, forma e carga elétrica das moléculas. O processo também sofre influência do tipo, concentração e pH do recipiente, e pela temperatura e força do campo elétrico. Em um experimento de eletroforese bidimensional em gel, as proteínas (ou outras moléculas) são separadas em duas dimensões, de modo que todas as proteínas/moléculas se espalhem por todo o gel.

#### *2.4.1.1 Preparação da amostra*

Antes de começar o processo de eletroforese se faz necessária a preparação da amostra, já que alguns componentes biológicos podem interferir na análise. Por exemplo, as substâncias insolúveis podem ser removidas por centrifugação. Ainda, é necessário remover sais antes da análises através de diálise, filtração de exclusão por tamanho, precipitação da proteína, ou cromatografia de fase reversa (BOGUTH et al., 2000; HIRSCH et al., 2004). Além disso, algumas amostras complexas precisam ser fracionadas a fim de facilitar a análise. A purificação por afinidade é uma abordagem poderosa para reduzir a complexidade de uma amostra por isolar especificamente proteínas individuais ou complexos de proteína (BOGUTH et al., 2000; HIRSCH et al., 2004).

#### *2.4.1.2 Processo de eletroforese bidimensional*

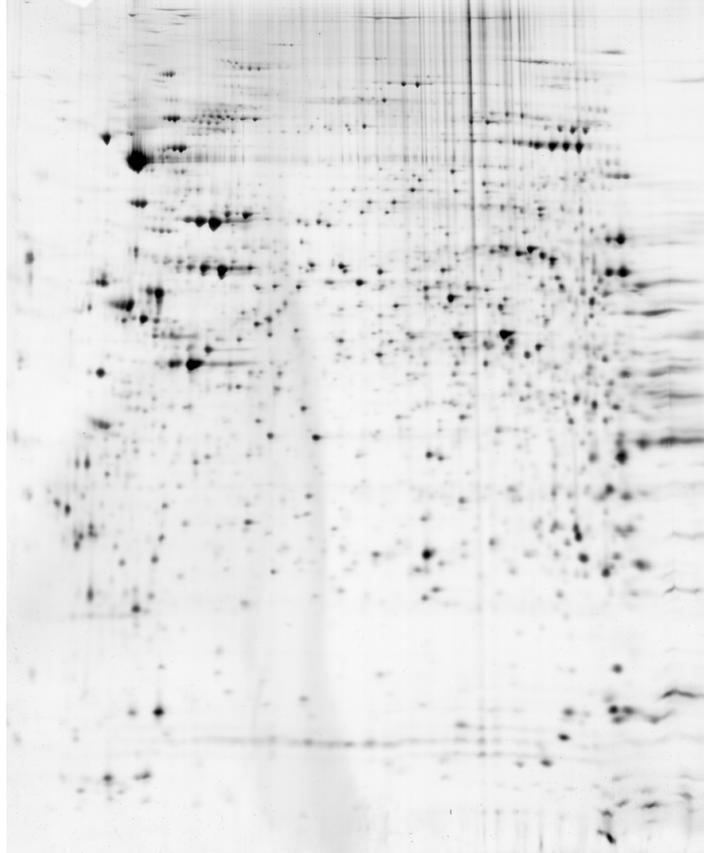
O processo de eletroforese bidimensional é dividido basicamente em duas etapas. Na primeira etapa do experimento, as proteínas são separadas com base nas

diferenças de ponto isoelétrico, sendo esta primeira etapa chamada de focalização isoelétrica (IEF). A amostra é aplicada em uma fita gelatinosa que possui um gradiente de pH e, em seguida, é submetida a um potencial elétrico. Assim, as proteínas migrarão até que o ponto isoelétrico seja alcançado, isto é, o ponto em que a carga global sobre a proteína é 0 (uma carga neutra). Neste ponto, a proteína para de migrar, ocorrendo então a primeira etapa de separação (CIERO; BELLATO, 2002).

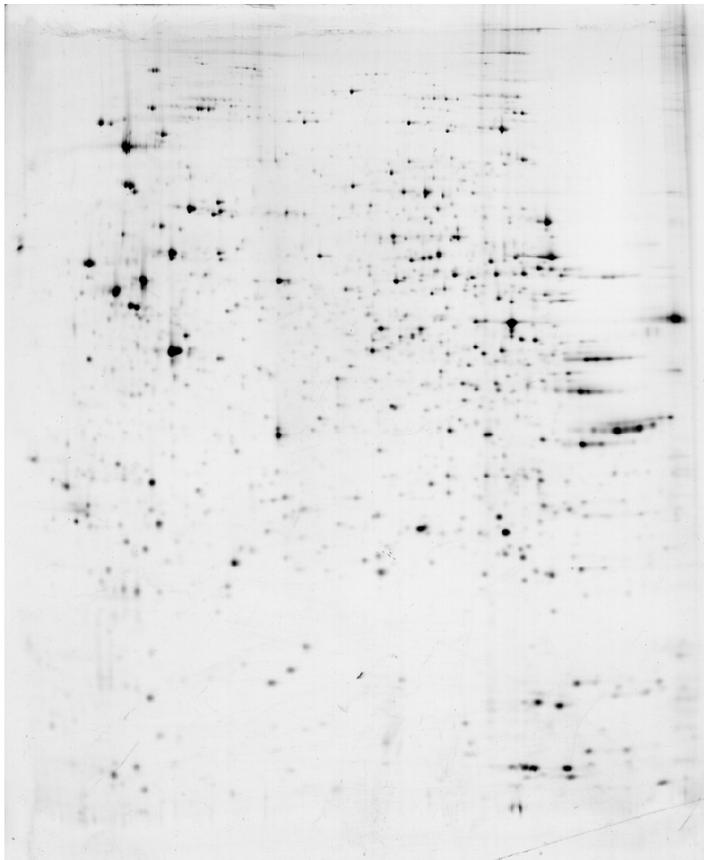
Na etapa seguinte, as proteínas são separadas pela diferença de massa molecular. A fita gelatinosa – contendo as proteínas separadas por diferença de ponto isoelétrico – é anexada a um segundo gel que permitirá a migração destas proteínas na segunda dimensão. Na segunda dimensão, um potencial elétrico é aplicado novamente, mas a um ângulo de 90 graus a partir do primeiro campo. Então, as proteínas são atraídas para o lado mais positivo do gel. O gel atua como uma peneira molecular e, portanto, quando a corrente é aplicada, separa as proteínas com base no seu peso molecular: as proteínas maiores migram vagarosamente pelo gel ficando retidas na peneira; e, as proteínas menores se locomovem mais rapidamente depositando-se na região mais baixa do gel (ROCHA; SILVA; SA, 2005). Como resultado deste processo, tem-se um gel com proteínas espalhadas sobre a sua superfície.

Estas proteínas podem ser detectadas por uma variedade de reagentes de revelação, sendo o nitrato de prata e o azul de Coomassie (*Coomassie Blue*) os mais utilizados. Após a coloração do gel, observa-se um perfil bidimensional de pontos (do inglês, *spots*), sendo que em cada ponto há múltiplas cópias de uma proteína. Por fim, este gel é escaneado e a imagem resultante pode ser processada (ROCHA; SILVA; SA, 2005). Essas imagens são amplamente utilizadas para extrair informações úteis a várias aplicações da biologia molecular. Porém, o processo manual de avaliação da imagem é subjetivo e propenso a erros, visto que depende da experiência do pesquisador responsável pela análise (PARK et al., 2012).

A Figura 33 mostra exemplos de imagens de géis oriundas de eletroforese bidimensional. Como pode ser observado, as imagens de eletroforese bidimensional podem conter ruídos, bem como partículas de poeira e, até mesmo, rachaduras no gel, e isso pode interferir no resultado final da análise de reconhecimento dos *spots* (SAVELONAS; MYLONA; MAROULIS, 2012). A Figura 34 mostra exemplos de imagens de géis oriundas de eletroforese bidimensional que ilustram esse problema. Pode ser observada a presença de uma quantidade considerável de ruído, na forma de borrões, o que torna mais complexa a tarefa de determinar os *spots*.



(a)

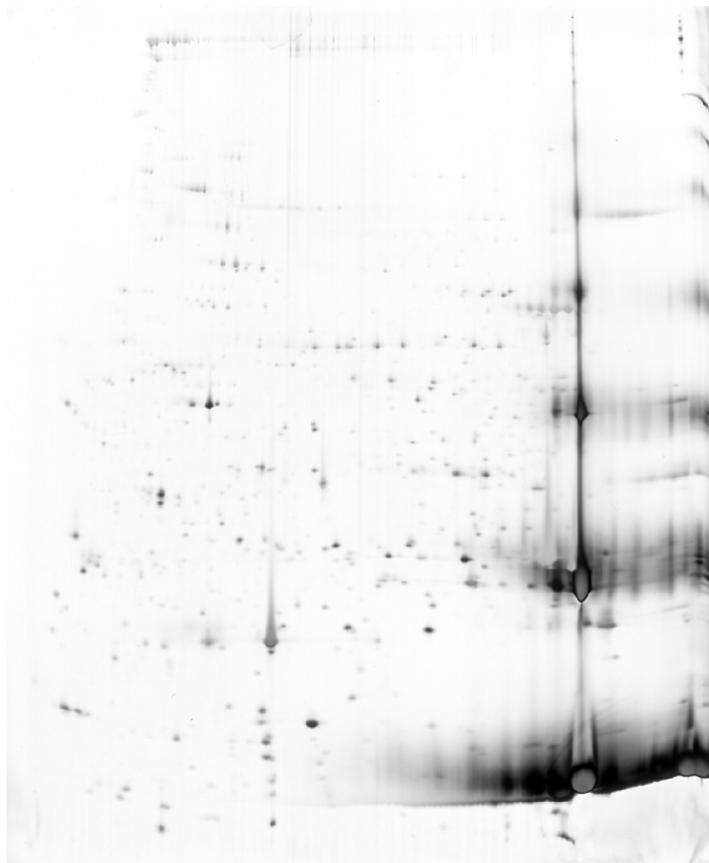


(b)

Figura 33: Imagens de géis oriundas de eletroforese bidimensional.



(a)



(b)

Figura 34: Imagens de géis com bastante ruído.

## 3 CLUSTERIZAÇÃO FUZZY APLICADO AO RECONHECIMENTO DE *SPOTS*

### 3.1 Trabalhos relacionados

Algumas abordagens têm sido usadas no processo de descoberta dos *spots* nas imagens de géis oriundas de eletroforese bidimensional. Por exemplo, em SAVELO-NAS; MYLONA; MAROULIS (2012) é utilizado um esquema de segmentação baseada em contorno. A técnica é dividida em 4 etapas: (a) detecção de *spots* múltiplos, capaz de identificar os limites em *spots* sobrepostos com base na observação de que esses limites estão associados com os mínimos de intensidade local; (b) equalização do histograma e reconstrução morfológica, de modo a evitar a amplificações indesejadas de ruído e facilitar a identificação de pontos fracos; (c) inicialização do contorno, visando formar uma superfície de nível definido, baseado na observação de que as manchas de proteína estão associadas com intensidade máximas regionais; e (d) evolução do contorno, com base em determinada região pela intensidade da imagem, bem como por informações derivadas de etapas anteriores. A técnica foi testada com imagens reais de géis de eletroforese bidimensional e os resultados obtidos foram comparados com três softwares comerciais: PDQuest 8.1, Melanie 7 e Delta2D. O autor afirma que os resultados foram melhores que os obtidos pelos softwares.

