



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E  
DA COMPUTAÇÃO



# **Uma proposição para o cálculo de mapas de disparidade de imagens estéreo usando um interpolador neural baseado em funções de base radial**

**Allan David Garcia de Araújo**

Orientador: Prof. Dr. Adrião Duarte Dória Neto

Co-orientador: Prof. Dr. Allan de Medeiros Martins

**Dissertação de Mestrado** apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da UFRN (área de concentração: Engenharia de Computação) como parte dos requisitos para obtenção do título de Mestre em Ciências.

Número de ordem PPgEE: M256  
Natal/RN, Janeiro de 2010

# **Uma proposição para o cálculo de mapas de disparidade de imagens estéreo usando um interpolador neural baseado em funções de base radial**

**Allan David Garcia de Araújo**

Dissertação de Mestrado aprovada em 13 de janeiro de 2010 pela banca examinadora composta pelos seguintes membros:

---

Prof. Dr. Adrião Duarte Dória Neto (orientador) ..... DCA/UFRN

---

Prof. Dr. Allan de Medeiros Martins (co-orientador) ..... DEE/UFRN

---

Prof. Dr. Antônio de Pádua de Miranda Henriques ..... UFERSA

---

Prof. Dr. Agostinho de Medeiros Brito Júnior ..... DCA/UFRN

*À minha esposa, Sayuri Yamazaki,  
pela paciência durante os anos em  
que tive que estar meio ausente para  
a concretização desse trabalho.*

---

# Agradecimentos

---

A Deus, em primeiro lugar, pelo seu misericordioso e incondicional amor, o qual me fortalece a cada dia, incentivando-me a continuar na luta pela concretização de meus objetivos.

Aos mestres, pelos preciosos conhecimentos transmitidos no decorrer do curso.

Aos meus orientadores Prof. Dr. Adrião Duarte Dória Neto e Prof. Dr. Allan de Medeiros Martins, pela dedicação, orientação e paciência ao longo do desenvolvimento deste trabalho.

Aos meus pais, pelas lições de amor, paciência e incentivo que sempre me deram ao longo de todos esses anos de vida.

Aos colegas do Laboratório de Sistemas Inteligentes pela força moral.

Aos demais colegas que dividiram disciplinas e sofreram juntos para fazer as listas de exercício e os trabalhos.

À CAPES, pelo apoio financeiro.

---

# Resumo

---

O presente trabalho visa buscar uma alternativa mais viável para o cálculo das disparidades em imagens de visão estéreo, utilizando um fator de salto que reduz a quantidade de pontos que são considerados da imagem capturada, e uma rede neural baseada em funções de base radial para interpolar os resultados obtidos. O objetivo a ser alcançado é produzir uma imagem de disparidades aproximada da real com algoritmos de baixo custo computacional, diferentemente dos algoritmos tradicionais.

**Palavras-chave:** Redes Neurais, Funções de Base Radial, Visão Computacional, Casamento Estéreo.

---

# Abstract

---

This study aims to seek a more viable alternative for the calculation of differences in images of stereo vision, using a factor that reduces the amount of points that are considered on the captured image, and a network neural-based radial basis functions to interpolate the results. The objective to be achieved is to produce an approximate picture of disparities using algorithms with low computational cost, unlike the classical algorithms.

**Keywords:** Neural Networks, Radial Basis Functions, Computer Vision, Stereo Matching.

---

# Sumário

---

<b>Sumário</b>	<b>i</b>
<b>Lista de Figuras</b>	<b>iii</b>
<b>Lista de Tabelas</b>	<b>vi</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	3
1.2 Objetivos . . . . .	4
<b>2 Visão Estéreo</b>	<b>5</b>
2.1 O que é Processamento Digital de Imagens . . . . .	5
2.2 Comentários iniciais sobre a Visão Estéreo . . . . .	7
2.3 Visão Humana . . . . .	8
2.4 Visão Binocular . . . . .	10
2.5 Geometria epipolar . . . . .	12
2.6 Reconstrução Estéreo . . . . .	13
2.7 Calibração . . . . .	15
2.8 Retificação . . . . .	15
2.9 Comentários finais . . . . .	16
<b>3 Redes Neurais Artificiais</b>	<b>18</b>
3.1 O neurônio biológico . . . . .	18
3.2 Propriedades básicas dos neurônios . . . . .	20
3.3 A história do estudo das Redes Neurais . . . . .	21
3.4 Neurônio artificial . . . . .	23
3.5 Perceptrons . . . . .	25
3.6 Redes Neurais de Função de Base Radial . . . . .	26
3.6.1 A estrutura básica de uma RBF . . . . .	27
3.6.2 Uso de RBF para classificação de padrões . . . . .	29

3.6.3	Uso de RBF para predição . . . . .	30
3.7	Comentários finais . . . . .	31
<b>4</b>	<b>Método Proposto</b>	<b>32</b>
4.1	Comentários iniciais sobre o método . . . . .	33
4.2	Hipóteses levantadas . . . . .	33
4.3	O procedimento adotado . . . . .	34
4.4	Comentários finais . . . . .	36
<b>5</b>	<b>Resultados</b>	<b>37</b>
5.1	Conjunto de entrada . . . . .	37
5.2	Disparidades sem interpolação . . . . .	39
5.3	Resultados do método . . . . .	40
5.4	Relação Erro-Tempo $\times$ Espaçamento da malha . . . . .	43
5.5	Comentários finais . . . . .	47
<b>6</b>	<b>Conclusões</b>	<b>48</b>
6.1	Perspectivas . . . . .	49
	<b>Referências bibliográficas</b>	<b>51</b>

---

# Lista de Figuras

---

1.1	Típico problema de visão estéreo. . . . .	2
2.1	Diagrama esquemático da visão humana. . . . .	9
2.2	Geometria Epipolar: O ponto $P$ , os centros óticos $O$ e $O'$ das duas câmeras, e as duas projeções $p$ e $p'$ de $P$ , todos estão no mesmo plano. . . . .	13
2.3	Triangulação na presença de erros de medição. . . . .	14
2.4	Triangulação em imagens retificadas. . . . .	16
3.1	Aspecto geral do neurônio biológico. . . . .	21
3.2	Modelo de McCulloch-Pitts para um neurônio artificial. . . . .	24
3.3	Neurônios artificiais simulando portas lógicas E, OU e NÃO. . . . .	25
3.4	Representação da classificação tomada pela RBF em relação a tomada pela MLP. . . . .	27
3.5	Estrutura básica de uma Rede de Função de Base Radial. . . . .	28
4.1	Primeira parte do diagrama do método proposto. . . . .	34
4.2	No algoritmo de correlação baseado em área, um vetor contendo todos os pixels vizinhos pertencentes a janela é considerado no momento da leitura do ponto. . . . .	35
4.3	Segunda parte do diagrama do método proposto. . . . .	35
5.1	Par "Aloe" de imagens estéreo originais que servirão de entrada para o algoritmo. . . . .	38
5.2	Par "Midd1" de imagens estéreo originais que servirão de entrada para o algoritmo. . . . .	38
5.3	Par "Bowling1" de imagens estéreo originais que servirão de entrada para o algoritmo. . . . .	38
5.4	Par "Lampshade1" de imagens estéreo originais que servirão de entrada para o algoritmo. . . . .	38
5.5	Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Aloe".	39

5.6	Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Midd1". . . . .	39
5.7	Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Bowling1". . . . .	39
5.8	Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Lampshade1". . . . .	39
5.9	Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Aloe". . . . .	40
5.10	Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Midd1". . . . .	40
5.11	Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Bowling1". . . . .	41
5.12	Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Lampshade1". . . . .	41
5.13	Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Aloe". . . . .	41
5.14	Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Midd1". . . . .	41
5.15	Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Bowling1". . . . .	41
5.16	Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Lampshade1". . . . .	41
5.17	Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Aloe". . . . .	42
5.18	Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Midd1". . . . .	42
5.19	Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Bowling1". . . . .	42
5.20	Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Lampshade1". . . . .	42
5.21	Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Aloe". . . . .	42
5.22	Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Midd1". . . . .	42
5.23	Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Bowling1". . . . .	43

5.24	Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Lampshade1". . . . .	43
5.25	Gráfico do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Aloe". . . . .	43
5.26	Gráfico do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Midd1". . . . .	43
5.27	Gráfico do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Bowling1". . . . .	44
5.28	Gráfico do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Lampshade1". . . . .	44
5.29	Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Aloe". . . . .	45
5.30	Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Midd1". . . . .	45
5.31	Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Bowling1". . . . .	46
5.32	Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Lampshade1". . . . .	46
5.33	Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Aloe". . . . .	46
5.34	Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Midd1". . . . .	46
5.35	Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Bowling1". . . . .	47
5.36	Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Lampshade1". . . . .	47

---

# Lista de Tabelas

---

5.1	Tabela dos valores do Erro-Tempo $\times$ Espaçamento da malha, gerado a partir do par estéreo "Aloe". . . . .	44
5.2	Tabela dos valores do Erro-Tempo $\times$ Espaçamento da malha, gerado a partir do par estéreo "Midd1". . . . .	44
5.3	Tabela dos valores do Erro-Tempo $\times$ Espaçamento da malha, gerado a partir do par estéreo "Bowling1". . . . .	45
5.4	Tabela dos valores do Erro-Tempo $\times$ Espaçamento da malha, gerado a partir do par estéreo "Lampshade1". . . . .	45

---

# Capítulo 1

## Introdução

---

A área de pesquisa de visão computacional é uma das mais recentes áreas derivada da também jovem área de processamento digital de imagens. Dentro da visão computacional estão inseridos problemas do mundo real que, embora sejam desempenhados de forma trivial por seres biológicos, encontram uma barreira sem precedentes quando tratados usando algum recurso computacional. Um dos desafios reside no fato de que seres biológicos conseguem ter uma noção de profundidade sobre o que enxergam, sendo capazes de auferir medidas sobre uma cena tal como "isto está perto" ou "isto está longe". Já em um sistema computacional, dotado de "visão" através de uma câmera, não se consegue a mesma percepção, daí porque no estudo da visão computacional existe uma sub-área chamada visão estereoscópica ou visão estéreo, que tem como desafio justamente conceber soluções computacionais que possam produzir tal noção de profundidade, utilizando para isso uma fonte de captura visual estéreo, ou seja, duas ou mais câmeras arranjadas de modo a possibilitar a triangulação dos pontos correlatos para o cálculo da coordenada de profundidade desejada. Para a referida triangulação são necessários algoritmos de correlação de imagens que visam encontrar esses pares de pontos correlatos. Todavia, os algoritmos disponíveis atualmente para calcular essa correlação possuem um elevado custo computacional para a maioria dos sistemas existentes.

A captura de uma imagem por uma câmera digital moderna registra a impressão luminosa que chega aos sensores internos dessa câmera. Essa captura consiste em uma projeção de uma cena tridimensional sobre um plano bidimensional, que é o sensor da câmera. Já o mapeamento de uma cena tridimensional sobre um plano da imagem é uma transformação do tipo "muitos para um", isto é, um ponto na imagem não determina unicamente a posição de um ponto correspondente do mundo [Gonzalez e Woods 1992].

Existe uma gama de aplicações onde a visão computacional é utilizada. Em aplicações reais, tais como navegação robótica, mapeamento geográfico, engenharia reversa, etc, são utilizadas câmeras como as do tipo citado anteriormente, onde a informação da

profundidade dos pontos na cena é perdida, fazendo-se necessário utilizar outra forma de recuperar essa informação. Uma solução possível é usar algum equipamento adicional (além da câmera digital) no momento da captura da imagem, a fim de recuperar a informação de profundidade. Exemplos disso são Sonar, Laser, e etc.. Entretanto, sabe-se que tal solução é muitas vezes onerosa, e não está disponível em máquinas mais simples. Outra forma que podemos lançar mão para recuperar a informação de profundidade é utilizar uma técnica conhecida por visão estereoscópica (ou estéreo).