NATALE et al. (2012) utiliza uma abordagem de extrapolação gaussiana. A abordagem é dividida em duas fases. Primeiramente, revela áreas superexpostas na imagem, onde os *spots* podem estar truncados, e nas regiões de planalto causadas por manchas de *spots* sobrepostos. Segundo, faz uma distribuição em tons de cinza dos pontos saturados, reconstruindo a região com uma aproximação Gaussiana. Segundo NATALE et al. (2012), o método produz uma detecção altamente refinada e automática, sem a necessidade de correções manuais por um especialista. Ainda, afirma que a correção de *pixel* em locais saturados e manchadas de acordo com a curva de Gauss generalizada permite a quantificação mais precisa, proporcionando resultados mais confiáveis. Para validar o método proposto foram processadas imagens com *spots* saturados. Os resultados indicam que o método pode eficientemente

reconstruir *spots* saturados e melhorar a quantificação dos *spots*.

LI; SEILLIER-MOISEIWITSCH; KOROSTYSHEVSKIY (2011) propõem a utilização de análise estatística para realizar a segmentação nas imagens. Neste trabalho a imagem é dividida em diferentes regiões usando um algoritmo de segmentação chamado de *watershed*. Assim, em cada região, cada *pixel* é avaliado se é um objeto ou pertencente ao fundo da imagem, utilizando análises estatísticas. Para a validação do modelo proposto, utilizou-se géis simulados. Os resultados obtidos foram comparados aos resultados de dois softwares comerciais: PDQuest e Dymension.

KAZHIYUR-MANNAR et al. (2006) descreve um algoritmo baseado em contornos para separar o fundo do primeiro plano da imagem. O algoritmo pode ser visto como um *thresholding* local ou como um algoritmo de *watershed*. A característica mais importante é que ele utiliza a área para determinar o objeto a ser segmentado. Assim, é relativamente imune a variações de brilho e contraste. A técnica foi desenvolvida especificamente para a análise de géis de eletroforese bidimensional, embora possa ser aplicada também a outras imagens em tons de cinza. O algoritmo foi comparado ao software ImageMaster e mostrou melhores resultados.

RODRIGUEZ et al. (2014) propõe uma técnica interativa baseada em *block-matching* (comum em tarefas de estimar deslocamento em imagens médicas) baseada em deformações sucessivas, busca, filtragem, a fim de medir deslocamentos em imagens de eletroforese bidimensional. A técnica utiliza diferentes modelos de deformação na tarefa de correlacionar proteínas em imagens reais de géis, obtendo uma precisão de 96,6% e melhorando os resultados obtidos com outras técnicas.

DIAS (2015) explora técnicas de aprendizado de máquina no reconhecimento de *spots*. O trabalho propõe um modelo baseado em aprendizado de máquina para o reconhecimento de *spots* em imagens de géis de eletroforese bidimensional. Os autores utilizam a base de dados apresentada em HOOGLAND et al. (2004). A partir desta base, foi criada uma base de treino para algoritmos de aprendizado de máquina, e as imagens foram utilizadas para testar os algoritmos. Com base nos testes realizados, árvores de decisão e *support vector machine* apresentaram os melhores resultados na avaliação do *recall*, e redes bayesianas na avaliação da métrica  $F_1$ . Este trabalho foi utilizado como referência para comparação do modelo proposto nesta dissertação.

### 3.2 Base de imagens

O Instituto Suíço de Bioinformática (SIB) mantém uma base de dados de imagens de eletroforese bidimensional já classificadas, chamado de SWISS-2DPAGE (HOOGLAND et al., 2004). Todas as imagens providas pelo SWISS-2DPAGE foram testadas e validadas por especialistas. Na sua versão mais atual, lançada em maio de 2011, o SWISS-2DPAGE conta com 1265 *spots* identificados, dos quais há ma-

peamento de 36 exames de eletroforese bidimensional. O SWISS-2DPAGE reúne informação de proteínas identificadas, onde cada entrada no SWISS-2DPAGE possui informação de uma proteína, incluindo em que processo de eletroforese ele foi encontrado, informações fisiológicas e patológicas, dados experimentais (como ponto isoelétrico, peso molecular, composição de aminoácidos e massa peptídica) e referencial bibliográfico. Além disso, foi calculada a localização aproximada de cada *spot* nas imagens, como exemplificado na Figura 35, onde os *pixels* em verde são as coordenadas dos *spots*.

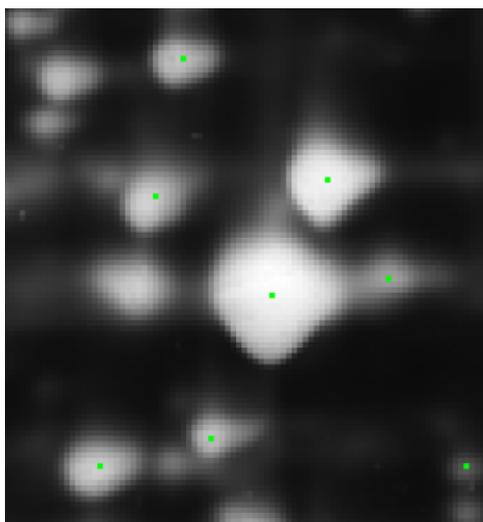


Figura 35: Representação dos *spots* indicados pelo SWISS-2DPAGE.

As imagens fornecidas pelo SWISS-2DPAGE foram criadas no formato TIFF (*Tagged image file format*). Este formato de arquivo é capaz de armazenar uma grande variedade de diferentes formas de dados de imagem e são bastante usadas para representar um valor de medida que não simplesmente intensidade ou a cor como parte de uma imagem científica ou médica. Imagens de pontos flutuantes são normalmente armazenados no formato de imagem TIFF (SOLOMON C.; BRECKON, 2011). Além disso, é possível utilizar o TIFF como base para a criação de um formato específico de domínio mais especializado, como por exemplo as imagens médicas DICOM ou MEL, utilizada pelo SWISS-2DPAGE.

A base do SWISS-2DPAGE conta com 37 imagens de eletroforese. Estas seguem o padrão TIFF de 16 bits. As imagens são apresentadas em tons de cinza e, por utilizarem 16 bits, podem atingir tonalidades de 0 a 65536. Mesmo tendo toda essa faixa de valores, nem todas fazem uso desta, e cada uma das imagens usa uma faixa de cores diferente. Por exemplo, as imagens ARABIDOPSIS, ELC.HUMAN, e STAPHYLOCOCCUS têm os tons de cinza variando entre 55 e 3043, 33 e 2319 e 7697 e 65535, respectivamente.

Além da faixa de tons de cinza diferentes, as imagens também possuem tamanhos

variados. Grande parte das imagens possui dimensão, *altura*  $\times$  *largura*, por volta de  $1000 \times 1000$ . O tamanho da imagem é relevante pois a maioria das técnicas aplicadas são sensíveis a essa característica, já que necessitam ser aplicadas em cada *pixel* da imagem.

Na Tabela 1 são apresentadas as informações sobre cada imagem disponibilizada pelo SWISS-2DPAGE. Cada uma das linhas consta o nome da imagem relacionada. As primeira e segunda colunas apresentam os tons de cinza mínimo e máximo, respectivamente. A quarta coluna apresenta as dimensões da imagem.

Tabela 1: Informações das imagens disponibilizadas pelo SWISS-2DPAGE.

<b>Nome da Imagem</b>	<b>Tom de Cinza Mínimo</b>	<b>Tom de Cinza Máximo</b>	<b>Dimensão</b> ( <i>largura</i> $\times$ <i>altura</i> )
ARABIDOPSIS	55	3043	932 $\times$ 1052
BAT_MOUSE	0	3000	1001 $\times$ 1062
CEC_HUMAN	25	4008	992 $\times$ 1073
CSF_HUMAN	1	2404	1003 $\times$ 1118
DICTYSLUG	0	2273	1004 $\times$ 1109
DLD1_HUMAN	11	3189	1000 $\times$ 1052
ECOLI	0	2361	998 $\times$ 1087
ELC_HUMAN	33	2319	881 $\times$ 1076
HEPG2SP_HUMAN	0	2876	999 $\times$ 1051
HEPG2_HUMAN	0	2886	1009 $\times$ 1029
HL60_HUMAN	0	2438	979 $\times$ 1073
ISLETS_MOUSE	33	3176	973 $\times$ 941
KIDNEY_HUMAN	0	2298	999 $\times$ 1135
LIVER_HUMAN	0	3011	1066 $\times$ 1158
LIVER_MOUSE	0	3143	1002 $\times$ 1104
LYMPHOCYTE_HUMAN	0	228	1000 $\times$ 1000
LYMPHOMA_HUMAN	0	2773	864 $\times$ 1029
MUSCLE_MOUSE	6	2804	1005 $\times$ 1105
NUCLEI_LIVER_HUMAN	0	3507	990 $\times$ 1070
NUCLEI_LIVER_MOUSE	4	3822	958 $\times$ 1103
NUCLEOLI_HELA_1D_HUMAN	0	254	325 $\times$ 1294
NUCLEOLI_HELA_2D_HUMAN	3	255	960 $\times$ 910
PLASMA_HUMAN	35	3565	958 $\times$ 974
PLATELET_HUMAN	40	2857	971 $\times$ 1088
RBC_HUMAN	0	3972	1022 $\times$ 1150
STAPHYLOCOCCUS	7697	65535	1075 $\times$ 1114
U937_HUMAN	44	1950	963 $\times$ 1115
WAT_MOUSE	27	3138	975 $\times$ 997
YEAST	10	2442	989 $\times$ 1086

### 3.3 Modelo de Clusterização Fuzzy

O modelo proposto tem por objetivo utilizar a clusterização *fuzzy* a fim de detectar os *spots* nas imagens de eletroforese bidimensional. Busca-se agrupar os *pixels* da imagem, de acordo com uma medida de similaridade.

A análise é feita usando imagens resultantes do processo de eletroforese bidimensional. Destas, utiliza-se as informações dos tons de cinza dos *pixels* que as compõem. Como visto anteriormente na Tabela 1, a escala de valores dos tons varia de uma imagem para outra, o que *a priori* foi considerado irrelevante.

Espera-se que os *spots*, em cada imagem, tenham tons de cinza similares e, baseando-se nos princípios da clusterização, é esperado que os *spots* estejam todos no mesmo *cluster* ao final do processo.



Figura 36: Fluxograma do modelo *fuzzy* proposto.

O modelo proposto neste trabalho pode ser dividido em 3 partes (ver Figura 36):

**Clusterização:** iniciada ao ler a imagem a ser analisada. Nesta etapa é aplicado o algoritmo Fuzzy C-Means. Portanto, faz-se necessário escolher os parâmetros do deste algoritmo, a saber, número de *clusters*, o valor de  $m$ ,  $\epsilon$  (menor mudança aceitável nos *clusters*) e o critério de parada. A imagem lida é passada na forma de matriz ao algoritmo, onde o conteúdo de cada elemento desta matriz é o valor do tom de cinza referente àquele *pixel* na imagem.

O processo de clusterização está ilustrado na Figura 37.

---

```

1: função CLUSTERIZACAO(path, c, m, epsilon, stop)
2:   matriz ← load_image(path);
3:   retorna fcmeans(matriz, c, m, epsilon, stop);
4: fim função
  
```

---

Figura 37: Pseudocódigo da etapa de clusterização.

O processo de clusterização tem como resultado uma matriz de pertinências, onde o número de colunas é o número de *pixels* da imagem e o número de linhas é o número de *clusters* escolhido. Portanto, cada posição  $ij$  dessa matriz representa a pertinência do *pixel*  $i$  em relação ao *cluster*  $j$ .

**Classificação:** Para fazer a classificação dos *pixels*, utiliza-se uma função máxima. Esta função é aplicada a cada coluna da matriz, onde o *pixel* respectivo é

atribuído ao *cluster* que ele tem maior valor de pertinência. Os valores aos quais os *pixels* vão sendo atribuídos são salvos em uma outra matriz com dimensões iguais as da imagem analisada. O processo de classificação pode ser observado em pseudocódigo na Figura 38. Assim que os *pixels* são classificados, faz-se necessário contar os *spots* encontrados.

---

```

1: função CLASSIFICACAO(matriz, x, y)
2:   novalmagem[][];
3:   lin;
4:   col;
5:   para i ← 1 até l faça
6:     maior ← 0;
7:     cluster ← 0;
8:     para j ← 1 até c faça
9:       se matriz[i][j] > maior então
10:        maior ← matriz[i][j];
11:        cluster ← j;
12:      fim se
13:    fim para
14:    calculaPosicao(&lin, &col, x, y, i);
15:    novalmagem[lin][col] ← cluster;
16:  fim para
17:  retorna novalmagem;
18: fim função

```

---

Figura 38: Pseudocódigo da etapa de classificação.

**Contagem dos spots:** Não é trivial contar os *spots*. Como visto anteriormente, os *pixels* são agrupados em *clusters*. Após a classificação, um destes *clusters* contém todos os *pixels* que fazem parte dos *spots*. Devemos, então, encontrar áreas na imagem que contenham *pixels* vizinhos classificados na etapa anterior como pertencendo ao *cluster* dos *spots*.

Para encontrar conjuntos de *pixel* classificados como *spots* é usada uma adaptação do algoritmo *Flood Fill*, descrito na Seção 2.1.5. Optou-se aqui por utilizar o *Flood Fill* com a vizinhança B8.

A técnica de contagem é aplicada a cada *pixel* da imagem. O processo de contagem começa por tomar como ponto inicial um ponto classificado como *spot*, e a partir deste ponto verifica-se a sua vizinhança recursivamente. Todos os pontos na vizinhança B8 que foram classificados como *spot* serão verificados. Cada ponto visitado é marcado como *visitado*, para evitar que algum *spot* seja

computado mais de uma vez. Este processo se repete até que não exista mais nenhum vizinho *spot* não visitado.

Assim, com todos os pontos verificados, aqueles vizinhos que foram classificados como *spots*, são considerados como parte de **um** *spot*. A Figura 39 mostra o processo de contagem em pseudocódigo.

---

```

1: função CONTAGEM(matriz, l, c)
2:   visitado[][];
3:   numSpots ← 0;
4:   para i ← 1 até l faça
5:     para j ← 1 até c faça
6:       se visitado[i][j] = 0 and matriz[i][j] = spot então
7:         verificaVizinhanca(matriz, &visitado, l, c);
8:         numSpots ← numSpots + 1;
9:       fim se
10:    fim para
11:  fim para
12:  retorna numSpots;
13: fim função

```

---

Figura 39: Pseudocódigo da etapa de contagem dos *spots*.

A Figura 40 apresenta o resultado do processo de contagem de *spots* na Figura 35, onde as áreas em azul representam os *spots* encontrados.

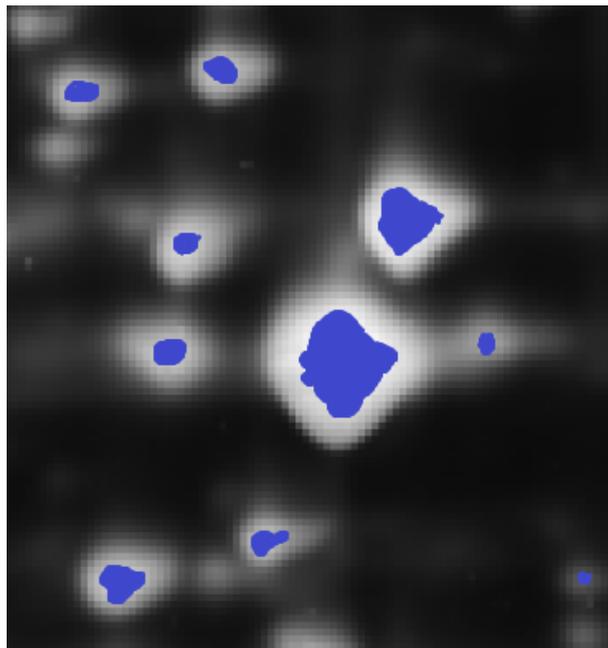


Figura 40: *Spots* encontrados utilizando o algoritmo de contagem.

---

```

1: ler(path);
2: ler(c);
3: ler(m);
4: ler(epsilon);
5: ler(stop);
6: pertinencias[][] ← clusterizacao(path, c, m, epsilon, stop);
7: novalmagem[][] ← classificacao(pertinencias);
8: totalSpots ← contagem(novalmagem);

```

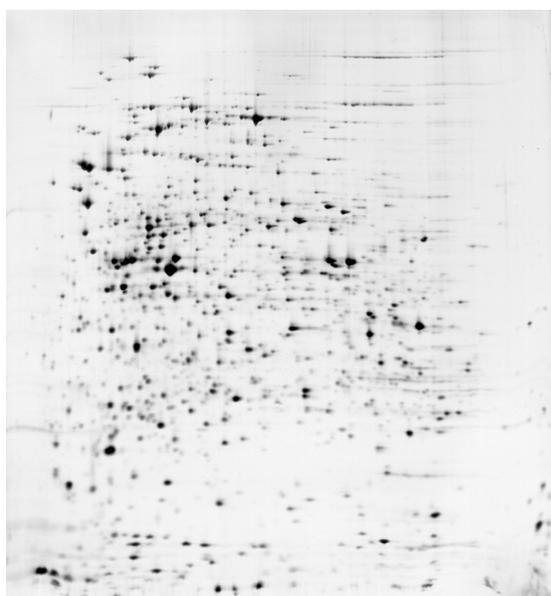
---

Figura 41: Pseudocódigo do modelo *fuzzy* para detecção dos *spots*.

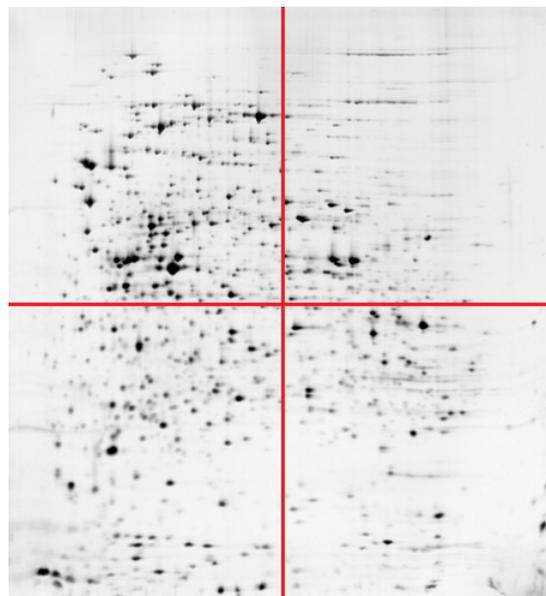
O modelo completo em pseudocódigo pode ser observado na Figura 41.

As imagens de eletroforese bidimensional contém muita informação de fundo. Na clusterização esses *pixels* acabam influenciando nos valores de centróides e, por conseguinte, na classificação feita. Ao desprezar estas áreas, ou utilizar menos áreas de fundo, a clusterização terá um resultado diferente resultando em *clusters* mais específicos. Por isso, uma variação deste modelo *fuzzy* também foi proposta, como alternativa. Em vez de realizar a clusterização a partir de toda a imagem, faz-se clusterizações separadamente em cada quadrante da imagem.

Para este trabalho, as imagens foram divididas nos quatro quadrantes antes da clusterização. A Figura 42(b) ilustra o particionamento da Figura 42(a).



(a) Imagem original - ECOLI



(b) Imagem particionada

Figura 42: Particionamento da imagem.

A Figura 43 mostra o fluxograma do método utilizando a imagem particionada.



Figura 43: Fluxograma do modelo *fuzzy* proposto para a imagem particionada em quatro quadrantes.

## 4 RESULTADOS

Neste Capítulo serão apresentados e discutidos os resultados obtidos da detecção de *spots* a partir das imagens fornecidas pelo SWISS-2DPAGE utilizando clusterização *fuzzy*. Os resultados deste trabalho serão comparados a um outro algoritmo de clusterização, o K-Means, e a uma técnica de aprendizado supervisionado (utilizado por DIAS (2015)), o *Gaussian Naïve-Bayes*. Também serão apresentados os resultados dos testes realizados com o modelo *fuzzy* proposto e as métricas utilizadas para avaliação dos resultados.

### 4.1 Métricas

Para realizar a avaliação dos resultados apresentados neste trabalho foram escolhidas métricas usuais em problemas de classificação: *recall*, precisão e  $F_1$ .

**Recall:** mede a taxa de *spots* positivos verdadeiros encontrados pelo modelo. Em MANNING; RAGHAVAN; SCHUTZE (2008) o *Recall* é formalizado da seguinte maneira:

$$Recall = \frac{\textit{itens relevantes encontrados}}{\textit{total de itens relevantes}} \quad (45)$$

Para este problema, itens relevantes são os *spots* contidos nas imagens. Portanto, o *Recall* pode ser definido como sendo:

$$Recall = \frac{\textit{spots confirmados encontrados}}{\textit{total de spots confirmados}} \quad (46)$$

**Precisão:** mede o nível de acerto na classificação dado o total itens relevantes e o total de itens classificados. Em MANNING; RAGHAVAN; SCHUTZE (2008) Precisão é formalizada assim:

$$Precisão = \frac{\textit{itens relevantes encontrados}}{\textit{total de itens encontrados}} \quad (47)$$

Para o problema da detecção de *spots*, a Precisão pode ser definida como:

$$Precisão = \frac{\text{spots confirmados encontrados}}{\text{total de spots encontrados}} \quad (48)$$

$F_1$ : é uma média harmônica entre o *Recall* e a precisão. Ela mede a eficiência levando em consideração o erro nas duas classes. É necessário que o acerto nas duas classes aumente para que a métrica aumente. Em MANNING; RAGHAVAN; SCHUTZE (2008) a métrica  $F_1$  é definida assim:

$$F_1 = \frac{2 \times \text{precisão} \times \text{recall}}{\text{precisão} + \text{recall}} \quad (49)$$

## 4.2 Definição dos Parâmetros

Os testes do modelo foram conduzidos utilizando o *toolbox* de lógica *fuzzy* para a linguagem *Python* chamado *sckit-fuzzy* (SCIKIT-FUZZY, 2015). No algoritmo Fuzzy C-Means existem vários parâmetros que podem ser configurados. Por exemplo:

- Número de *clusters*;
- $m$ , o fator de peso expoente da Eq. 38;
- $\varepsilon$ , menor mudança aceitável nos *clusters* da Eq. 44; e,
- Critério de parada: o algoritmo pára quando satisfaz a Eq. 44 ou quando o número máximo de iterações é atingido.