A estereoscopia, ou visão estéreo, é o caso particular de processamento de visão computacional que utiliza por base imagens bidimensionais, adquiridas por um sistema de duas ou mais câmeras para perceber a dimensão de profundidade em uma cena [Marr 1982]. Neste caso, cada dispositivo vê a cena de dois referenciais diferentes, permitindo ao sistema computacional combinar as informações obtidas a partir do par de imagens, de modo a auferir uma representação tridimensional, como podemos observar na Figura 1.1. Este modelo tem forte inspiração biológica, e é a forma natural de funcionamento da nossa visão humana.

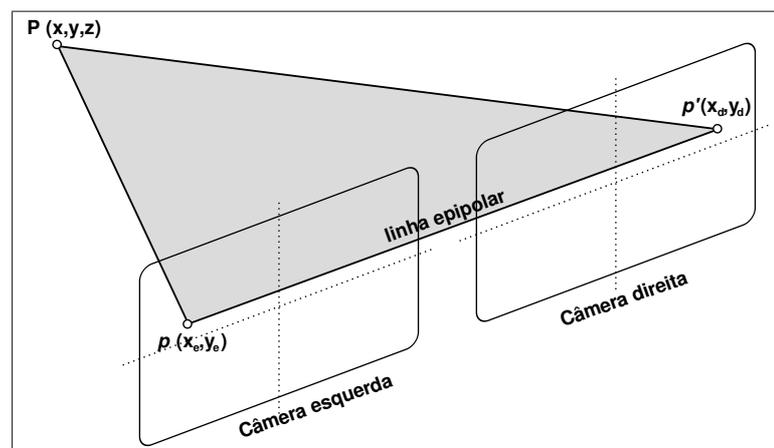


Figura 1.1: Típico problema de visão estéreo.

A combinação de duas imagens bidimensionais, que tem por objetivo localizar os pontos correspondentes entre uma imagem e outra, não é uma tarefa trivial. Exige-se ajustes de parâmetros que são dependentes da cena, e tempo computacional extremamente elevado. Duas técnicas principais comumente utilizadas para encontrar essa correlação são: as técnicas baseadas em áreas [Marapane e Trivedi 1989][Li e He 2002] e as técnicas baseadas em características [Goulermas e Liatsis 1999].

As técnicas baseadas em áreas utilizam a correlação entre os valores de intensidade de uma janela na imagem da esquerda, e outra na imagem da direita, produzindo um mapa

de disparidades denso. O tamanho da janela, bem como a área de busca na imagem, influenciam a exatidão da correspondência, e também, a complexidade do processamento [Sunyoto et al. 2004].

As técnicas baseadas em características baseiam-se no uso de características retiradas das imagens, tais como segmentos de borda, gradientes, e etc., realizando comparações simples entre atributos dessas características. Tais técnicas possuem maior velocidade e exatidão do que as técnicas baseadas em área, todavia, possuem a desvantagem da geração de um mapa de disparidade esparso, sendo necessário o uso de técnicas de interpolação para gerar mapas densos [Fernandes et al. 2004].

Além dessas duas técnicas, se mostram como técnicas poderosas para solucionar o problema da correspondência as baseadas em Redes Neurais Artificiais. Neste campo de atuação podemos citar o trabalho do Wang e Hsiao [Wang e Hsiao 1999], que faz uso de Rede-Neural de Múltiplas Camadas de Perceptrons (MLP) com treinamento supervisionado para encontrar os pontos correspondentes.

Portanto, o ideal de um algoritmo de visão estéreo consiste na capacidade de obter, a partir de um par de imagens em alta resolução, uma malha densa de pontos estruturados de tal forma que essa malha seja a representação exata da cena real, podendo essa informação ser obtida com o menor espaço de tempo possível, e utilizada em aplicações práticas. No entanto, sabe-se que a maioria das técnicas atuais não são adequados para sistemas de tempo real, já que os algoritmos de correlação desse par de imagens tem um peso computacional bem elevado.

## 1.1 Motivação

Este trabalho foi motivado principalmente pelo fato de que os algoritmos de correlação de imagens estéreo possuem um alto custo computacional. Além deste alto custo computacional, os algoritmos de correlação encontram em imagens reais problemas que são difíceis de serem contornados, tais como regiões de oclusão, distorções inerentes ao sistema de captura, variações luminosas, ruídos e outros. Portanto, procuramos neste trabalho contornar alguns desses problemas, mas, sobretudo, oferecer um método mais eficiente de cálculo das disparidades.

## 1.2 Objetivos

O objetivo do presente trabalho será o de encontrar os mapas de disparidades a partir de pares de imagens capturadas de um sistema de visão estéreo usando câmeras paralelas. Para alcançar este objetivo iremos, primeiramente, realizar os cálculos de disparidade entre as imagens capturadas em apenas alguns pontos, ou seja, tomaremos uma amostra das disparidades neste par de imagens. Depois, para encontrar os demais pontos, usaremos um interpolador neural de funções de base radial (RBF), que irá encontrar a superfície que interpola as disparidades previamente calculadas e assim poderemos reproduzir as demais disparidades de toda a imagem. Pretendemos, com isso, que o algoritmo resultante deste trabalho seja mais eficiente do que o algoritmo clássico de cálculo de disparidades baseado em área e que o mapa de disparidades resultante aproxime o mapa real.

---

# Capítulo 2

## Visão Estéreo

---

São duas as principais áreas de aplicação dos métodos de processamento digital de imagens: realce da informação pictorial para interpretação humana e processamento de dados de imagem para armazenamento, transmissão e representação para percepção de máquinas autônomas [Shapiro e Stockman 2001]. Neste segundo encontramos a área de processamento de imagens estereo, ou mais comumente chamada de visão estereo. Em particular, iremos elucidar as restrições geométricas e algébricas para o cenário mais comum de visão estereo binocular, ou seja, a partir de duas fontes de aquisição de imagens, no entanto, iniciaremos o capítulo com as definições formais a respeito do processamento digital de imagens.

### 2.1 O que é Processamento Digital de Imagens

Pode-se definir uma imagem como uma função de duas dimensões,  $f(x,y)$ , onde  $x$  e  $y$  são coordenadas planas, e a amplitude de  $f$  em qualquer par de coordenadas  $(x,y)$  é chamada de intensidade ou nível de cinza da imagem naquele ponto. Quando  $x$ ,  $y$ , e os valores de amplitude de  $f$  estão todos em quantidades discretas e finitas, nós dizemos que a imagem é digital [Petrou e Bosdogianni 1999]. O campo do processamento digital de imagem se refere ao processamento de imagens através de um computador digital.

A visão é o mais avançado de nossos sentidos, por isso não é surpreendente que as imagens desempenham o papel mais importante na percepção humana. No entanto, diferentemente dos seres humanos, que somente conseguem enxergar uma parte do espectro electromagnético, máquinas de imagens cobrem quase todo o espectro, que vai desde raios gama até ondas de rádio. Tais dispositivos podem operar em imagens geradas por fontes que os seres humanos não estão habituados a perceber, incluindo aí ultra-som, microscopia eletrônica, e imagens geradas por computador. Portanto, o processamento digital de imagem engloba um campo amplo e variado de aplicações.

Sobre quando o processamento de imagem acaba e outras áreas relacionadas iniciam, tais como análise de imagens e visão computacional, não existe um consenso entre os autores, visto que, às vezes, uma distinção é feita através da definição de processamento de imagem como uma disciplina em que a entrada e a saída de um processo são imagens. Segundo esta definição, até mesmo uma tarefa trivial de cálculo da intensidade média de uma imagem (que produz um único número) não seria considerada uma operação de processamento de imagem. Existem, entretanto, áreas como a visão computacional, cujo objetivo final é usar computadores para imitar a visão humana, incluindo o aprendizado, e ser capaz de fazer inferências e tomar ações com base em estímulos visuais. A área de visão computacional em si é um ramo da inteligência artificial (IA), sendo que a mesma tem como objetivo imitar a inteligência humana. Já a área de análise de imagem (também chamada de compreensão da imagem) é algo que está entre a definição de processamento de imagens e a de visão computacional.

Sendo assim, não há distinção clara entre processamento de imagens e visão computacional. No entanto, um paradigma útil é considerar três tipos de processos informatizados, são eles: processo de baixo, intermediário, e alto nível. O processo de baixo nível envolve operações primitivas, tais como, pré-processamento de imagem para reduzir o ruído, ganho de contraste e realce de imagem. De fato, o processo de baixo nível é caracterizado por serem, tanto as suas entradas como as suas saídas, compostas por imagens. O processo de nível intermediário de imagens envolve tarefas como a segmentação (particionamento de uma imagem em regiões ou objetos) ou a extração de descritores desses objetos para reduzi-los a uma forma adequada para o processamento e classificação (reconhecimento) computadorizados de objetos individuais. O processo de nível intermediário é caracterizado por suas entradas serem geralmente imagens, mas, as suas saídas, atributos extraídos dessas imagens, como por exemplo, bordas, contornos, objetos identificados, etc. Finalmente, o processo de alto nível envolve a compreensão de um conjunto de objetos reconhecidos, desempenhando funções cognitivas normalmente associadas com a visão.

Com base nas observações anteriores, vemos que um lugar lógico de sobreposição entre processamento de imagem e análise de imagem é a área de reconhecimento de regiões ou objetos individuais em uma imagem. Como exemplo podemos citar a área de análise automática de texto. O processo de aquisição de uma imagem da área que contém o texto e o pré-processamento dessa imagem extraíndo (segmentação) os caracteres individuais, descrevendo esses caracteres de uma forma adequada para o processamento do computador, e reconhecendo os caracteres individuais, podem também ser definidos como várias sub-áreas do processamento digital de imagem. Fazer o sentido do conteúdo deste texto pode ser visto como sendo do domínio da análise de imagem e até mesmo visão

computacional, dependendo do nível de complexidade implícita na frase "fazer sentido".

## 2.2 Comentários iniciais sobre a Visão Estéreo

Através de uma única imagem, apesar de toda informação que ela contém, não conseguimos acessar diretamente a medida de profundidade de um ponto posicionado dentro da cena. São necessárias pelo menos duas imagens para que possamos calcular, através de triangulação, uma medida de profundidade do que está sendo visto. Isto é, provavelmente, o motivo porque a maioria dos animais tem ao menos dois olhos e movimentam sua cabeça para direcionar sua visão para um alvo. Pelos mesmos motivos é que, quando construímos robôs autônomos que necessitam tomar decisões baseando-se no que veem, os criamos com um sistema de visão robótica composto por duas ou mais câmeras, assim como com algum sistema de análise de movimento. Portanto, antes de construirmos tal sistema de visão estéreo, precisamos entender como várias visões de uma mesma cena podem nos dar uma estrutura tridimensional do que está sendo visto, bem como as possíveis configurações de câmeras para tal reconstrução.

Sabemos que existem restrições geométricas e algébricas entre duas, três ou mais vistas da uma mesma cena. No cenário mais comum de visão estéreo binocular, vamos mostrar que qualquer ponto da primeira imagem deve estar posicionado no plano formado pela segunda imagem e os centros ópticos das duas câmeras. Essa restrição epipolar algebricamente pode ser representada por uma matriz  $3 \times 3$ , chamada de matriz essencial, quando os parâmetros intrínsecos das câmeras são conhecidos, e pela matriz fundamental, caso contrário. Três fotos da mesma linha vão introduzir uma restrição diferente, ou seja, que a interseção dos planos formados por suas imagens seja degenerado. Algebricamente, esta relação geométrica pode ser representada por um tensor de  $3 \times 3 \times 3$  trifocal. Mais imagens irão introduzir restrições adicionais, por exemplo, quatro projeções do mesmo ponto irão satisfazer certas relações tetra-lineares, cujos coeficientes são capturados pelo tensor tetra-focal, etc [Baumgart 1974]. Notavelmente, as equações que satisfazem as restrições geométricas para várias imagens do recurso da mesma cena podem ser criadas sem qualquer conhecimento das câmeras que observam a cena e, adicionalmente, existe um conjunto de métodos para estimar os parâmetros diretamente a partir dos dados da imagem.