Por conta dos erros detectados na informação da localização exata dos *spots* confirmados na base do SWISS-2DPAGE, utilizou-se a informação original adicionando-se um raio extra de 10 *pixels* aos *spots* encontrados, onde todos os pontos nessa área podem ser considerados *spots* confirmados pelo SWISS-2DPAGE.

O modelo foi testado utilizando diferentes configurações dos parâmetros listados acima. Durante esta análise preliminar, foram identificados que parâmetros significativos para o Fuzzy C-Means foram o número de *clusters* e o  $m$ . Pode-se verificar na Tabela 2 os resultados dos testes do modelo em uma imagem fornecida pelo Laboratório de Bioinformática da Universidade Federal de Pelotas.

A partir destes testes foi possível observar que os valores de *recall* eram melhores para 3 *clusters*. Conforme o valor de  $m$  aumenta, o modelo passa a encontrar mais *spots* confirmados, porém, a precisão diminui, devido ao aumento dos falsos positivos.

Assim, optou-se por utilizar nos testes seguintes três *clusters*, que intuitivamente representam: (a) fundo da imagem; (b) rastro ou ruído na imagem; e, (c) os *spots* propriamente ditos.

Tabela 2: Testes executados em uma imagem fornecida pelo Laboratório de Bioinformática da Universidade Federal de Pelotas.

<b>Clusters</b>	<b><i>m</i></b>	<b>Spots encontrados</b>	<b>Confirmados faltando</b>	<b>Confirmados encontrados</b>	<b>Recall</b>	<b>Precisão</b>	<b>F<sub>1</sub></b>
3	2	279	55	151	0,7330	0,5412	0,6227
3	3	371	29	177	0,8592	0,4771	0,6135
3	4	419	25	181	0,8786	0,4320	0,5792
3	5	448	25	181	0,8786	0,4040	0,5535
3	10	534	16	190	0,9223	0,3558	0,5135
4	2	225	69	137	0,6650	0,6089	0,6357
4	3	260	57	149	0,7233	0,5731	0,6395
4	4	282	55	151	0,7330	0,5355	0,6189
4	5	325	38	168	0,8155	0,5169	0,6328
4	10	428	25	181	0,8786	0,4229	0,5710
5	2	175	93	113	0,5485	0,6457	0,5932
5	3	203	79	127	0,6165	0,6256	0,6210
5	4	230	68	138	0,6699	0,6000	0,6330
5	5	260	57	149	0,7233	0,5731	0,6395
5	10	338	37	169	0,8204	0,5000	0,6213
6	2	130	113	93	0,4515	0,7154	0,5536
6	3	165	98	108	0,5243	0,6545	0,5822
6	4	190	87	119	0,5777	0,6263	0,6010
6	5	216	71	135	0,6553	0,6250	0,6398
6	10	300	49	157	0,7621	0,5233	0,6206

### 4.3 Análise do Fuzzy C-Means com a Imagem Inteira

Com base nos testes preliminares, os valores padrão dos parâmetros do Fuzzy C-Means foram assumidos para os testes com as imagens do SWISS-2DPAGE. Estes são:

- $m = 2,0$ ;
- $\varepsilon = 1 \times 10^{-5}$ ;
- $maxiter = 100.000$

A Figura 44 mostra um teste realizado em uma imagem do SWISS-2DPAGE utilizando a configuração citada anteriormente.

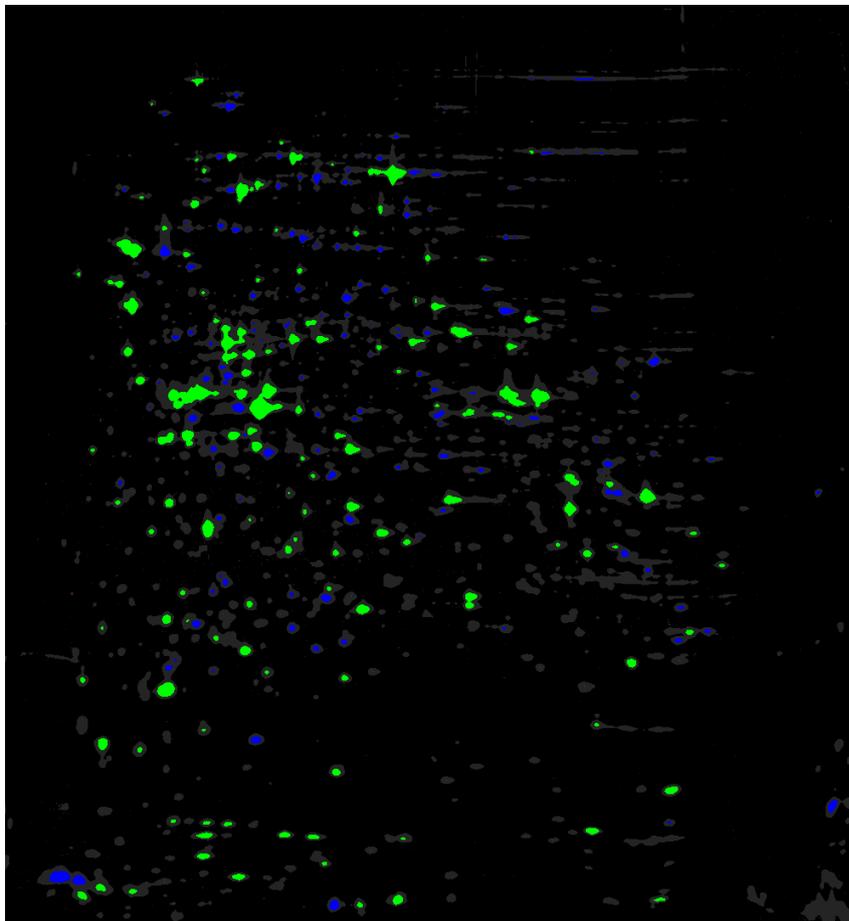


Figura 44: Clusterização resultante da Figura 42(a)

A Figura 42(a) é a imagem original, chamada ECOLI. A Figura 44 mostra classificação feita pelo algoritmo. Para melhor visualização e entendimento, pode-se observar o recorte da Figura 44 na Figura 45. Nesta, é possível observar cinco cores, sendo elas:

- Preto - representa o fundo da imagem (*cluster um*);

- Cinza - representa os rastros e ruídos na imagem (*cluster dois*);
- Azul - são os *spots* encontrados pelo algoritmo, mas que não foram confirmados pelo SWISS-2DPAGE (*cluster três*);
- Verde - são os *spots* encontrados pelo algoritmo, e confirmados pelo SWISS-2DPAGE (*cluster três*);
- Vermelho - a localização dos *spots* confirmados pelo SWISS-2DPAGE e que não foram encontrados pelo algoritmo.

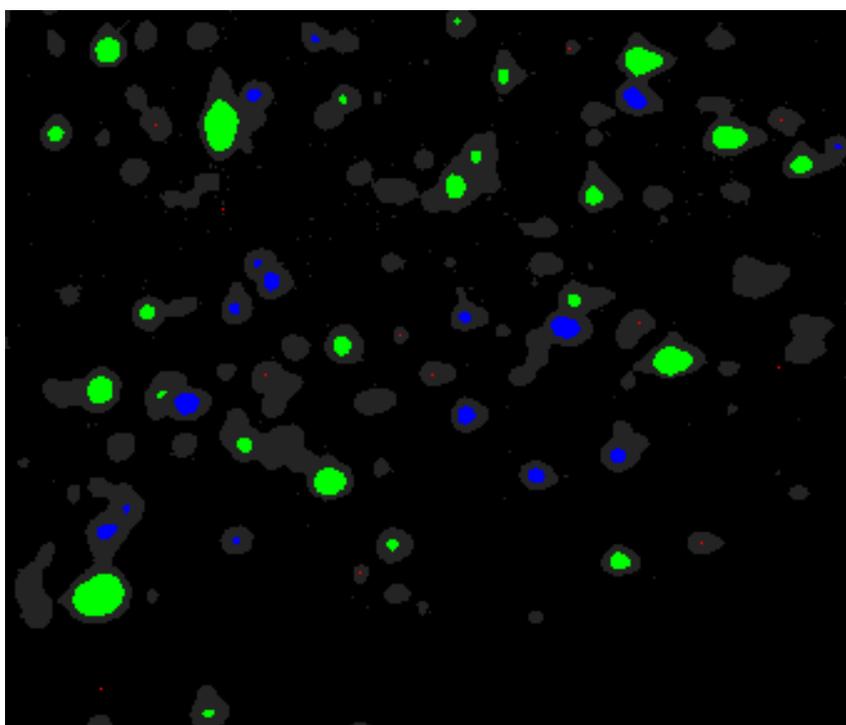


Figura 45: Recorte da Figura 44.

O *cluster três* representa os pontos considerados *spots* pelo modelo, tanto os pontos em azul como os em verde. A diferença entre estes pontos, é que os pontos em verde foram *spots* confirmados por HOOGLAND et al. (2004). Já os pontos em azul são pontos encontrados pelo modelo, que não estão na base em HOOGLAND et al. (2004), mas isso não significa que eles não possam ser *spots*. São os chamados *falsos positivos*. Na verdade, eles representam áreas que podem ser *spots* e que podem ser investigadas por um especialista para fazer a confirmação se são ou não *spots*.

Ainda, existem *spots* confirmados em base que não foram encontrados pelo modelo. Ao todo, na Figura 42(a) existem 206 *spots* confirmados. O modelo proposto encontrou 279 *spots*, aos quais, 179 são confirmados e 55 confirmados não foram encontrados, resultando assim em um *recall* de 73,3%. O melhor resultado foi na

imagem NUCLEI.LIVER.MOUSE onde apenas 1 (um) *spot* confirmado não foi encontrado, acarretando assim um *recall* de 93,33%. A pior execução foi na imagem RBC\_HUMAN, onde o algoritmo modelo encontrou 63 dos 190 (33,16%) *spots* confirmados. Com base na métrica  $F_1$ , a melhor execução foi na imagem STAPHYLOCOCCUS, onde o valor foi 0,7775, e a pior execução foi na imagem DICTYSLUG com o valor de  $F_1$  sendo 0,0849.

A Tabela 3 mostra os resultados dos testes do FCM em todas as imagens providas pelo SWISS-2DPAGE.

Tabela 3: Resultado da aplicação do Fuzzy C-Means nas imagens do SWISS-2DPAGE.

Nome da imagem	Spots encontrados	Confirmados faltando	Confirmados encontrados	Recall	Precisão	F1
ARABIDOPSIS	149	21	58	0,7342	0,3893	0,5088
BAT MOUSE	203	27	58	0,6824	0,2857	0,4028
CEC HUMAN	358	13	43	0,7679	0,1201	0,2077
CSF HUMAN	198	149	160	0,5178	0,8081	0,6312
DICTYSLUG	517	2	23	0,9200	0,0445	0,0849
DLD1 HUMAN	480	17	91	0,8426	0,1896	0,3095
ECOLI	279	55	151	0,7330	0,5412	0,6227
ELC HUMAN	356	4	31	0,8857	0,0871	0,1586
HEPG2 HUMAN	461	26	72	0,7347	0,1562	0,2576
HEPG2SP HUMAN	113	63	92	0,5935	0,8142	0,6866
HL60 HUMAN	353	6	20	0,7692	0,0567	0,1055
ISLETS MOUSE	262	21	42	0,6667	0,1603	0,2585
KIDNEY HUMAN	265	11	33	0,7500	0,1245	0,2136
LIVER HUMAN	220	44	94	0,6812	0,4273	0,5251
LIVER MOUSE	281	54	146	0,7300	0,5196	0,6071
LYMPHOCYTE HUMAN	181	39	38	0,4935	0,2099	0,2946
LYMPHOMA HUMAN	214	22	38	0,6333	0,1776	0,2774
MUSCLE MOUSE	197	48	88	0,6471	0,4467	0,5285
NUCLEI LIVER HUMAN	143	2	25	0,9259	0,1748	0,2941
NUCLEI LIVER MOUSE	386	1	14	0,9333	0,0363	0,0698
NUCLEOLI HELA 1D HUMAN	465	19	79	0,8061	0,1699	0,2806
NUCLEOLI HELA 2D HUMAN	55	23	28	0,5490	0,5091	0,5283
PLASMA HUMAN	429	317	309	0,4936	0,7203	0,5858
PLATELET HUMAN	337	3	41	0,9318	0,1217	0,2152
RBC HUMAN	139	127	63	0,3316	0,4532	0,3830
STAPHYLOCOCCUS	195	77	173	0,6920	0,8872	0,7775
U937 HUMAN	78	20	22	0,5238	0,2821	0,3667
WAT MOUSE	457	22	114	0,8382	0,2495	0,3845
YEAST	149	40	61	0,6040	0,4094	0,4880
			<b>Média</b>	0,7300	0,2495	0,3667
			<b>Desvio Padrão</b>	0,1519	0,2460	0,1912

A partir das imagens geradas após a execução do algoritmo, foi possível observar que alguns *spots* confirmados têm tons de cinza distintos dos demais. Enquanto a maioria dos *spots* têm valores bem altos, outros *spots*, no entanto, têm valores bem baixos. Este é, sem dúvidas, um grande desafio no processo de clusterização, pois a clusterização é uma estratégia que procura os *spots* com características similares,

e foi-se observado que existem *spots* com uma variação considerável no seu tom de cinza.

#### 4.4 Análise do Fuzzy C-Means dividindo-se a Imagem em Quadrantes

O mesmo experimento descrito anteriormente foi realizado com a imagem dividida em quatro quadrantes e cada quadrante foi individualmente clusterizado. Para a mesma Figura 42(a), 15 novos *spots* foram encontrados. Destes, 14 estavam na lista de confirmados, o que resultou num *recall* de 80,01%. A Tabela 4 mostra os resultados dos testes do FCM fazendo a clusterização na imagem dividida em partes menores.

Tabela 4: Resultado da execução do Fuzzy C-Means nas imagens do SWISS-2DPAGE, utilizando a clusterização por partes.

Nome da imagem	Spots encontrados	Confirmados faltando	Confirmados encontrados	Recall	Precisão	F1
ARABIDOPSIS	149	21	58	0,7342	0,3893	0,5088
BAT MOUSE	206	28	57	0,6706	0,2767	0,3918
CEC HUMAN	415	11	45	0,8036	0,1084	0,1911
CSF HUMAN	220	158	151	0,4887	0,6864	0,5709
DICTYSLUG	518	2	25	0,9259	0,0483	0,0917
DLD1 HUMAN	531	14	94	0,8704	0,1770	0,2942
ECOLI	294	41	165	0,8010	0,5612	0,6600
ELC HUMAN	331	3	32	0,9143	0,0967	0,1749
HEPG2 HUMAN	466	24	74	0,7551	0,1588	0,2624
HEPG2SP HUMAN	133	72	83	0,5355	0,6241	0,5764
HL60 HUMAN	331	5	21	0,8077	0,0634	0,1176
ISLETS MOUSE	274	21	42	0,6667	0,1533	0,2493
KIDNEY HUMAN	298	9	35	0,7955	0,1174	0,2047
LIVER HUMAN	274	30	108	0,7826	0,3942	0,5243
LIVER MOUSE	306	48	152	0,7600	0,4967	0,6008
LYMPHOCYTE HUMAN	178	34	43	0,5584	0,2416	0,3373
LYMPHOMA HUMAN	271	15	45	0,7500	0,1661	0,2719
MUSCLE MOUSE	182	51	85	0,6250	0,4670	0,5346
NUCLEI LIVER HUMAN	317	1	26	0,9630	0,0820	0,1512
NUCLEI LIVER MOUSE	386	1	14	0,9333	0,0363	0,0698
NUCLEOLI HELA 1D HUMAN	514	16	82	0,8367	0,1595	0,2680
NUCLEOLI HELA 2D HUMAN	58	22	29	0,5686	0,5000	0,5321
PLASMA HUMAN	521	301	325	0,5192	0,6238	0,5667
PLATELET HUMAN	335	2	39	0,9512	0,1164	0,2074
RBC HUMAN	153	90	100	0,5263	0,6536	0,5831
STAPHYLOCOCCUS	266	78	172	0,6880	0,6466	0,6667
U937 HUMAN	102	16	26	0,6190	0,2549	0,3611
WAT MOUSE	494	20	116	0,8529	0,2348	0,3683
YEAST	151	34	67	0,6634	0,4437	0,5317
			<b>Média</b>	0,7551	0,2416	0,3611
			<b>Desvio Padrão</b>	0,1417	0,2155	0,1862

A Tabela 5 apresenta sinteticamente os valores de *recall* nos testes do modelo nas imagens providas pela base nas duas situações: clusterização na imagem completa;

e, clusterização na mesma imagem dividida em quadrantes. Também, apresenta o ganho, em percentual, da utilização do modelo na imagem dividida sobre a utilização na imagem completa. Ainda, a Figura 46 apresenta um gráfico em barras da Tabela 5.

Tabela 5: Comparação de *recall* nos testes do Fuzzy C-Means usando a imagem completa e dividida.

Imagem	Método		Ganho (%)
	Imagem inteira	Imagem dividida	
ARABIDOPSIS	0,7342	0,7342	0,00
BAT_MOUSE	<b>0,6824</b>	0,6706	-1,18
CEC_HUMAN	0,7679	<b>0,8036</b>	3,57
CSF_HUMAN	<b>0,5178</b>	0,4887	-2,91
DICTYSLUG	0,9200	0,9200	0,00
DLD1_HUMAN	0,8426	<b>0,8704</b>	2,78
ECOLI	0,7330	<b>0,8010</b>	6,80
ELC_HUMAN	0,8857	<b>0,9143</b>	2,86
HEPG2_HUMAN	0,7347	<b>0,7551</b>	2,04
HEPG2SP_HUMAN	<b>0,5935</b>	0,5355	-5,81
HL60_HUMAN	0,7692	<b>0,8077</b>	3,85
ISLETS_MOUSE	0,6667	0,6667	0,00
KIDNEY_HUMAN	0,7500	<b>0,7955</b>	4,55
LIVER_HUMAN	0,6812	<b>0,7826</b>	10,14
LIVER_MOUSE	0,7300	<b>0,7600</b>	3,00
LYMPHOCYTE_HUMAN	0,4935	<b>0,5584</b>	6,49
LYMPHOMA_HUMAN	0,6333	<b>0,7500</b>	11,67
MUSCLE_MOUSE	<b>0,6471</b>	0,6250	-2,21
NUCLEI_LIVER_HUMAN	0,9259	<b>0,9630</b>	3,70
NUCLEI_LIVER_MOUSE	0,9333	0,9333	0,00
NUCLEOLI_HELA_1D_HUMAN	0,8061	<b>0,8367</b>	3,06
NUCLEOLI_HELA_2D_HUMAN	0,5490	<b>0,5686</b>	1,96
PLASMA_HUMAN	0,4936	<b>0,5192</b>	2,56
PLATELET_HUMAN	0,9268	<b>0,9512</b>	2,44
RBC_HUMAN	0,3316	<b>0,5263</b>	19,47
STAPHYLOCOCCUS	<b>0,6920</b>	0,6880	-0,40
U937_HUMAN	0,5238	<b>0,6190</b>	9,52
WAT_MOUSE	0,8382	<b>0,8529</b>	1,47
YEAST	0,6040	<b>0,6634</b>	5,94

Na maioria das imagens, percebe-se um ganho no *recall* ao clusterizar a imagem por partes. O intuito desta estratégia de divisão em quadrantes é reduzir a influência global dos tons de *pixels* na clusterização dos *spots*, promovendo uma análise de similaridade mais local. Cada imagem tem suas particularidades, principalmente no que tange a concentração dos *spots* em dadas regiões da imagem, e esta é a razão pelas quais alguns testes não mostraram melhora, e também significa que a imagem não foi dividida apropriadamente. Apesar de apresentar alguma melhora, dividir em-

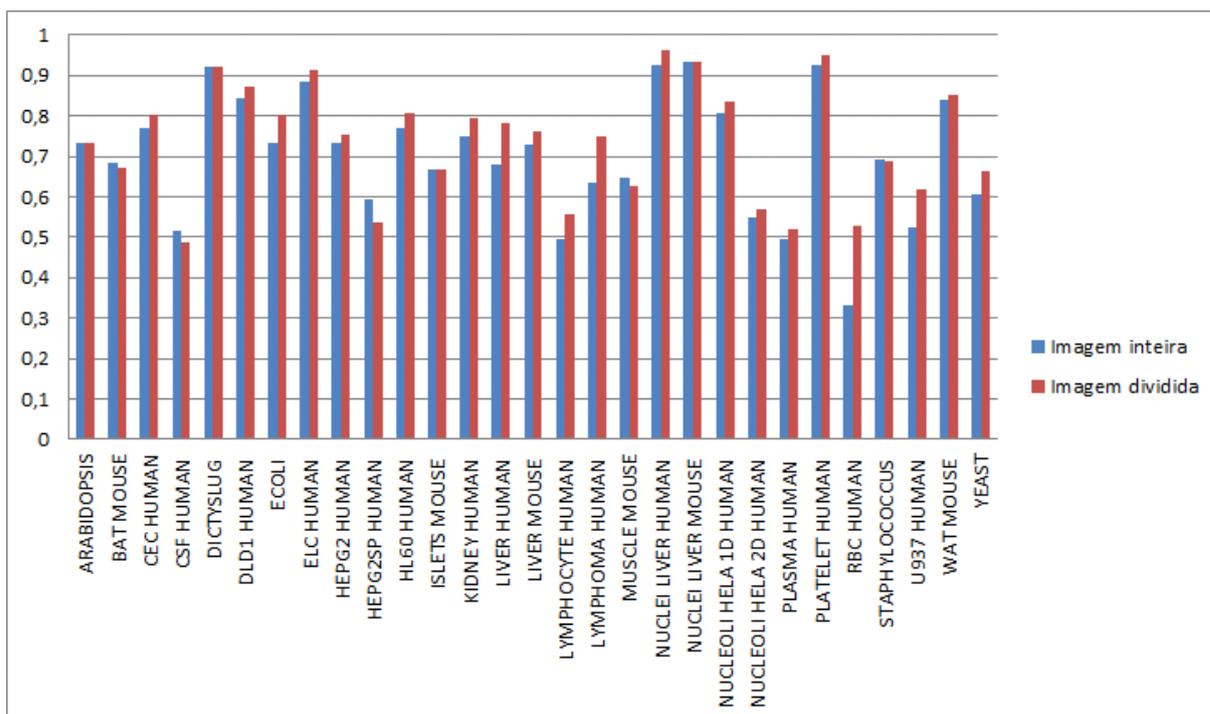


Figura 46: Uma visão gráfica da Tabela 5.

piricamente pode não ser a melhor alternativa. Neste caso, a imagem foi dividida em quatro quadrantes. Faz-se necessário, então, aprofundar os estudos nesta questão para determinar como exatamente cada imagem pode ser dividida para apresentar melhoria na identificação destas áreas específicas.