A combinação das imagens gravadas pelos nossos dois olhos e que exploram a diferença (ou disparidade) entre eles permite-nos ter uma forte sensação de profundidade. Nesta seção estaremos preocupados com a concepção e a implementação de algoritmos que imitam nossa capacidade de executar esta tarefa, conhecida como estereoscopia. Note

que uma máquina pode ser equipada com três "olhos" ou mais, e isso nos levaria a investigar abordagens multi-câmeras. Todavia, este trabalho se deterá em comentar o caso particular de visão binocular, ou seja, baseada em duas câmeras.

Programas de computador confiáveis para a percepção estereoscópica possuem valor inestimável na navegação visual de robôs, cartografia, reconhecimento aéreo e fotogrametria de curto alcance. Eles também são de grande importância em tarefas como a segmentação de imagens para reconhecimento de objetos e na reconstrução de modelos tridimensionais da cena baseando-se em imagens estéreo [Binford 1984].

Visão estéreo envolve dois processos: o casamento binocular de características observadas pelos dois olhos, e a reconstrução de sua pré-imagem tridimensional. Esta última é relativamente simples: a pré-imagem de correspondência de pontos pode (em princípio) ser encontrada na intersecção dos raios passando por estes pontos com os respectivos centros associados. Assim, quando uma característica única na imagem é observada em um dado momento, a visão estéreo é fácil. No entanto, cada imagem é composta de centenas de milhares de pixels, com dezenas de milhares de características, tais como elementos de borda, é aí que outro método deve ser pensado para estabelecer as correspondências corretas e evitar medidas de profundidade erradas.

Embora o casamento da visão humana seja fácil e confiável na maioria das situações, podemos ser enganados: o resumo "single-image stereograms" que ficou popular no final dos anos noventa demonstra isso muito bem: neste caso, os padrões repetitivos ou criteriosamente montados de pontos aleatórios eram usados para enganar os olhos para concentrar a atenção nas correspondências erradas, produzindo uma impressão vívida de muitas camadas. Isto sugere que a construção de um programa de visão estéreo confiável é difícil, fato que será atestado repetidamente neste trabalho.

## 2.3 Visão Humana

Antes de comentarmos algo sobre os algoritmos existentes para realização de correspondência binocular, vamos fazer uma pausa por um momento para discutirmos os mecanismos subjacentes da estereoscopia humana. Em primeiro lugar, convém notar que, ao contrário das câmeras fixas ligadas a um equipamento estéreo, os olhos de uma pessoa podem girar em suas órbitas. A cada momento, eles se fixam em um determinado ponto no espaço, ou seja, eles giram para que duas imagens se formem nos centros das fóveas dos olhos.

Se  $l$  e  $r$  denotam os ângulos anti-horários entre os planos verticais de simetria dos dois olhos e dois raios que passam pelo mesmo ponto na cena, definimos a diferença corres-

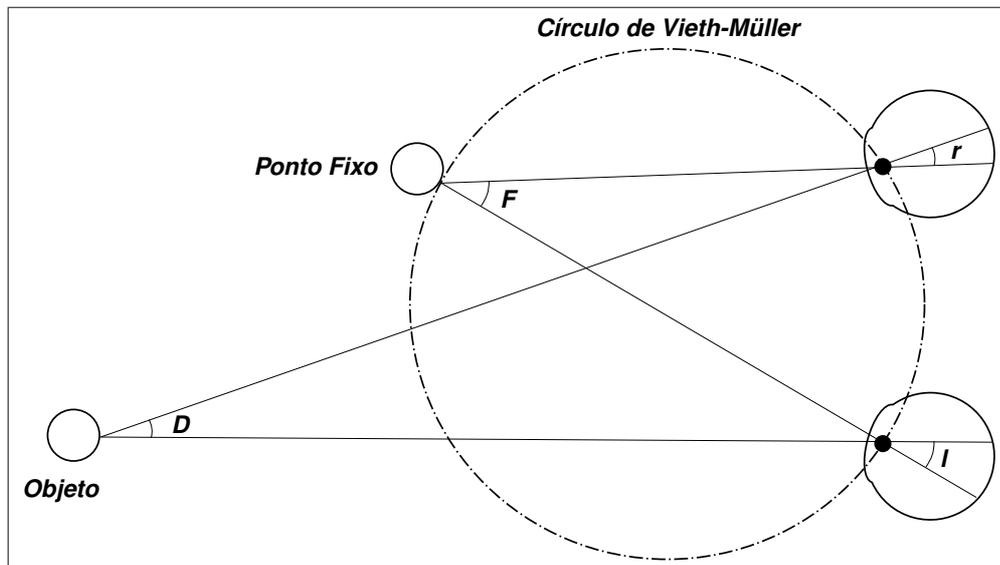


Figura 2.1: Diagrama esquemático da visão humana.

pondente,  $d = r - l$ , como vemos na Figura 2.1. É trivial provar trigonometricamente que  $d = D - F$ , onde  $D$  denota o ângulo entre esses raios, e  $F$  é o ângulo entre os dois raios que passam pelo ponto fixo. Os pontos com disparidade zero se encontram no chamado círculo de Vieth-Müller que passa através do ponto fixo e dos pontos focais no interior dos olhos. Os pontos situados no interior deste círculo tem disparidades positivas (também ditas convergentes), enquanto os pontos situados fora do círculo tem disparidade negativa (divergente), a localidade geométrica  $d$  irá variar na medida que o ponto fixo  $F$  também variar. Esta propriedade é suficiente para classificar as profundidades da região localizada próxima ao ponto fixo. No entanto, é também evidente que os ângulos de convergência entre o plano vertical longitudinal de simetria da cabeça e os dois raios de fixação devem ser conhecidos, a fim de reconstruir a posição absoluta de pontos de cena [Marr e Poggio 1979].

O caso tridimensional é naturalmente um pouco mais complicado, a localidade onde a disparidade é zero se torna uma superfície, um horoptero, mas a conclusão geral é a mesma, e exige o posicionamento absoluto os ângulos de convergência. Como já foi demonstrado por Wundt e de Helmholtz cerca cem anos atrás, há evidências de que esses ângulos não podem ser medidos com grande precisão pelo nosso sistema nervoso. Na verdade, o sistema visual humano pode ser enganado à acreditar que segmentos que realmente se encontram no mesmo plano vertical seja visto como em uma superfície convexa ou côncava, dependendo da distância entre o observador e este plano. Da mesma forma, modelos em miniatura de sólidos, se vistos através de uma perspectiva preparada são

quase indistinguíveis dos originais. Por outro lado, o ordenamento de pontos ao longo da linha de visão, pode ser julgado com muita precisão: por exemplo, é possível decidir qual dos dois alvos perto da horoptero está mais perto de um observador para as disparidades próximas do arco limiar (chamado estéreo-acuidade), que corresponde à distância mínima que pode ser medido com um olho (limiar de hiper-acuidade monocular). Por conseguinte, podemos afirmar que a saída de estereoscópica humana consiste principalmente de um mapa de informação relativa de profundidade, transmitindo uma ordem de profundidade parcial entre os pontos de cena. Nesse contexto, o principal papel dos movimentos oculares na estereoscopia seria trazer as imagens focalizadas para dentro da área de "Panum", um pequeno disco no centro da fóvea, onde as disparidades podem ser percebidas com muito maior precisão. Percebamos que objetos com altas disparidades ainda podem ser percebidos pela visão humana, todavia, eles irão aparecer como imagens duplicadas, um fenômeno conhecido como "diplopia".

## 2.4 Visão Binocular

Métodos de correlação localizam os pixels correspondentes em um par de imagens estéreo comparando os perfis de intensidade em uma dada vizinhança em busca de possíveis casamentos. Mais detalhadamente vamos considerar um par estéreo retificado e um ponto  $(u, v)$  na primeira imagem. Nós associamos com este ponto uma janela de tamanho  $p = (2m + 1) \times (2n + 1)$  centrada em  $(u, v)$  e compomos um vetor  $w(u, v) \in \mathbb{R}^p$  obtido pela varredura dos valores da janela, uma linha de cada vez. Agora, dada uma área de casamento em potencial na segunda imagem como sendo  $(u + d, v)$ , podemos construir um segundo vetor  $w(u + d, v)$  e definir a equação de correspondência normalizada entre estes vetores como sendo definida pela Equação 2.1, onde os índices  $u, v$  e  $d$  foram omitidos por motivos de concisão e  $\bar{w}$  representa o vetor cujas coordenadas são todas iguais à média das coordenadas de  $w$ .

$$C(d) = \frac{1}{|w - \bar{w}|} \frac{1}{|w' - \bar{w}'|} (w - \bar{w})(w' - \bar{w}') \quad (2.1)$$

A função de correlação normalizada  $C$  claramente varia de  $-1$  a  $1$ , e alcança seu valor máximo quando a intensidade da imagem está relacionada por uma transformação  $I' = \lambda I + \mu$  sendo  $\lambda$  e  $\mu$  constantes dependentes do ambiente, com  $\lambda > 0$ . Em outras palavras, o máximo desta função corresponde a padrões de imagem separados por uma constante de deslocamento e um fator de escala positiva, o correspondente estéreo pode ser encontrado buscando o máximo da função  $C$  em um intervalo pré-determinado de

disparidades.

Neste ponto, vamos fazer algumas observações sobre os métodos de correspondência com base na correlação. Primeiro, é facilmente demonstrado que maximizar a função de correlação é equivalente a minimizar a norma da diferença entre os vetores  $(1/|w - \bar{w}|)(w - \bar{w})$  e  $(1/|w' - \bar{w}'|)(w' - \bar{w}')$ , ou o equivalente a comparar a soma dos quadrados das diferenças entre os valores de pixel das janelas normalizadas. Em segundo lugar, embora o cálculo da função de correlação normalizada em cada pixel da imagem para um intervalo de algumas disparidades seja computacionalmente custoso, podemos aplicar de forma eficaz algoritmos recursivos para resolver este problema. Finalmente, um grande problema com relação a técnicas baseadas em estabelecer correspondência estéreo é que elas assumem implicitamente que a superfície observada é paralela aos planos da imagem. Isso sugere um algoritmo de dois passos, onde as estimativas iniciais da disparidade são usadas para ajustar os valores de disparidade a fim de compensar as disparidades de superfícies não paralelas.

Outro fator que deve ser considerado na aplicação das técnicas de correlação em imagens estéreo é comentado por Julesz [Julesz 1960] quando diz que embora a busca por pontos correspondentes pudesse ser conseguida aplicando algum critério de similaridade, como por exemplo por máxima correlação cruzada, isso não é garantia de funcionamento. Se a zona utilizada para procurar correspondências é pequena, o ruído pode facilmente destruir qualquer chance de correspondência, de outra forma, se o tamanho da zona é aumentada, inserimos ambiguidades nas fronteiras dos objetos que estão em distâncias diferentes. Assim, não se deve tentar realizar correspondência entre matrizes utilizando apenas a intensidade da imagem, precisamente porque um pixel corresponde apenas a uma intensidade que em padrões repetitivos pode facilmente ocorrer em outra região da imagem, confundindo a percepção de similaridade. Estes argumentos sugerem que as correspondências devem ser encontradas em uma variedade de atributos, alguns aspectos físicos da imagem devem ser considerados. Bordas são preferidas para compor os atributos de uma matriz conjuntamente com as intensidades do pixel bruto. Marr e Poggio propõem um algoritmo que segue estes dois princípios. Sua estrutura geral é bastante simples, conforme descrito abaixo:

1. Faça a convolução de duas imagens retificadas com filtros  $\nabla^2 G_\sigma$  aumentando os desvios padrão  $\sigma_1 < \sigma_2 < \sigma_3 < \sigma_4$ ;
2. Localize os cruzamentos de zero no Laplaciano ao longo da linha horizontal das imagens filtradas;
3. Para cada imagem filtrada  $\sigma$ , case os cruzamentos de zero com a mesma paridade e

- orientação no intervalo  $[-w_\sigma, +w_\sigma]$  do intervalo  $w_\sigma = 2\sqrt{2}\sigma$  das disparidades;
4. Use primeiro as disparidades encontradas nos filtros com maior desvio padrão e depois nos de menor desvio padrão para correspondências não encontradas.

Note que as correspondências são procuradas em cada filtro no intervalo  $[-w_\sigma, +w_\sigma]$ , onde  $w_\sigma = 2\sqrt{2}\sigma$  é a largura da porção central negativa do filtro  $\delta^2 G_\sigma$ . Esta escolha é motivada por considerações de natureza psicofísica e estatística, partindo do princípio de que as imagens são em sua essência ruídos brancos gaussianos. Claro que, limitando a busca de resultados no intervalo  $[-w_\sigma, +w_\sigma]$  impede que o algoritmo de correspondência encontre pares corretos cuja disparidade cai fora deste intervalo, todavia, este algoritmo foi implementado e extensivamente testado por Grimson [Grimson 1981] tanto em esteogramas aleatórios como em imagens naturais, apresentando excelentes resultados.

Uma característica natural no processo de correspondências é supor que a ordem dos pontos casados na primeira imagem será a mesma para seus correspondentes na segunda imagem, isso em uma mesma linha epipolar. Esta é a chamada restrição de ordenação que tem sido utilizado em técnicas de correspondência estéreo desde o início dos anos oitenta. Curiosamente, essa restrição não pode ser atendida por todas as cenas reais, em especial quando um pequeno objeto está a frente de outro maior, ocultando partes deste último ou quando, mais raramente, quando objetos transparentes estão envolvidos. Apesar destas exceções, a restrição continua a ser razoável, e pode ser usada para desenvolver algoritmos mais eficientes.

## 2.5 Geometria epipolar

Considere as projeções  $p$  e  $p'$  de um determinado ponto  $P$  observado por duas câmeras com centros óticos em  $O$  e  $O'$ , respectivamente. Estes cinco pontos pertencem ao plano epipolar definido pela interseção dos segmentos  $OP$  e  $O'P$  conforme podemos ver na Figura 2.2. Em particular, o ponto  $p'$  está na linha  $l'$  onde este plano e a retina  $\Pi'$  da segunda câmera se interceptam. A linha  $l'$  é a linha epipolar associada com o ponto  $p$ , e esta passa através do ponto  $e'$  onde a linha de base une os centros óticos  $O$  e  $O'$  interseção  $\Pi'$ . Da mesma forma, passa através da interseção  $e$  da linha de base com o plano  $\Pi$ .

Os pontos  $e$  e  $e'$  são chamados de epípólos das duas câmeras. O epípólo  $e'$  é a imagem (virtual) do centro ótico  $O$  da primeira câmera visto pela segunda câmera, e vice-versa. Como observado antes, se  $p$  e  $p'$  são imagens do mesmo ponto, então  $p'$  deve estar na linha de epipolar associadas a  $p$ . Essa restrição epipolar desempenha um papel fundamental na visão estéreo e na análise de movimento.

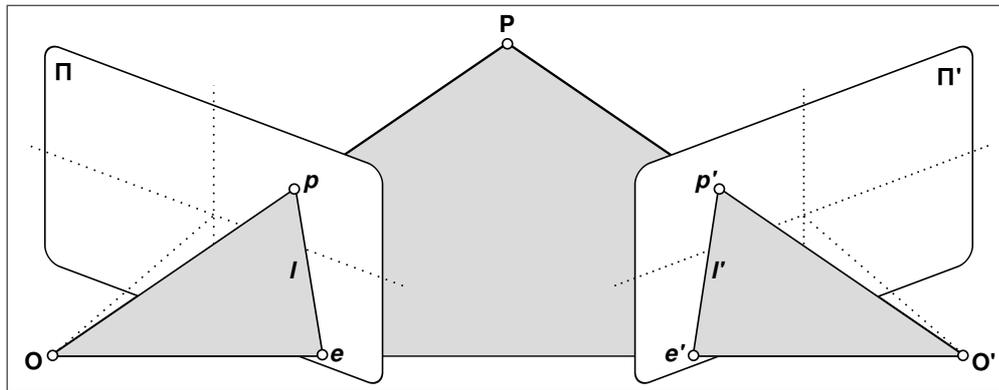


Figura 2.2: Geometria Epipolar: O ponto  $P$ , os centros óticos  $O$  e  $O'$  das duas câmeras, e as duas projeções  $p$  e  $p'$  de  $P$ , todos estão no mesmo plano.

Vamos supor, por exemplo, que conhecemos os parâmetros das duas câmeras de um equipamento de visão estéreo. Veremos que a parte mais difícil da análise de dados em visão estéreo é estabelecer as correspondências entre as duas imagens, ou seja, decidir quais os pontos na imagem da direita que correspondem aos pontos na imagem da esquerda. A restrição epipolar limita muito a busca por essas correspondências: na verdade, uma vez que assumimos que o equipamento esteja calibrado, as coordenadas do ponto  $p$  determinam completamente o raio que une  $O$  a  $p$ , e assim, o plano associado  $OO'p$  e a linha epipolar. A busca por correspondência pode ser restrita a essa linha, em vez de buscar em toda a imagem. Na análise de movimento de dois quadros, por outro lado, cada câmera pode ser calibrada internamente, mas, a transformação que rege as correspondências de coordenadas entre um quadro e outro é desconhecida. Neste caso, a geometria epipolar obviamente, restringe o conjunto de movimentos possíveis.

## 2.6 Reconstrução Estéreo

Dado um equipamento de visão estéreo calibrado e dois pontos de imagem  $p$  e  $p'$ , é do princípio direto para reconstruir a cena correspondente calcular a interseção dos dois raios  $R = Op$  e  $R' = O'p'$ . No entanto, os raios  $R$  e  $R'$ , na prática, nunca irão se cruzar devido aos erros naturais de calibração e localização, como notamos na Figura 2.3. Neste contexto, várias abordagens razoáveis para o problema da reconstrução podem ser adotadas. Por exemplo, podemos optar por construir o segmento de reta perpendicular a  $R$  e  $R'$ , que interceptam ambos os raios: o ponto médio deste segmento  $P$  é o ponto mais próximo de ambos os raios e pode ser tomado como a pré-imagem de  $p$  e  $p'$ .

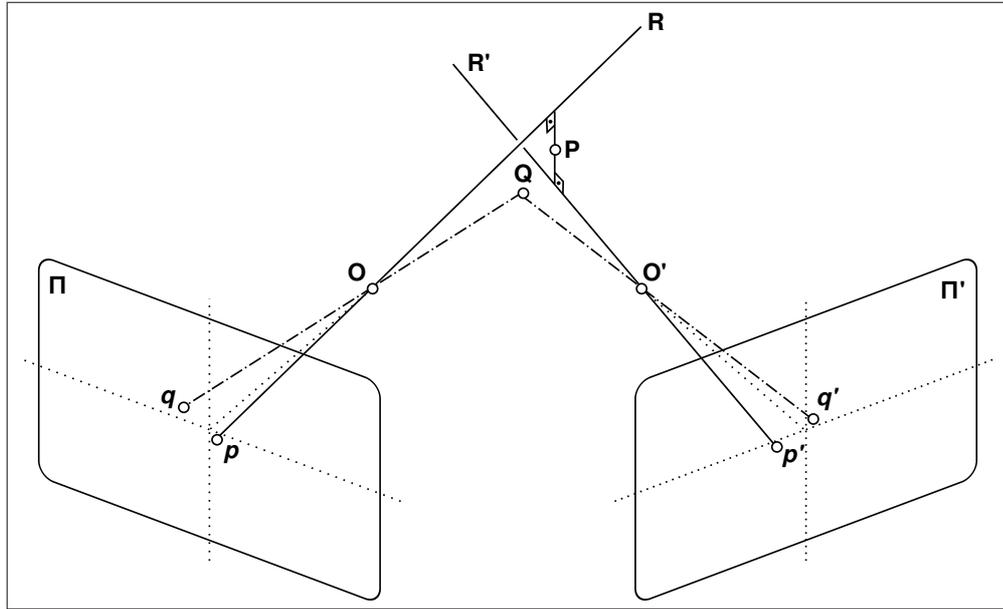


Figura 2.3: Triangulação na presença de erros de medição.

Alternativamente, nós podemos reconstruir um ponto da cena usando uma abordagem puramente algébrica: dadas as matrizes de projeção  $M$  e  $M'$  e os pontos coincidentes  $p$  e  $p'$ , podemos reescrever as restrições  $zp = MP$  e  $z'p' = M'P$  como definido na Equação 2.2.

$$\begin{cases} p \times MP = 0 \\ p' \times M'P = 0 \end{cases} \Rightarrow \begin{pmatrix} (p_x)M \\ (p'_x)M' \end{pmatrix} P = 0 \quad (2.2)$$

Este é um sistema restrito com quatro equações lineares independentes nas coordenadas homogêneas de  $P$ , que é facilmente resolvido usando as técnicas pelo método dos mínimos quadrados. Diferentemente da abordagem anterior, este método de reconstrução não tem uma interpretação geométrica óbvia, mas, generaliza facilmente para o caso de três ou mais câmeras, cada nova imagem simplesmente adicionando duas restrições adicionais.

Finalmente, nós podemos reconstruir a cena do ponto associado a  $p$  e  $p'$  como o ponto  $Q$  com imagens  $q$  e  $q'$  que minimizam  $d^2(p, q) + d^2(p', q')$ . Ao contrário dos outros dois métodos citados anteriormente, esta abordagem não permite o cálculo de forma fechada do ponto de reconstruído, que deve ser estimado através de técnicas dos mínimos quadrados não-lineares. A reconstrução obtida por qualquer um dos outros dois métodos pode ser usada como um palpite inicial para o processo de otimização. Esta abordagem não-linear também se generaliza rapidamente para o caso de várias imagens.

Antes de passar a estudar o problema da visão binocular, vamos comentar brevemente sobre dois componentes-chave dos sistemas de visão estéreo: calibração da câmera e retificação da imagem.

## 2.7 Calibração

Vamos assumir ao longo deste trabalho que todas as câmeras foram cuidadosamente calibradas para que seus parâmetros fossem precisamente conhecidos em relação a algum sistema de coordenadas fixo mundo. Este é, naturalmente, um pré-requisito para que algum métodos de reconstrução seja utilizado, uma vez que exigem que a projeção das matrizes associadas a duas câmeras sejam conhecidas, ou, equivalentemente, que um raio definido esteja associado a cada ponto da imagem. É importante também salientar que, uma vez que os parâmetros da câmera são conhecidos, é uma questão simples estimar a geometria de multi-visões (matriz essencial para duas exibições, tensor trifocal para três, etc).

## 2.8 Retificação

Os cálculos associados com algoritmos estéreo se tornam, muitas vezes, extremamente simples quando as imagens de interesse são retificadas, ou seja, passa-se a ter duas fotos projetivamente equivalentes com um plano de imagem comum, paralelo à linha de base que une os dois centros óticos. O processo de retificação pode ser implementado projetando as imagens originais no plano da nova imagem. Com uma escolha apropriada do sistema de coordenadas, as linhas epipolares são corrigidas nas novas imagens, e elas também se tornam paralelas à linha de base.