## 4.5 K-Means

No mesmo conjunto de imagens disponibilizadas pelo SWISS-2DPAGE, foi aplicado o algoritmo K-Means, descrito anteriormente, a fim de comparar esta técnica *crisp* ao Fuzzy C-Means. O processo de classificação e contagem dos *spots* foram os mesmos utilizados para o Fuzzy C-Means. Também foram utilizados os parâmetros padrão do K-Means.

Dada a execução deste, a melhor taxa de positivos verdadeiros foi obtida na imagem NUCLEI\_LIVER\_MOUSE. Nesta imagem, apenas 1 (um) *spot* confirmado não foi encontrado, apresentando assim um *recall* de 93,33%. Enquanto isso, a pior execução foi na imagem RBC\_HUMAN, onde o algoritmo K-Means encontrou 64 dos 190 (33,68%) *spots* confirmados. A Tabela 6 mostra o resultado da aplicação do K-Means nas imagens do SWISS-2DPAGE.

Tabela 6: Resultado da execução do K-Means nas imagens do SWISS-2DPAGE

Nome da imagem	Spots encontrados	Confirmados faltando	Confirmados encontrados	Recall	Precisão	F1
ARABIDOPSIS	147	21	58	0,7342	0,3946	0,5133
BAT MOUSE	204	27	58	0,6824	0,2843	0,4014
CEC HUMAN	354	13	43	0,7679	0,1215	0,2098
CSF HUMAN	208	154	155	0,5016	0,7452	0,5996
DICTYSLUG	514	2	23	0,9200	0,0447	0,0853
DL1 HUMAN	454	19	89	0,8241	0,1960	0,3167
ECOLI	274	55	151	0,7330	0,5511	0,6292
ELC HUMAN	331	4	31	0,8857	0,0937	0,1694
HEPG2 HUMAN	453	26	72	0,7347	0,1589	0,2613
HEPG2SP HUMAN	107	66	89	0,5742	0,8318	0,6794
HL60 HUMAN	332	6	20	0,7692	0,0602	0,1117
ISLETS MOUSE	256	22	41	0,6508	0,1602	0,2571
KIDNEY HUMAN	258	11	33	0,7500	0,1279	0,2185
LIVER HUMAN	209	46	92	0,6667	0,4402	0,5303
LIVER MOUSE	261	55	145	0,7250	0,5556	0,6291
LYMPHOCYTE HUMAN	176	39	38	0,4935	0,2159	0,3004
LYMPHOMA HUMAN	211	22	38	0,6333	0,1801	0,2804
MUSCLE MOUSE	194	48	88	0,6471	0,4536	0,5333
NUCLEI LIVER HUMAN	131	4	23	0,8519	0,1756	0,2911
NUCLEI LIVER MOUSE	347	1	14	0,9333	0,0403	0,0773
NUCLEOLI HELA 1D HUMAN	462	19	79	0,8061	0,1710	0,2821
NUCLEOLI HELA 2D HUMAN	54	23	28	0,5490	0,5185	0,5333
PLASMA HUMAN	456	317	309	0,4936	0,6776	0,5712
PLATELET HUMAN	338	4	37	0,9024	0,1095	0,1953
RBC HUMAN	146	126	64	0,3368	0,4384	0,3810
STAPHYLOCOCCUS	186	81	169	0,6760	0,9086	0,7752
U937 HUMAN	137	22	20	0,4762	0,1460	0,2235
WAT MOUSE	455	22	114	0,8382	0,2505	0,3858
YEAST	157	37	64	0,6337	0,4076	0,4961
			<b>Média</b>	0,6962	0,3262	0,3772
			<b>Desvio Padrão</b>	0,1493	0,2460	0,1919

## 4.6 Análise Comparativa

A Tabela 7 apresenta a comparação dos valores de *recall* entre as aplicações do K-Means, do Fuzzy C-Means e dos resultados do *Gaussian Naïve-Bayes* descritos em DIAS (2015), nas imagens do SWISS-2DPAGE. Para melhor entendimento, o melhor valor de *recall* em cada entrada está destacado em negrito. No final da Tabela são apresentadas a média e o desvio padrão do *recall* de cada técnica no conjunto de imagens. A Figura 47 apresenta um gráfico em barras da Tabela 7.

Tabela 7: Valores de *Recall* apresentados pelas técnicas.

Nome da Imagem	Recall			
	KMeans	FCM	FCM (imagem dividida)	<i>Gaussian Naïve-Bayes</i>
ARABIDOPSIS	<b>0,7342</b>	<b>0,7342</b>	<b>0,7342</b>	0,7215
BAT_MOUSE	<b>0,6824</b>	<b>0,6824</b>	0,6706	0,6118
CEC_HUMAN	0,7679	0,7679	<b>0,8036</b>	0,7679
CSF_HUMAN	0,5016	<b>0,5178</b>	0,4887	0,3657
DICTYSLUG	0,9200	0,9200	<b>0,9259</b>	0,88
DLD1_HUMAN	0,8241	0,8426	<b>0,8704</b>	0,75
ECOLI	0,7330	0,7330	<b>0,8010</b>	0,7087
ELC_HUMAN	0,8857	0,8857	<b>0,9143</b>	0,8
HEPG2SP_HUMAN	0,7347	0,7347	<b>0,7551</b>	0,7347
HEPG2_HUMAN	0,5742	<b>0,5935</b>	0,5355	0,4581
HL60_HUMAN	0,7692	0,7692	<b>0,8077</b>	0,7692
ISLETS_MOUSE	0,6508	<b>0,6667</b>	<b>0,6667</b>	0,5079
KIDNEY_HUMAN	0,7500	0,7500	<b>0,7955</b>	0,75
LIVER_HUMAN	0,6667	0,6812	<b>0,7826</b>	0,587
LIVER_MOUSE	0,7250	0,7300	<b>0,7600</b>	0,705
LYMPHOCYTE_HUMAN	0,4935	0,4935	<b>0,5584</b>	0,4286
LYMPHOMA_HUMAN	0,6333	0,6333	<b>0,7500</b>	0,5167
MUSCLE_MOUSE	<b>0,6471</b>	<b>0,6471</b>	0,6250	0,5882
NUCLEI_LIVER_HUMAN	0,8519	0,9259	<b>0,9630</b>	0,6296
NUCLEI_LIVER_MOUSE	<b>0,9333</b>	<b>0,9333</b>	<b>0,9333</b>	<b>0,9333</b>
NUCLEOLI_HELA_1D_HUMAN	0,8061	0,8061	<b>0,8367</b>	0,7959
NUCLEOLI_HELA_2D_HUMAN	0,5490	0,5490	0,5686	<b>0,6863</b>
PLASMA_HUMAN	0,4936	0,4936	<b>0,5192</b>	0,5096
PLATELET_HUMAN	0,9024	0,9318	<b>0,9512</b>	0,878
RBC_HUMAN	0,3368	0,3316	<b>0,5263</b>	0,3895
STAPHYLOCOCCUS	0,6760	0,6920	0,6880	<b>0,808</b>
U937_HUMAN	0,4762	0,5238	<b>0,6190</b>	0,3571
WAT_MOUSE	0,8382	0,8382	<b>0,8529</b>	0,8382
YEAST	0,6337	0,6040	<b>0,6634</b>	0,5743
<b>Média</b>	0,6962	0,7300	0,7551	0,6569
<b>Desvio Padrão</b>	0,1493	0,1519	0,1417	0,1636

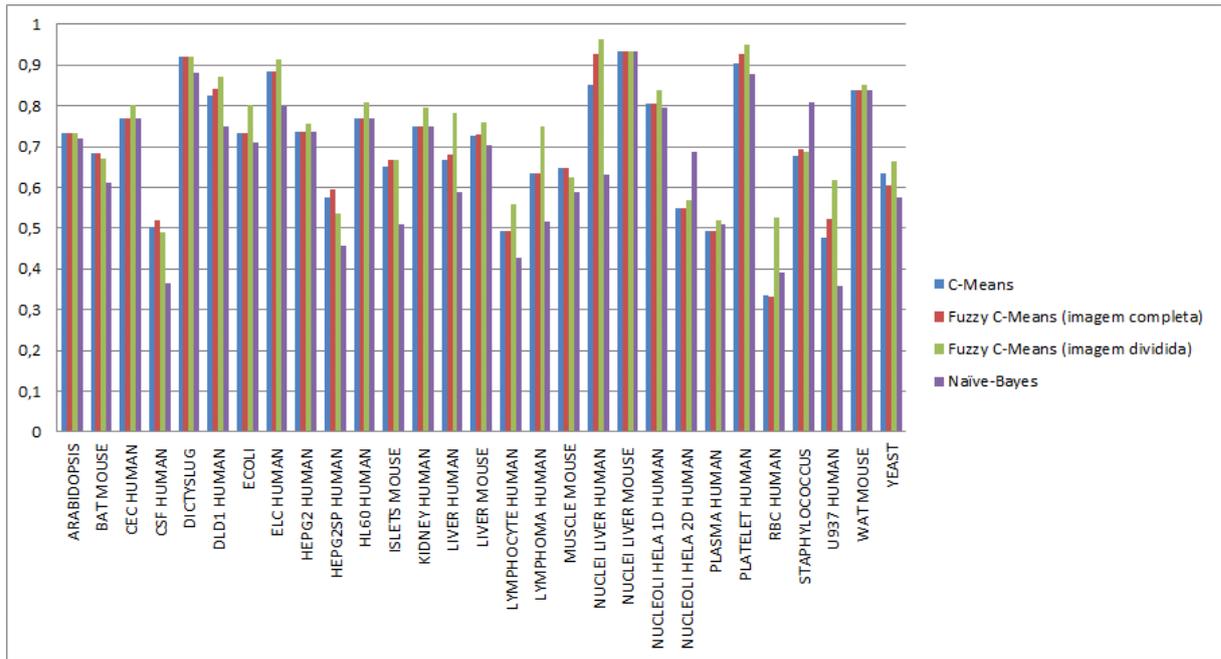


Figura 47: Uma visão gráfica da Tabela 7.