Conforme se observa, existem dois parâmetros envolvidos na escolha do plano de imagem retificada, a distância entre este plano e a linha de base, que é essencialmente irrelevante, pois, modificá-la só mudará a escala da retificação das imagens, um efeito facilmente compensado por uma escala inversa no sistema de eixos coordenados da imagem, e a direção do plano retificado normal no plano perpendicular à linha de base. A opção mais natural é justamente escolher um plano paralelo à linha onde as duas retinas originais se cruzam, minimizando a distorção associada ao processo projeção.

No caso das imagens retificadas, a noção de disparidade comentada anteriormente de modo informal tem um significado preciso agora: dado dois pontos  $p$  e  $p'$  localizados na mesma linha de varredura das imagens esquerda e direita, com coordenadas  $(u, v)$  e  $(u', v)$ ,

a disparidade é definida como a diferença  $d = u' - u$ . Vamos assumir a partir de agora coordenadas da imagem normalizadas. Se  $B$  denota a distância entre os centros ópticos, também chamados de referência, neste contexto, é fácil mostrar que a profundidade de  $P$  no sistema de coordenadas (normalizado) ligado à primeira câmera é  $z = -B/D$ .

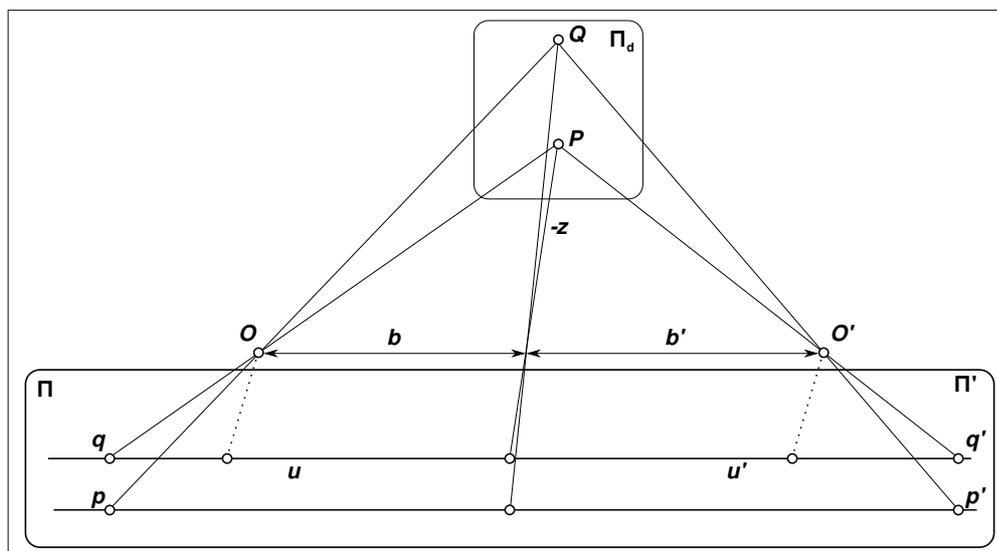


Figura 2.4: Triangulação em imagens retificadas.

Para mostrar isso, consideremos a Figura 2.4. Vamos atentar primeiro aos pontos  $q$  e  $q'$  com as coordenadas  $(u, 0)$  e  $(u', 0)$ , e ponto correspondente na cena sendo  $Q$ . Deixemos que  $b$  e  $b'$  representem as distâncias respectivas entre a projeção ortogonal de  $Q$  na linha de base e os dois centros óticos  $O$  e  $O'$ . Os triângulos  $qQq'$  e  $OOO'$  são similares, e assumimos imediatamente que  $b = zu$  e  $b' = -zu'$ . Assim,  $B = -zd$ , o que comprova o resultado de  $q$  e  $q'$ . O caso geral envolvendo  $p$  e  $p'$  com  $v \neq 0$  segue-se imediatamente do fato de que a linha  $PQ$  é paralela as duas linhas  $pq$  e  $p'q'$  e, portanto, paralela ao plano imagem corrigida. Em particular, o vetor de coordenadas do ponto  $P$  no quadro anexo à primeira câmera é  $P = -(B/d)p$ , onde  $p = (u, v, 1)^T$  é o vetor de coordenadas normalizadas da imagem  $p$ . Esta equação ainda possibilita um outro método para retificação de pares de imagens estéreo.

## 2.9 Comentários finais

Neste capítulo comentamos sobre as características básicas de um sistema de visão estéreo, assim como sobre os seus paralelos biológicos. Apresentamos, ainda, a geometria básica envolvida neste tipo de sistema, como também os elementos matemáticos

fundamentais, as técnicas de pré-tratamento, as correlação e as restrições inerentes à visão binocular.

A seguir, veremos um conteúdo complementar ao trabalho, discorrendo sobre as redes neurais e, mais especificamente, sobre redes de função de base radial, como elas podem agregar valor no provimento de soluções para tratar as dificuldades dos sistemas de visão estéreo.

---

## Capítulo 3

# Redes Neurais Artificiais

---

Neste capítulo discorreremos, resumidamente, sobre um conjunto de conceitos estabelecidos no estudo formal dos neurônios e das redes neurais, são eles: o neurônio biológico, as propriedades básicas dos neurônios, o neurônio artificial, a história do estudo das redes neurais, o aprendizado de máquina, e, por último, as redes neurais de função de base radial (RBF). Faremos, ainda, algumas comparações entre o modelo computacional artificial das redes neurais e os seus correlatos biológicos.

### 3.1 O neurônio biológico

Embora estejamos falando sobre o neurônio biológico no singular, trazendo a ideia de um único tipo de neurônio, a verdade é que existem no mundo diversas variações de neurônios, cada um com características especializadas para determinadas tarefas. Vejamos o ser humano, em um único indivíduo podemos perceber a presença de células nervosas responsáveis pela visão, outras pelo tato, outras pelo cheiro, outras para levar informação de um canto para outro, etc. e estas também diferem de neurônios encontrados em outras espécies, todas com suas particularidades e variações próprias. Na teoria, no entanto, essas variedades são ignoradas para simplificar o estudo e as suas complexidades abstraídas para facilitar a compreensão dos diferentes aspectos do desenvolvimento da rede neural, aprendizagem e função. Desta forma, é importante que saibamos que não existe uma coisa como um "neurônio típico" ou uma forma única do neurônio [Arbib 1981]. O que apresentaremos, então, será apenas um ponto de partida para o entendimento do neurônio e seus conceitos-chave, que poderão ser aprofundados em estudo mais detalhado.

Em computação neural, a tecnologia que tenta imitar o funcionamento das cadeias de neurônios encontradas no mundo natural define o neurônio como tendo um aspecto que também pode sofrer mudanças e variações. Cada arquitetura de rede neural apresenta suas peculiaridades inerentes à função final daquela rede, sendo tudo isso perfeitamente aceito

pela correlação com os neurônios reais, que também sofrem variações. Uma analogia para o entendimento do neurônio poderia ser em relação a um veículo de transporte. Ora, um veículo pode ser um carro, um trem, um avião, um barco, etc. Ele pode transportar pessoas ou cargas. As pessoas podem ser tripulação, passageiros, ou piloto. No entanto, se fossemos estudar o veículo de transporte, na verdade falaríamos sobre alguns exemplos-chave de forma (ex. as diferenças entre o carro e o avião.), ou de função (ex. aviões de carga ou de pessoas.). Além disso, gostaríamos de encontrar exemplos interessantes de evolução: como sistemas para a auto-estrada moderna, que não teria sido criada sem a pressão do aumento do tráfego automóvel. A maioria das características dos automóveis é adaptada para a existência de estradas fechadas, e alguns recursos (por exemplo, piloto automático) são especificamente adaptados às condições de boa estrada.

Iniciaremos nosso comentário observando que um cérebro é composto por uma rede de células chamadas neurônios, todas unidas a receptores e atuadores. A entrada para a redes de neurônios é fornecida por receptores, que monitoram continuamente as mudanças no ambiente externo e interno. Células chamadas neurônio motores, regidas pela rede cerebral, controlam o movimento dos músculos e da secreção de glândulas. No meio deste processo está uma rede intrincada de neurônios, que em alguns animais simples não passa de uma centena, e no cérebro humano chega a centenas de bilhões, combinando continuamente sinais dos receptores em um sistema próprio de codificação e transferindo esses sinais para neurônios motores que irão atuar adaptativamente no ambiente. Em animais com espinha dorsal, especialmente nos humanos, essa rede motora principal, também chamada de "sistema nervoso central", é o principal sistema de neurônios motores que atuam sobre o ambiente com estímulos que já foram processados na rede neural do cérebro. Já os invertebrados possuem variações extremamente peculiares do sistema nervoso central. Ambos vertebrados e invertebrados são fonte infinita de inspiração para tecnólogos projetarem redes neurais para os mais variados fins, como processamento sensorial, visão computacional, controle de robôs, e várias outras aplicações.

O cérebro fornece muito mais do que um simples sistema de estímulo-resposta, embora exista o que chamamos de reflexo, que mais parece com um atalho para certos tipos de estímulos que necessitam de ação imediata "programada de fábrica". O que acontece é que os sinais nervosos que são enviados ao cérebro, vindos do ambiente, se unem a outros já existentes em uma emaranhada cadeia de interligações. Estes sinais não apenas produzem outros sinais que são reenviados ao sistema nervoso central, mas, também reprogramam as propriedades armazenadas na própria rede, como uma espécie de histórico, de tal modo que os estímulos posteriores já não mais produzirão os mesmos efeitos, mas, sim, serão analisados levando em consideração as experiências anteriores.

As células sensoriais, que servem de estímulo para o que é processado na rede neural, também existem em diversas variações. Os cones e os bastonetes nos olhos respondem à luz, já as células ciliadas no ouvidos à pressão, e as células gustativas na boca respondem à vestígios de elementos químicos. Isso sem falar nos receptores de toque, existentes na pele que são sensíveis ao movimento, à temperatura, ou a sensações de dor. Podemos, portanto, classificá-los em duas categorias: os sentidos de proximidade, como o tato e o paladar, estes são estimulados quando em contato com a coisa sentida, e os sentidos de distância, tais como a visão e a audição, que são estimulados mesmo à distância da coisa sentida. O olfato é algo entre as duas classes, pois, podemos sentir odores que estão em contato, mas, também, o sentimos de longe. Além disso, mesmo os sentidos de proximidade também são, em alguns casos, sentidos de distância quando podemos sentir o vento ou o calor de uma fogueira. De uma forma mais geral, grande parte do que percebemos do mundo que nos rodeia, chega no nosso cérebro de uma forma até inconsciente como um conjunto de estímulos provenientes de vários sistemas sensoriais.

O correto funcionamento das células atuadoras também dependem dos estímulos provenientes das células receptoras. Um movimento, por exemplo, se dá pela comparação de onde se está para onde se quer chegar. Desta forma, além dos monitores externos, temos, ainda, receptores que monitoram as atividades dos músculos, tendões e articulações para fornecer continuamente informação sobre as tensões, o comprimento dos músculos, os ângulos das articulações e, também, as velocidades de movimento dos mesmos. O sistema de equilíbrio na cabeça monitora gravidade e acelerações. Neste sistema, as células ciliadas percebem o movimento dos fluidos existentes no seu interior para dar respostas adequadas. Existe, ainda, receptores para monitorar o nível de química do sangue, do estado do coração e dos intestinos, nível de glicose no fígado e quantidade de líquidos nos rins. Talvez o principal componente do cérebro, o hipotálamo, também reage baseado no que as células receptoras lhe informam. Se estimulado, um animal pode beber e comer em quantidades demasiadas, mesmo já estando abastecido. O cérebro recebeu a mensagem de que água ou comida estão faltando, e assim, instrui o animal em conformidade com essa mensagem, independente dos sinais contraditórios enviados, por exemplo, de um estômago dilatado.

## 3.2 Propriedades básicas dos neurônios

Para compreender melhor a dinâmica do processamento neural e como se dá essa interligação entre receptores e atuadores, devemos observar o funcionamento geral do neurônio. Como já comentamos, não existe tal coisa como um neurônio definitivo. No

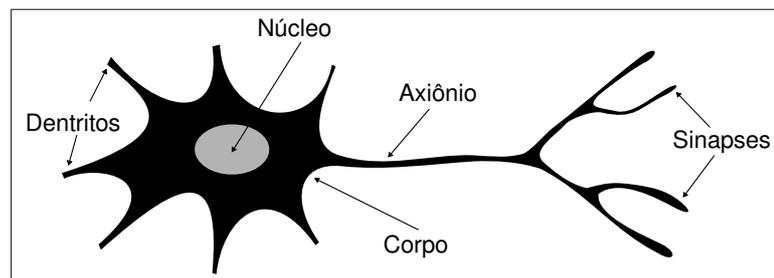


Figura 3.1: Aspecto geral do neurônio biológico.

entanto, iremos apresentar as propriedades comuns à maioria dos neurônios. O neurônio apresentado na Figura 3.1 é uma ilustração do aspecto de um neurônio presente na medula espinhal de um mamífero, como o homem. Sua forma básica é constituída do corpo, do qual se projetam ramificações chamadas dendritos. Neste corpo existe um núcleo (comum a todas as células vivas) que controla o funcionamento da célula. A junção do corpo com os dendritos formam a superfície de entrada de estímulos da célula. Em um dos extremos do corpo se projeta uma ramificação mais alongada chamada axônio, e no extremo do axônio surgem ramificações adicionais chamadas de terminais nervosos. Estes terminais nervosos geralmente estão ligados a outras células nervosas através de seus dendritos. A união entre uma terminação nervosa e um dendrito de uma outra célula é chamada de sinapse. Na verdade, em um axônio, ao longo de seu prolongamento, podem existir diversas sinapses com outras células, não apenas no seu extremo.

Quanto à forma de comunicação dos neurônios, embora o fluxo possa ir em qualquer direção sobre o axônio, as células tendem a se comunicar a partir dos dendritos, ou do corpo, através do axônio até as terminações nervosas, quando, por meio das sinapses, transmitem a informação para outra célula. O axônio pode ser muito longo. Por exemplo, o corpo celular de um neurônio que controla o dedão do pé encontra-se na medula espinhal e, portanto, tem um axônio que se estende ao longo do comprimento total da perna. Podemos contrastar a imensa extensão do referido neurônio com o tamanho muito pequeno de outros como os das células da retina, que possuem ramificações cujos dendritos e axônios não podem ser adequadamente diferenciados, pois, ambos são curtos e podem comunicar-se em qualquer direção.

### 3.3 A história do estudo das Redes Neurais

A história fundamental do estudo do cérebro e das redes neurais se restringe, principalmente, a apenas três itens: estudos de McCulloch e Pitts (1943), Hebb (1949),

e Rosenblatt (1958). Estas publicações introduziram, respectivamente, o primeiro modelo de redes neurais como "máquinas computacionais", o modelo básico das redes auto-organizáveis, e o modelo de "aprendizagem com um professor". Antes disso, no ano de 1931, Kurt Gödel publicou seu famoso Teorema da Incompletude (vide Arbib de 1987, para uma prova de que o Teorema de Gödel estabelecia um limite para a inteligência da máquina). O "programa formalista" iniciado por David Hilbert, que pretendia por toda a verdade matemática dentro de um único sistema formal, havia atingido a sua expressão máxima no "Principia Mathematica" de Whitehead e Russell. Mas Gödel mostrou que, se usada a abordagem oferecida em "Principia Mathematica" para configurar axiomas consistentes para a aritmética, e provar teoremas por dedução lógica a partir delas, a teoria deve estar incompleta, não importa com quantos axiomas se inicie - devem existir regras aritméticas que não podem ser deduzidas a partir dos axiomas.

Seguindo esse estudo Gödel, muitos matemáticos tem procurado formalizar a noção de um procedimento eficaz, do que pode e do que não pode ser feito de forma explícita na sequência de um algoritmo ou conjunto de regras. Kleene (1936) desenvolveu a teoria das funções recursivas parciais; Turing (1936) desenvolveu suas máquinas; Church (1941) desenvolveu o cálculo lambda, o precursor da linguagem de processamento de McCarthy, LISP, a favorita para os pesquisadores de inteligência artificial (IA); enquanto Emil Post (1943) apresentou os sistemas de reescrita de cadeias de símbolos, Chomsky iniciou seu sistema de formalização de gramáticas em 1959. Felizmente, estes métodos provaram ser equivalentes. O que quer que pudesse ser computado por um desses métodos poderia ser computado por qualquer outro método, se fosse equipado com o "programa apropriado". Deste modo, passou-se a acreditar na tese de Church de que se uma função pode ser computada por uma máquina, poderia ser calculado por qualquer um desses métodos.

Turing (1936) ajudou a traçar os limites do que poderia ser computável com o seu modelo do que chamou de máquina de Turing, um dispositivo que possuía um conjunto finito de instruções para ler, escrever, e mover sobre uma fita finita, porém, extensivelmente indefinida, sendo que cada parte dessa fita poderia ser gravada com um símbolo de algum alfabeto finito. Como um dos ingredientes da tese de Church, Turing ofereceu uma "psicologia do que é computável", tornando plausível a alegação de que qualquer cálculo efetivamente definível, isto é, qualquer coisa que um ser humano pode fazer na forma de manipulação simbólica, seguindo um finito e completo conjunto explícito de regras, poderia ser realizado por essa máquina equipada com um programa adequado. Turing também forneceu o exemplo mais famoso de um problema não computável, qual seja, "The unsolvability of the halting problem". Seja  $p$  o código numérico para um programa na máquina de Turing, e  $x$  o conteúdo inicial da fita da máquina de Turing. Então, a função

"halting"  $h(p, x) = 1$  se  $p$  eventualmente travar e se iniciar com um dado  $x$ , caso contrário será 0. Turing mostrou que não havia programa computacional capaz de calcular  $h$ .

Em 1943, o ano-chave da aproximação das noções de mecanismo de controle e autômatos inteligentes, McCulloch e Pitts uniu os estudos de neurofisiologia e lógica matemática. O seu modelo formal de neurônio como uma unidade lógica limiar foi construído sobre a doutrina do neurônio de Ramón y Cajal e as sinapses excitatórias e inibitórias de Sherrington, usando a notação da lógica matemática de Whitehead, Russell e Carnap. McCulloch e Pitts, desde a "fisiologia do computável", mostrou que a caixa de controle de qualquer máquina de Turing poderia ser implementada por uma rede dinâmica de seus neurônios formais. As idéias de McCulloch e Pitts influenciaram John von Neumann e seus colegas quando eles definiram a arquitetura básica do programa armazenado de computação. Assim, como computadores eletrônicos foram construídos para o fim da II Guerra Mundial, foi entendido que o que eles poderiam fazer poderia ser feito por uma rede de neurônios.

O livro de Craik (1943), "A Natureza da Explicação", viu o sistema nervoso como uma máquina de calcular capaz de modelar ou simular eventos externos, sugerindo que o processo de formação de um "modelo interno", que simula o mundo, é uma característica fundamental do pensamento e explicação. No mesmo ano, Rosenblueth, Wiener e Bigelow publicaram o "Comportamento, Propósito e Teleologia".

Todos estes estudos lançaram as bases para o surgimento da cibernética, que por sua vez deu origem a uma série de distintas disciplinas novas, como a IA, a teoria de controle biológico, a psicologia cognitiva e a modelagem neural, e cada uma seguiu caminhos diferentes na década de 1970.

### 3.4 Neurônio artificial

O modelo de neurônio mais elementar foi o criado por McCulloch e Pitts em 1943, que combinou a inspiração biológica com a lógica matemática [Haykin 2001], usando a propriedade tudo-ou-nada em uma lógica binária válida em tempos discretos de análise. Este trabalho demonstrou como os estados de excitação, inibição e limiar poderiam ser usados para construir uma grande variedade de neurônios artificiais. Este foi o primeiro estudo sobre neurônios artificiais que lançou as bases necessárias ao uso moderno desse tipo de lógica. A ideia básica desse modelo é dividir o tempo em unidades comparáveis a um período de tal forma que, em cada período de tempo, no máximo, um impulso pode ser gerado na saída do neurônio.

O neurônio de McCulloch-Pitts (Figura 3.2), portanto, opera em uma escala de tempo

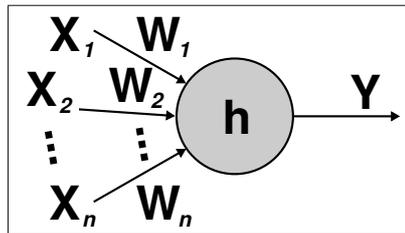


Figura 3.2: Modelo de McCulloch-Pitts para um neurônio artificial.

discreto,  $t = 0, 1, 2, 3, \dots$ , onde a unidade de tempo é (na biologia) na ordem de um milissegundo. Nós escrevemos que a saída do neurônio equivale a  $y(t) = 1$  se um impulso surgir na saída no espaço de tempo  $t$ , e  $y(t) = 0$ , caso contrário. Cada ligação, ou sinapse, a partir da saída de um neurônio para a entrada de outro tem um peso anexado. Digamos que  $W_i$  é um peso anexado a  $i$ -ésima sinapse de um determinado neurônio. Chamamos de sinapse excitatória se  $W_i > 0$ , sinapse inibitória se  $W_i < 0$  e indiferente se  $W_i = 0$ . Também associamos um limiar  $h$  em cada neurônio e assumimos que o impulso será considerado apenas se a soma de todas as sinapses em um mesmo tempo produzir um valor acima desse limiar. Formalmente, se no tempo  $t$  o valor da  $i$ -ésima entrada é  $x_i(t)$  e a saída um passo adiante é  $y(t + 1)$ , como vemos na Equação 3.1.

$$y(t + 1) = 1, \text{ se e somente se } \sum_i w_i x_i(t) \geq h \quad (3.1)$$

A Figura 3.3 mostra como os pesos e o limiar podem ser ajustados para produzir neurônios que realizam as funções lógicas E, OU e NÃO. Desta forma, percebemos que os neurônios, tal como foram modelados por McCulloch-Pitts, são suficientes para construir redes que podem funcionar como os circuitos de controle para um computador realizar cálculos de complexidade arbitrária. Esta descoberta teve papel fundamental no desenvolvimento da teoria de autômatos e no estudo de máquinas de aprendizagem. Embora o neurônio de McCulloch-Pitts já não modele com tanta precisão o neurônio biológico atual, devido aos avanços da neurociência, ainda é amplamente utilizada para a computação neural, sobretudo em problemas cujo o resultado de interesse está compreendido no intervalo  $[0, 1]$  e a função  $f(\sum_i w_i x_i(t))$  que calcula a saída para o tempo seguinte  $y(t + 1)$  é uma função variável contínua, e não uma função degrau.