A Tabela 8 apresenta a comparação dos valores de  $F_1$  entre as aplicações do K-Means, do Fuzzy C-Means e dos resultados do *Gaussian Naïve-Bayes* descritos em DIAS (2015), nas imagens do SWISS-2DPAGE. Para melhor entendimento, o melhor valor de  $F_1$  em cada entrada está destacado em negrito. No final da Tabela são apresentadas a média e o desvio padrão do  $F_1$  de cada técnica no conjunto de imagens. A Figura 48 apresenta um gráfico em barras da Tabela 8.

Tabela 8: Valores de  $F_1$  apresentados pelas técnicas.

Nome da Imagem	$F_1$			
	KMeans	FCM	FCM (imagem dividida)	<i>Gaussian Naïve-Bayes</i>
ARABIDOPSIS	<b>0,5133</b>	0,5088	0,5088	0,5044
BAT_MOUSE	0,4014	<b>0,4028</b>	0,3918	0,3954
CEC_HUMAN	0,2098	0,2077	0,1911	<b>0,2177</b>
CSF_HUMAN	<b>0,5996</b>	0,6312	0,5709	0,5636
DICTYSLUG	0,0853	0,0849	0,0917	<b>0,0978</b>
DLD1_HUMAN	<b>0,3167</b>	0,3095	0,2942	0,3127
ECOLI	0,6292	0,6227	<b>0,6600</b>	0,6307
ELC_HUMAN	0,1694	0,1586	0,1749	<b>0,2171</b>
HEPG2SP_HUMAN	0,2613	0,2576	0,2624	<b>0,2748</b>
HEPG2_HUMAN	0,6794	<b>0,6866</b>	0,5764	0,5820
HL60_HUMAN	0,1117	0,1055	0,1176	<b>0,1384</b>
ISLETS_MOUSE	0,2571	0,2585	0,2493	<b>0,2700</b>
KIDNEY_HUMAN	0,2185	0,2136	0,2047	<b>0,2332</b>
LIVER_HUMAN	<b>0,5303</b>	0,5251	0,5243	0,5047
LIVER_MOUSE	0,6291	0,6071	0,6008	<b>0,6351</b>
LYMPHOCYTE_HUMAN	0,3004	0,2946	<b>0,3373</b>	0,3333
LYMPHOMA_HUMAN	0,2804	0,2774	0,2719	<b>0,3370</b>
MUSCLE_MOUSE	0,5333	0,5285	0,5346	<b>0,5369</b>
NUCLEI_LIVER_HUMAN	0,2911	0,2941	0,1512	<b>0,2956</b>
NUCLEI_LIVER_MOUSE	0,0773	0,0698	0,0698	<b>0,1081</b>
NUCLEOLI_HELA_1D_HUMAN	0,2821	0,2806	0,2680	<b>0,2994</b>
NUCLEOLI_HELA_2D_HUMAN	0,5333	0,5283	0,5321	<b>0,5469</b>
PLASMA_HUMAN	0,5712	0,5858	0,5667	<b>0,7867</b>
PLATELET_HUMAN	0,1953	<b>0,2152</b>	0,2074	0,1978
RBC_HUMAN	0,3810	0,3830	0,5831	<b>0,6066</b>
STAPHYLOCOCCUS	<b>0,7752</b>	0,7775	0,6667	0,7608
U937_HUMAN	0,2235	<b>0,3667</b>	0,3611	0,1923
WAT_MOUSE	<b>0,3858</b>	0,3845	0,3683	0,4021
YEAST	0,4961	0,4880	<b>0,5317</b>	0,5110
<b>Média</b>	0,3772	0,3667	0,3611	0,3963
<b>Desvio Padrão</b>	0,1919	0,1912	0,1862	0,1944

Foi aplicado o Teste T pareado nos resultados de *recall* para que fosse possível compara-los a partir de suas médias. Os valores de  $p$  para cada par testado podem ser vistos na Tabela 9. Com grau de confiança de 95%, pode-se afirmar que as médias apresentam valores significativamente diferentes. Assim, pode-se dizer que os melhores resultados, baseados no *recall*, foram apresentados pelo Fuzzy C-Means aplicando a clusterização por quadrantes, seguido por Fuzzy C-Means clusterizando a imagem inteira, K-Means e por fim o *Gaussian Naïve-Bayes*.

O Teste T pareado também foi aplicado nos resultados de  $F_1$ . Os valores de  $p$  para cada par testado podem ser vistos na Tabela 10. Com grau de confiança de 95%, pode-se afirmar que as médias não apresentam valores significativamente diferentes. Baseado nesta métrica não é possível afirmar que uma técnica é melhor que outra.

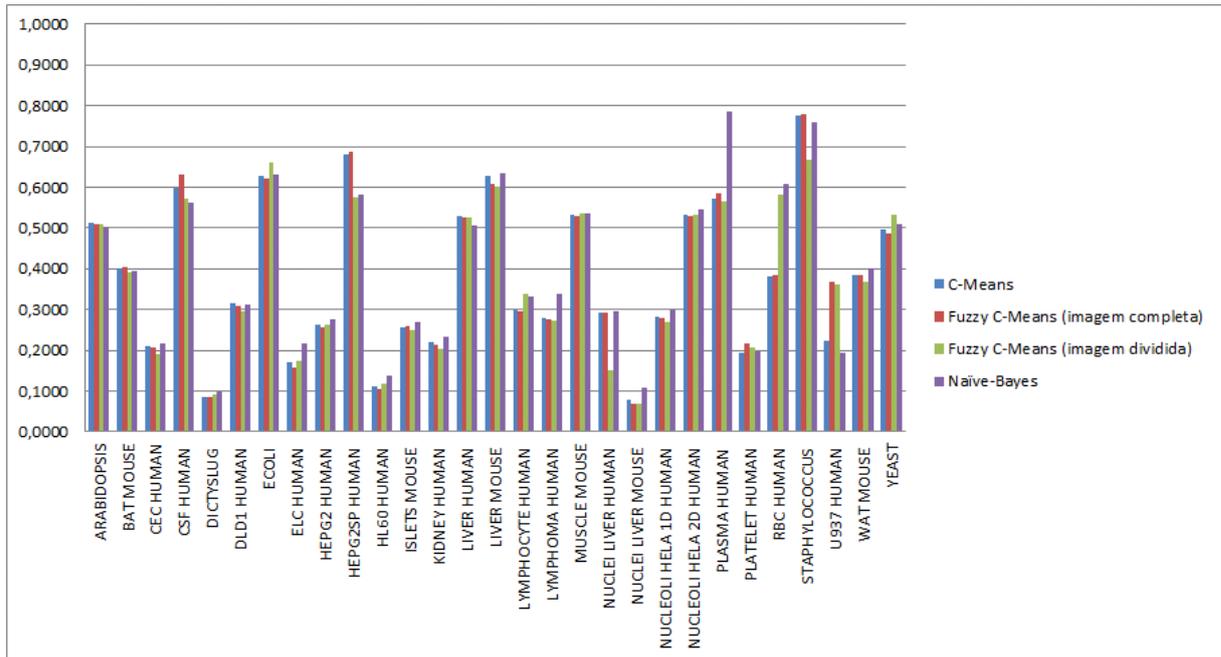


Figura 48: Uma visão gráfica da Tabela 8.

Tabela 9: Valores de  $p$  no Teste T pareado para o *Recall*.

Método	Fuzzy C-Means	FCM imagem dividida	KMeans	Gaussian Naïve-Bayes
Fuzzy C-Means	-	0,0013	0,0342	0,0071
FCM imagem dividida	0,0013	-	0,0002	0,0001
KMeans	0,0342	0,0002	-	0,0097
Gaussian Naïve-Bayes	0,0071	0,0001	0,0097	-

Tabela 10: Valores de  $p$  no Teste T pareado para o  $F_1$ .

Método	Fuzzy C-Means	FCM imagem dividida	KMeans	Gaussian Naïve-Bayes
Fuzzy C-Means	-	0,5586	0,4566	0,2682
FCM imagem dividida	0,5586	-	0,8375	0,0747
KMeans	0,4566	0,8375	-	0,1111
Gaussian Naïve-Bayes	0,2682	0,0747	0,1111	-

## 5 CONCLUSÕES

Imagens de eletroforese bidimensional são difíceis de analisar. Há o problema de possíveis ruídos na imagem, o que pode dificultar na análise. Além disso, os *spots* podem apresentar características bem diferentes. Por exemplo, dos *spots* confirmados na lista, mas não encontrados pelo modelo, apresentam tons de cinza muito distintos. A maioria dos confirmados possui um tom mais forte, entretanto, alguns dos confirmados possuem tons de cinza bem baixos. Isso é um grande desafio a ser superado.

Para a realização deste trabalho, foram estudados os conceitos de lógica *fuzzy* e suas aplicações, em especial, no processamento digital de imagens. Diversos métodos *fuzzy* foram propostos para o processamento de imagens, como o uso de funções de pertinência, agrupamento *fuzzy*, sistemas baseados em regras *fuzzy*, entropia *fuzzy*, medida *fuzzy* e integral *fuzzy*. Dentre estes, aprofundou-se os estudos em agrupamento *fuzzy* a fim de aplicar no reconhecimento de *spots* em imagens de eletroforese bidimensional.

Como contribuição deste trabalho destaca-se um modelo *fuzzy*, que utiliza o algoritmo de agrupamento Fuzzy C-Means a fim de detectar *spots* em imagens de géis de eletroforese bidimensional. Este modelo foi estendido para realizar a clusterização da imagens por partes. O modelo foi aplicado em imagens reais, disponibilizadas pelo Instituto Suíço de Bioinformática (SIB) na base de dados SWISS-2DPAGE. Os resultados obtidos foram comparados aos de outro modelo de agrupamento, não-*fuzzy*, chamado C-Means (ou K-Means) e aos de uma técnica de aprendizado não supervisionado, o *Gaussian Naïve-Bayes*, ambos nas mesmas imagens. Os resultados foram analisados com base em duas métricas: *recall* e  $F_1$ . Com base no *recall*, a clusterização *fuzzy* sendo realizada por partes mostrou melhores resultados. Com base na análise dos resultados pela métrica  $F_1$  pelo teste t pareado de Student não pode-se afirmar qual técnica é melhor.

Os resultados deste trabalho foram publicados em dois eventos, a saber:

- *14th Mexican International Conference on Artificial Intelligence (MICAI)* (DIAS; MAYDANA; AGUIAR, 2015).;
- *XXVIII Conference on Graphics, Patterns and Images (SIBGRAPI)* (MAYDANA;

DIAS; AGUIAR, 2015).

Há espaço para melhora no modelo *fuzzy*. Seja por desenvolver um método de dividir melhor a imagem a fim de clusterizar regiões bem específicas ou por utilizar técnicas de pré-processamento antes de fazer a clusterização utilizando o algoritmo FCM. É possível utilizar algumas técnicas como remoção do fundo, normalização, equalização e segmentação da imagem. As imagens utilizadas são relativamente grandes, contendo muita informação, e isto pode se tornar um obstáculo na clusterização destas imagens.

A primeira técnica, remoção o fundo, tem o intuito de reduzir os dados a serem processados, o que pode acelerar o processo. A segunda e terceira, normalização e equalização, visam mudar os dados apresentados, podendo influenciar diretamente na maneira com que a função *fuzzy* os clusteriza.