Percebamos que, uma coisa é definir um modelo lógico com poder suficiente para cálculo de qualquer problema no domínio da computação discreta, e outra, bem diferente, é entender o funcionamento dos neurônios em cérebros reais. Geralmente o problema reside em decidir como essas unidades lógicas devem ser representadas, pois, tentar apro-

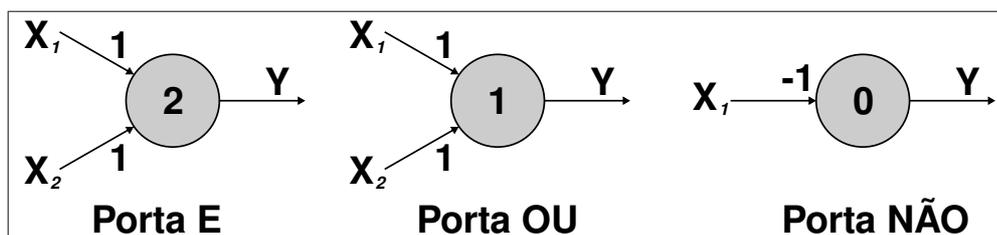


Figura 3.3: Neurônios artificiais simulando portas lógicas E, OU e NÃO.

ximar muito os neurônios artificiais dos neurônios reais tornará a resolução de problemas computacionais extremamente difícil. Por outro lado, podemos dizer que a computação neural não pode atingir o seu pleno poder sem aplicar novos mecanismos com base no estudo atual e futuro de redes neurais biológicas.

### 3.5 Perceptrons

Perceptrons são redes neurais que mudam com a experiência, usando uma regra de correção de erro projetada para alterar os pesos sinápticos de cada unidade de resposta quando estas fornecem respostas erradas aos estímulos de entrada da rede. Nos referimos ao julgador que diz o que está correto como "Professor", no entanto a figura do professor pode ser desempenhada por outra rede neural, alguma entrada do ambiente ou até mesmo a uma informação registrada por um humano. Considere o caso em que um conjunto de  $R$  linhas de entrada alimenta uma rede neural cujos neurônios de McCulloch-Pitts são chamados de unidades associadoras e que, por sua vez, fornecem os dados para um único neurônio (chamado de unidade de saída do perceptron) com pesos ajustáveis  $(W_1, \dots, W_n)$  e o limiar  $h$ . No caso de reconhecimento de padrões visuais, pensamos em  $R$  como uma retina "retangular" para que padrões possam ser projetados. Um perceptron simples é aquele em que as unidades associadoras não estão interligadas, o que significa que ele não tem memória de curto prazo. Se tais conexões estão presentes, o perceptron é chamado cruzado. Se as unidades associadoras alimentarem o padrão  $x = (x_1, \dots, x_n)$  na unidade de saída, então, a resposta que a unidade oferecerá será através da função  $f(x) = w_1x_1 + \dots + w_nx_n - h$ . Em outras palavras, o perceptron simples só pode computar uma função linearmente separável do padrão. A pergunta feita por Rosenblatt (1958) e respondida por muitos outros foi: "Dado um perceptron simples (ou seja, apenas os pesos sinápticos da unidade de saída são ajustáveis), podemos treiná-lo para reconhecer um determinado conjunto de padrões linearmente separáveis, ajustando o peso sobre interconexões diferentes usando unicamente a experiência de que a rede classifica ou não

um padrão corretamente?" A resposta foi "Sim: se os padrões são linearmente separáveis, então, há um regime de aprendizagem que acabará por produzir um conjunto de pesos que classificará esses padrões."

Observemos, finalmente, que o teorema da convergência do perceptron mostra que o procedimento de Rosenblatt não executará indefinidamente, mas, acabará por encontrar os valores adequados para as sinapses, mesmo que depois de executar muitas iterações através do conjunto de padrões.

### 3.6 Redes Neurais de Função de Base Radial

As Redes de Função de Base Radial (RBF) são ao mesmo tempo simples e poderosas, com aplicação em vários campos de pesquisa e extremamente estudadas por vários nomes da ciência [Rahmatulloh 2009]. Dentre outros, podemos citar os trabalhos de Buhmann (2000), que faz uma abordagem matemática das RBF, Pontil e Poggio (2000), na aplicação em Máquinas de Vetor de Suporte, e Lowe (1999), sob um ponto de vista estatístico.

Para entendermos a diferença básica entre as Redes RBF e as estruturas clássicas de Perceptron de Múltiplas Camadas (MLP), vejamos o que ilustra a Figura 3.4 em uma visão bi-dimensional de um problema hipotético. Dado um agrupamento de dados, existem duas formas de classificar seus conglomerados através de uma rede neural, a primeira, é usando a abordagem fronteira e procurando encontrar regiões onde possamos, literalmente, traçar hiper-planos de separação desses dados. O resultado disso é que teremos estruturas poligonais de separação indicando os limites de classificação para cada grupo de dados, e onde eles se encontram. A outra forma, é descrever esses conjuntos de dados através de uma função de densidade de probabilidade que represente esses dados. Esta função se posiciona na região de maior aglomeração dos dados, sendo ela própria o descritor de classificação. A primeira abordagem é realizada pelas redes de perceptrons de múltiplas camadas, e a segunda pelas redes RBF. Não existe uma abordagem definitiva para os problemas de classificação, o que ocorre é que ambas abordagens se complementam, cada uma com suas vantagens e desvantagens.

A RBF simula uma função de aproximação global, usando combinações de funções de base radial centradas nos vetores de peso. Esses vetores de peso podem ser escolhidos aleatoriamente, usando o próprio conjunto de dados, ou podem ser escolhidos de acordo com algum critério, ou como algoritmos de agrupamento (cluster), ou por densidade probabilística, ou, ainda, utilizando critérios de distância. Uma diferença adicional é que a RBF utiliza uma função de distância para transformar o vetor de entrada em um valor escalar na camada escondida da rede.

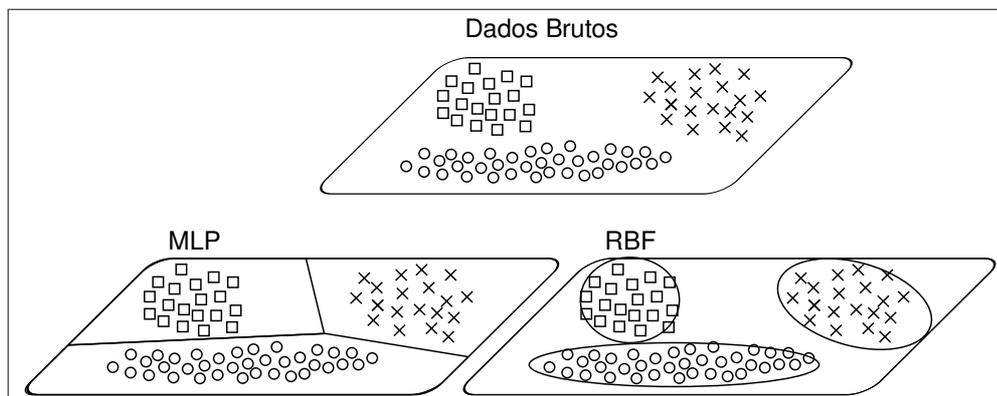


Figura 3.4: Representação da classificação tomada pela RBF em relação a tomada pela MLP.

Embora tenha sido inicialmente concebida para aproximação de funções (Broomhead and Lowe, 1988; Powell, 1992), as RBFs tem sido utilizadas para uma variedade de aplicações de forma eficiente e robusta, tais como, em processamento estatístico de padrões (Lowe, 1999), regressão e regularização (Evgeniou et al., 2000), formação de padrões biológicos (Logothetis, Pauls, e Poggio, 1995), mapeamento da presença de dados ruidosos e, mais recentemente, em termos de aproximações "kriging" (Wan e Bone, 1997) e em máquinas de vetor de suporte.

No entanto, além de exibir uma série de propriedades teóricas úteis, a estrutura da rede RBF é sobretudo importante para construções práticas, pois, pode ser aplicada eficazmente em problemas de discriminação (como reconhecimento de voz), na previsão de séries temporais (como modelagem econômica), e na extração de características ou mesmo em mapeamento topográfico (como na codificação do espaço sensorial de um nariz artificial na análise química de gases).

### 3.6.1 A estrutura básica de uma RBF

A RBF é uma rede de apenas uma camada oculta, alimentada para frente, com função de ativação linear, na camada de saída, e função de transferência não linear, na camada oculta, por isso suporta vários tipos de não linearidades (Figura 3.5). Os parâmetros ajustáveis são os vetores de peso de cada camada da RBF. Os primeiros vetores-peso são os que interligam os neurônios da camada escondida com os neurônios da camada de saída ( $\vec{S}_{jk}$ ), os outros são os que interligam o vetor de entrada com os neurônios da camada escondida ( $\vec{W}_{ij}$ ). Sobre estes últimos, podemos imaginá-los como sendo vetores de suporte dos padrões de entrada, tanto podendo associá-los como centros dos agrupamentos

da amostra, como representantes do conjunto de dados de treinamento.

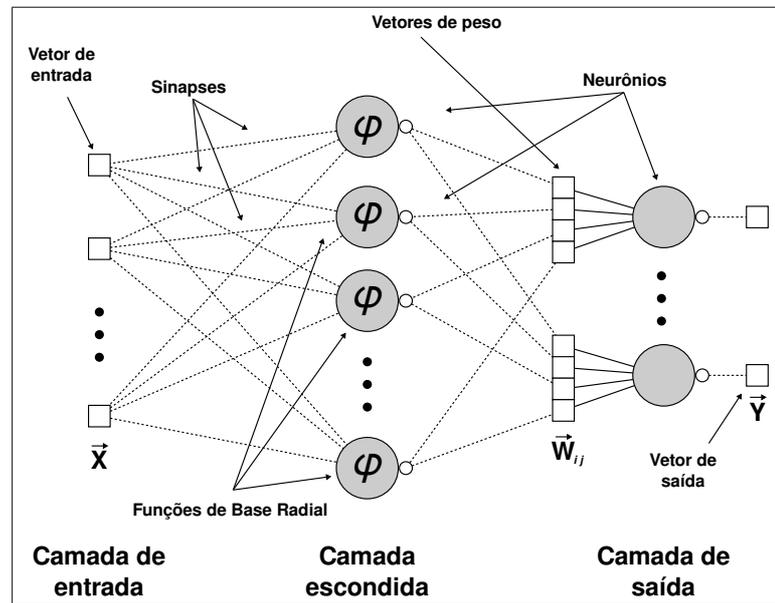


Figura 3.5: Estrutura básica de uma Rede de Função de Base Radial.

A fundamentação matemática da RBF demonstra que  $k$ -ésimo componente do vetor de saída  $y_p$  corresponde ao  $p$ -ésimo padrão de entrada  $x_p$  expresso pela Equação 3.2.

$$[y(x_p)]_k = \sum_{j=0}^h S_{jk} G_j(\|x_p - W_j\|; \Sigma) \quad (3.2)$$

Onde  $G$  corresponde a função de transferência não-linear no neurônio  $j$  escondido. Nesta função usa-se, geralmente, uma função de transferência com características gaussianas, como por exemplo,  $G(z) \approx \exp - [z^T \Sigma^{-1} z]$ . Uma característica diferenciada das redes RBF é que a saída  $h$  do  $j$ -ésimo neurônio da camada escondida é dado por uma função radial da distância entre o vetor de entrada e cada vetor de peso  $W_{ij}$ , dado por,  $h_j = G_j(\|x - W_j\|)$ .

Uma das vantagens da RBF é que a primeira camada de pesos pode ser, muitas vezes, determinada empiricamente, a partir de conhecimento prévio ou de técnicas simples. Percebe-se que vetores posicionados sobre as maiores massas de dados apresentam melhor resultado de classificação por representarem melhor o conjunto de dados. Uma técnica simples de posicionamento dos centros é através do processo de aglomeração "K-means", que procura ajustar os vetores para os centros dos grupos dos dados, usando uma métrica de distância média. Portanto, uma vez que os pesos associados com a primeira camada foram especificados, o grande problema na formação de uma rede RBF é focado em

determinar os pesos da camada final. Como a rede RBF é normalmente empregada para executar uma tarefa de classificação supervisionada ou de predição, tal como predição de séries temporais, esta construção assume normalmente a forma de otimizar uma função de custo exigida na saída da rede, que de alguma forma se aproxima de um conjunto de valores-alvo conhecidos. É comum a tentativa de minimizar a média do quadrado do erro, embora outras funções de custo possam ser empregadas.