E, por fim, a última técnica, a segmentação da imagem, tem como objetivo dividir a imagem, resultando na clusterização de imagens menores. A imagem original apresenta regiões com intensidades diferentes, com isso pode-se achar aqueles *spots* com tons de cinza mais discrepantes.

## REFERÊNCIAS

ANDERSON, N. G.; ANDERSON, N. L. Twenty years of Two-dimensional electrophoresis: past, present and future. **Electrophoresis**, [S.l.], v.17, p.443–453, 1996.

ANGEL E.; SHREINER, D. **Interactive Computer Graphics: A Top-down Approach with WebGL**. Boston: Addison-Wesley, 2014.

BARROS, L. C. **Sobre sistemas dinâmicos fuzzy - teoria e aplicações**. 1997. Tese (Doutorado em Ciência da Computação) — Unicamp, Campinas.

BEZDEK, J. C. **Pattern Recognition with Fuzzy Objective Function Algorithms**. New York: Plenum Press, 1981.

BEZDEK, J. C.; PAL, S. K. **Fuzzy Models for Pattern Recognition**. [S.l.]: IEEE Press, 1992.

BOGUTH, G.; HARDER, A.; B., S.; WILDGRUBER, R.; WEISS, W. The current state of two-dimensional electrophoresis with immobilized pH gradients. **Electrophoresis**, [S.l.], v.21, p.1037–1053, 2000.

CHI, Z.; WU, J.; H., Y. Handwritten numeral recognition using self-organizing maps and fuzzy rules. **Pattern Recognition**, [S.l.], v.28, n.1, p.59–66, 1995.

CHI, Z.; WU, J.; H., Y. Handwritten numeral recognition using a small number of fuzzy rules with optimized defuzzification parameters. **Neural Networks**, [S.l.], v.8, n.5, p.821–827, 1995.

CHI, Z.; YAN, H. AVQ based fuzzy rules for image segmentation. In: FIFTH AUSTRALIAN CONFERENCE ON NEURAL NETWORKS (ACNN'94), 1994, Brisbane, Australia. **Anais...** [S.l.: s.n.], 1994. p.137–140.

CHI, Z.; YAN, H. Image segmentation using fuzzy rules derived from K-means clusters. **J. Electronic Imaging**, [S.l.], v.4, n.2, p.199–206, 1995.

CHI, Z.; YAN, H.; PHAM, T. **Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition.** [S.l.]: World Scientific, 1996. (Advances in fuzzy systems - applications and theory).

CIERO, L.; BELLATO, C. Proteoma: avanços recentes em técnicas de eletroforese bi-dimensional e espectrometria de massa. **Biotecnologia Ciência e Desenvolvimento**, [S.l.], v.29, p.158–164, 2002.

DIAS, M. d. S. **Detecção de spots em imagens oriundas de géis de eletroforese bi-dimensional utilizando aprendizado supervisionado.** Trabalho Acadêmico (Curso de Ciência da Computação) – Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2012.

DIAS, M. d. S.; MAYDANA, G. S.; AGUIAR, M. S. An Extension of Fuzzy C-Means applied to Spot Recognition in Images of Two-dimensional Electrophoresis. In: MEXICAN INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE (MICAI 2015), 14., 2015. **Proceedings...** IEEE, 2015. n.14, p.58.

DICKERSON, J. A.; KOSKO, B. Fuzzy function learning with covariance ellipsoids. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS (JOINT FUZZY-IEEE'93 AND ICNN'93 [IJCNN93]), 1993., 1993, San Francisco, California. **Proceedings...** IEEE/INNS, 1993. v.III, p.1162–1167.

DRIANKOV, D.; HELLENDORF, H.; REINFRANK, M. **An introduction to fuzzy control.** [S.l.]: Springer-Verlag, 1993.

ESQUEF, I. **Técnicas de Entropia em Processamento de Imagens.** Dissertação (Mestrado) – Centro Brasileiro de Pesquisas Físicas, Rio de Janeiro, RJ, Brasil, 2002.

GONZALEZ, R. C.; WOODS, R. E. **Digital image processing.** [S.l.]: Addison-Wesley, 1992.

HIRSCH, J.; HANSEN, K. C.; BURLINGAME, A. L.; MATTHAY, M. A. Proteomics: current techniques and potential applications to lung disease. **Lung Cellular and Molecular Physiology**, [S.l.], v.287, n.1, p.1–23, 2004.

HOOGLAND, C.; MOSTAGUIR, K.; SANCHEZ, J.-C.; HOCHSTRASSER, D. F.; APPEL, R. D. SWISS-2DPAGE, ten years later. **Proteomics**, [S.l.], v.4, n.8, p.2352–2356, 2004.

JAIN, A. K. **Fundamentals of Digital Image Processing.** Upper Saddle River, NJ: Prentice Hall, 1989.

KAZHIYUR-MANNAR, R.; SMIRAGLIA, D. J.; PLASS, C.; WENGER, R. Fuzzy-connective-based hierarchical aggregation networks for decision making. **Medical Image Analysis**, [S.l.], v.10, p.353–365, 2006.

KOHONEN, T. The self-organizing map. **Proceedings of the IEEE**, [S.l.], v.78, n.9, p.1464–1480, Sept. 1990.

KRISHAPURAM, R.; LEE, J. Fuzzy-connective-based hierarchical aggregation networks for decision making. **Fuzzy Sets and Systems**, [S.l.], v.46, p.11–27, 1992.

LEE, C. C. Fuzzy logic in control systems: fuzzy logic controller - Part I. **IEEE Transactions on Systems, Man, and Cybernetics**, [S.l.], v.20, n.2, p.404–418, 1990.

LEE, C. C. Fuzzy logic in control systems: fuzzy logic controller - Part II. **IEEE Transactions on Systems, Man, and Cybernetics**, [S.l.], v.20, n.2, p.419–435, 1990.

LI, F.; SEILLIER-MOISEWITSCH, F.; KOROSTYSHEVSKIY, V. R. Region-based statistical analysis of 2D PAGE images. **Computational Statistics & Data Analysis**, [S.l.], v.55, n.11, p.3059–3072, 2011.

MANNING, C. D.; RAGHAVAN, P.; SCHUTZE, H. **Introduction to Information Retrieval**. Cambridge, England: Cambridge University Press, 2008.

MARQUES FILHO, O.; VIEIRA NETO, H. **Processamento Digital de Imagens**. Rio de Janeiro: Brasport, 1999.

MASCARENHAS N.; VELASCO, F. **Processamento Digital de Imagens**. Buenos Aires, Argentina: Kapelusz, 1989.

MAYDANA, G. S.; DIAS, M.; AGUIAR, M. S. Aplicando Fuzzy C-Means para o Reconhecimento de Spots em Imagens Oriundas de Géis de Eletroforese Bidimensional. In: CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES (SIBGRAPI 2015), 28., 2015. **Proceedings...** [S.l.: s.n.], 2015. p.1–8.

NATALE, M.; CAIAZZO, A.; BUCCI, E. M.; FICARRA, E. A Novel Gaussian Extrapolation Approach for 2D Gel Electrophoresis Saturated Protein Spots. **Genomics, Proteomics & Bioinformatics**, [S.l.], v.10, n.6, p.336 – 344, 2012.

O'FARRELL, P. H. High resolution two-dimensional electrophoresis of proteins. **J. Biol. Chem.**, [S.l.], v.250, p.4007–4021, 1975.

OGAWA, H. Labeled Point Pattern Matching by Fuzzy Relaxation. **Pattern Recognition**, [S.l.], v.17, p.569–573, 1984.

OGAWA, H. A Fuzzy Relaxation Technique For Partial Shape-Matching. **Pattern Recognition Letters**, [S.l.], v.15, n.4, p.349–355, Apr. 1994.

PARK, S. C.; NA, I. S.; HAN, T. H.; KIM, S. H.; LEE, G. S. Lane detection and tracking in PCR gel electrophoresis images. **Computers and Electronics in Agriculture**, [S.l.], v.83, n.1, p.85–91, 2012.

ROCHA, T. L.; SILVA, M. C.; SA, M. F. Eletroforese bidimensional e análise de proteomas. **Comunicado Técnico da Embrapa**, [S.l.], v.136, 2005.

RODRIGUEZ, A.; FERNANDEZ-LOZANO, C.; DORADO, J.; RABUÑAL, J. R. Two-dimensional gel electrophoresis image registration using block-matching techniques and deformation models. **Analytical Biochemistry**, [S.l.], v.136, 2014.

SAVELONAS, M. A.; MYLONA, E. A.; MAROULIS, D. Unsupervised 2D Gel Electrophoresis Image Segmentation Based on Active Contours. **Pattern Recogn.**, [S.l.], v.45, n.2, p.720–731, 2012.

SCIKIT-FUZZY. **Fuzzy logic toolkit for SciPy**. (disponível via WWW em <https://pypi.python.org/pypi/scikit-fuzzy>).

SOLOMON C.; BRECKON, T. **Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab**. Hoboken: Wiley, 2011.

TANSCHKEIT, R.; GOMIDE, F.; TEIXEIRA, M. Modelagem e controle nebuloso. **Enciclopédia de Automática**, São Paulo, SP, Brasil, v.3, p.283–323, 2007.

THERRIEN, C. W. **Decision Estimation and Classification — an Introduction to Pattern Recognition and Related Topics**. New York: Wiley, 1989.

WILKINS, M. R.; PASQUALI, C.; APPEL, R. D.; OU, K.; GOLAZ, O.; SANCHEZ, J. C.; JAN, J. X.; GOOLEY, A. A.; HUGHES, E.; HUMOHERY-SMITH, I.; WILLIAMS, K. L.; HOCHSTRASSER, D. F. From proteins to proteomes: large scale protein identification by two-dimensional electrophoresis and amino acid analysis. **Nature Biotechnology**, [S.l.], v.14, p.61–65, 1996.

YAGER, R. R.; OVCHINNIKOV, S.; TONG, R. M.; NGUYEN, H. T. **Fuzzy Sets and Applications: Selected Papers**. New York: John Wiley & Sons, Inc., 1987.

ZADEH, L. A. Fuzzy Sets. **Information Control**, [S.l.], v.8, p.338–353, 1965.

ZADEH, L. A. The concept of a linguistic variable and its application to approximate reasoning, Part III. **Information Sciences**, [S.l.], v.8, p.301–357, 1975.

ZADEH, L. A. Fuzzy Logic. **Computer**, Los Alamitos, CA, USA, v.21, n.4, p.83–93, Apr. 1988.

ZIMMERMANN, H. J. **Fuzzy Set Theory and its Applications**. Boston, MA: Kluwer, 1985.

**Uma extensão do algoritmo Fuzzy C-Means aplicado  
ao reconhecimento de spots em imagens oriundas  
de Eletroforese Bidimensional – Geancarlo Saldanha  
Maydana**



**UNIVERSIDADE FEDERAL DE PELOTAS**  
Centro de Desenvolvimento Tecnológico  
Programa de Pós-Graduação em Computação



Dissertação

**Uma extensão do algoritmo Fuzzy C-Means aplicado ao  
reconhecimento de spots em imagens oriundas de  
Eletroforese Bidimensional**

**GEANCARLO SALDANHA MAYDANA**

Pelotas, 2016