### 3.6.2 Uso de RBF para classificação de padrões

Um dos usos mais comuns de uma rede RBF é a sua utilização como um classificador capaz de modelar as fronteiras não-lineares entre os agrupamentos. Vamos comentar a seguir sobre como a rede RBF se insere neste ponto de vista estatístico, apesar de historicamente ter surgido a partir da teoria da interpolação.

Na classificação estamos interessados, principalmente, na probabilidade  $p(C|x)$ , ou seja, a probabilidade de  $x$  pertencer a classe  $C$ , no entanto, é mais fácil modelar outros aspectos dos dados, tais como a distribuição incondicional dos dados,  $p(x)$ , e a verossimilhança dos dados,  $p(x|C)$ , que é a probabilidade de que os dados foram gerados uma vez que vieram de uma determinada classe  $C$ . Podemos, então, recriar estas quantidades de acordo com o teorema de Bayes,  $p(C_i|x) = p(C_i)p(x|C_i)/p(x)$ . A distribuição dos dados é modelada como se fosse gerada por uma junção de distribuições, ou seja, uma combinação linear de estados parametrizados ou de funções de base radial, como Gaussianas. Já que conjuntos de dados individuais para cada classe não são susceptíveis de serem aproximados por uma única distribuição de Gauss, precisamos de um grupo de várias funções de base radial. Supomos que a verossimilhança e a distribuição incondicional podem ser modeladas pelo mesmo conjunto de distribuições,  $q(x|s)$ , mas com diferentes coeficientes, ou seja,  $p(x) = \sum_s \hat{p}(s)q(x|s)$  e  $p(x|C_i) = \sum_s p(s;i)q(x|s)$ . Então, a quantidade que estamos interessados em  $p(C_i|x) = p(C_i)p(x|C_i)/p(x)$  é dada pela Equação 3.3.

$$p(C_i|x) = \sum_s \frac{p(C_i)p(s;i)}{\hat{p}(s)} \frac{\hat{p}(s)q(x|s)}{\sum_{s'} \hat{p}(s')q(x|s')} = \sum_j \lambda_{ij} \phi(x|j) \quad (3.3)$$

Onde  $\lambda_{ij} = p(C_i)p(j;i)|\hat{p}(j)$  descreve a significância geral do estado  $j$  para a classe  $i$ , e  $\sigma(x|j)$  é a função de base normalizada,  $\hat{p}(j)q(x|j)/\sum_j \hat{p}(j)q(x|j)$ .

A arquitetura da rede RBF é dada por um total de funções  $h$  usadas para aproximar a verossimilhança e a densidade incondicional. O estado final dos pesos calculados nos dá o grau de significância dos nós da camada oculta em relação a camada de saída. Claro que a localização e os intervalos de influência devem ser ajustados para permitir a clas-

sificação adequada dos dados. Isto pode ser conseguido usando técnicas de classificação não supervisionadas.

Desta forma, a rede RBF é apropriada para classificação de padrões. Perceba que a rede RBF, para estimação de densidade, é mais abrangente do que a mencionada aqui, em especial, não é necessário que a função de base seja necessariamente uma função de densidade de probabilidade.

### 3.6.3 Uso de RBF para predição

Na formulação probabilística da RBF, sua estrutura é interpretada através da estatística da distribuição dos dados, nesse caso a função densidade de probabilidade é amostrada estocasticamente. No entanto, a formulação original da rede RBF foi desenvolvida a fim de produzir um mapeamento determinístico de dados através da aproximação de funções tradicionais. Esta abordagem introduz o conceito de que a construção de uma rede RBF pode ser vista com um "ajustador" de curvas. Desta forma, podemos dizer que a interpretação natural desta rede é a de um generalizador ou um interpolador de superfícies.

A idéia básica é a seguinte. Suponha que temos um conjunto de entrada/saída de pares de entrada/padrões alvo representando dados de uma superfície desconhecida  $N \times C$ . Como um exemplo simples, considere um conjunto de pares  $(x, y)$  gerados de acordo com  $y = x^2$ . Nesta abordagem, o problema é escolher uma função  $y : N \in C$ , que satisfaça as condições de interpolação  $y(x_p) = t_p, p = 1, 2, \dots, P$ . Esta é a interpolação estrita, em que a função é obrigada a passar por todos os pontos de dados conhecidos. A estratégia de interpolação é construir um espaço de função linear combinada com um conjunto de funções de base radial ajustadas com os dados conhecidos. A expansão da função de base radial para uma dimensão originalmente pode ser expressa pela Equação 3.4. Usando as condições de interpolação o parâmetro de ajuste  $\lambda$  pode ser determinado por métodos de matriz inversa. Estas equações podem facilmente serem generalizadas para o caso de múltiplas dimensões.

$$y(x) = \sum_{j=1}^P \lambda_j \sigma(\|x - x_j\|) \quad (3.4)$$

No caso da função simples  $y = x^2$ , as entradas são os valores de  $x$  e os alvos são os valores de  $y$ , a rede RBF é construída de tal forma a ajustar sua superfície de modo a adequar-se a esta parábola. Esta parábola pode ser considerada um generalizador da função e pode produzir outros pares de dados de forma a respeitar a função original. Na medida em que a função a ser aproximada seja suave e não linear, podemos aproximá-la com uma certa eficiência usando uma rede RBF. Isto explica o porquê que redes RBF

podem prever, com alto grau de certeza, sistemas caóticos de séries temporais, pois, embora aparentem ser aleatórios, o que está por trás são amostras de dados geralmente bem suaves.

### **3.7 Comentários finais**

Vimos neste capítulo que as redes RBFs podem ser empregadas em tarefas de classificação, de previsão de séries temporais, e de extração de características. Por causa de sua tratabilidade computacional, as RBFs tem sido aplicadas a diversos problemas do mundo real, e há algumas evidências de que sua estrutura pode ter semelhanças com aspectos biológicos de tratamento de visão. Mas, acima de tudo, a sua força e utilidade derivam da sua simplicidade e de uma estreita relação com outras áreas de processamento de sinal, detecção de padrões e outras arquiteturas de rede neural. Essas conexões e interpretações ainda estão para serem descobertas.

Veremos no próximo capítulo como os conhecimentos de Processamento Digital de Imagens, Visão Computacional, Redes Neurais e sobretudo RBFs foram unidos na missão de produzir um cálculo de disparidade de imagens estéreo mais ágil e eficaz.

---

## Capítulo 4

# Método Proposto

---

A área de visão computacional consiste hoje numa disciplina derivada de processamento digital de imagens, que tomou uma abrangência tal que pode ser considerada como um campo totalmente separado de pesquisa. Inseridos nesta área estão vários problemas relacionados em como seres biológicos processam a informação da visão, reconhecem coisas e tomam decisões. Uma característica notada é que a maioria dos seres biológicos possuem um sistema de visão que utiliza dois olhos para enxergar. Essa característica, ou seja, a visão binocular, é o que dota o ser da capacidade de processar a noção de profundidade da cena. Percebido isso, uma das mais interessantes linhas da visão computacional é o desenvolvimento de recursos computacionais capazes de imitar a característica acima citada, a fim de que tal sistema também seja capaz de perceber as diferentes profundidades existentes em uma cena.

Precisamos entender, ainda, que a área de visão computacional é na verdade uma confluência de diversas áreas do conhecimento, tais como o próprio processamento digital de imagens, redes neurais, neurociência, ótica, geometria, álgebra e outros, sendo, portanto, uma área bastante complexa de estudo justamente por envolver vários aspectos da ciência e uma vasta gama de conceitos.

Além disso, percebe-se que os recursos computacionais existentes processam a informação de forma bem mais lenta do que, digamos, a velocidade com que nosso cérebro a processa, ou seja, nós, seres biológicos, conseguimos, numa fração de segundos, perceber coisas como um copo sobre a mesa, um lápis caindo, uma bola arremessada em nossa direção, e outras coisas triviais do nosso cotidiano. Todavia, para processar a mesma informação em um computador, mesmo moderno, essas tarefas não são tão simples, e demoram muito mais do que uma fração de segundos para chegarmos às mesmas conclusões. Portanto, todas as tarefas em visão computacional, mesmo as mais básicas, são demoradas em comparação à velocidade que nosso cérebro as processa, e necessitam de formas mais ágeis de resolução. Esse foi um motivo que levou esse trabalho a desenvolver

um método, cuja principal contribuição é o ganho na velocidade de processamento.

## 4.1 Comentários iniciais sobre o método

Quando começamos a desenvolver esse método, nossa ideia era produzir um procedimento capaz de ter como resultado uma noção dos objetos da cena em uma representação tridimensional. Portanto, usaríamos a visão computacional e seus conceitos como base para o desenvolvimento deste método. Percebemos, assim, que algum tipo de visão estéreo deveria ser usada, e optamos por utilizar a visão binocular com câmeras paralelas, devido à sua simplicidade no cálculo das correlações, e por não se fazer necessário a retificação das imagens adquiridas, uma vez que a linha epipolar se encontraria na mesma posição  $y$  nas duas câmeras.

Outro recurso que utilizamos para desenvolver nosso trabalho para fins de simplificação foi o uso de um dataset [Scharstein e Pal. 2007] [Hirschmüller e Scharstein 2007], um conjunto de imagens preparadas em um cenário controlado de luz, com o objetivo de eliminar a etapa de calibração das câmeras e outros requisitos necessários para auferir os parâmetros intrínsecos e extrínsecos das câmeras utilizadas. Desta forma, utilizando o dataset, temos disponível todos esses parâmetros de forma precisa, e podemos utilizá-los no nosso algoritmo de visão estéreo.

Por fim, é importante registrar que nenhuma forma nova de correlação das imagens foi utilizada. Como nosso interesse neste trabalho era apenas ganhar velocidade de processamento, utilizamos os algoritmos tradicionais de correlação como parâmetro para as modificações que se seguiriam, e comparamos se tais modificações surtiriam algum tipo de melhoria em relação a estes algoritmos tradicionais. Maiores detalhes sobre os algoritmos de correlação podem ser vistos no Capítulo 2.

## 4.2 Hipóteses levantadas

Quando iniciamos essa pesquisa, conforme já mencionado, nosso objetivo era principalmente oferecer uma forma de processar os algoritmos de correlação de um modo mais eficiente, e assim produzir uma representação tridimensional da cena mais rapidamente.

Lembremos que o algoritmo de correlação tradicional tem um alto custo computacional, pois, associa cada pixel de uma imagem a um conjunto de todos os pixels vizinhos que estão dentro da área de abrangência da janela definida, comparando, posteriormente, esse vetor de valores com um conjunto de outros vetores similares, obtidos da mesma

forma na outra imagem. Esta grande quantidade de comparações, necessárias para buscar o par de pixels mais similares, é o principal motivo porque esses algoritmos demoram tanto para processar.

Pensando nisso é que nossa hipótese foi a seguinte: se eu preciso reduzir meu custo computacional, uma forma de conseguir isto é justamente não calculando todas as iterações necessárias ao algoritmo de correlação tradicional. Então, decidimos que iríamos obter uma amostra da imagem, pegando assim apenas alguns pontos para fazermos a correlação, descartando todos os demais pontos, os quais seriam depois obtidos usando algum outro método.

### 4.3 O procedimento adotado

Para explicarmos o método proposto vamos adotar o bem oportuno ditado que diz que "uma imagem fala mais do que mil palavras". Vejamos abaixo as descrições das etapas de processamento utilizadas para produzir o resultado do método e seus respectivos diagramas. As figuras (Figura 4.1 e Figura 4.3) mostram duas etapas distintas, a primeira compreendendo a leitura, a geração da malha de saltos e o cálculo das disparidades, e a segunda compreendendo o treinamento, simulação e interpolação da imagem obtida na primeira etapa.

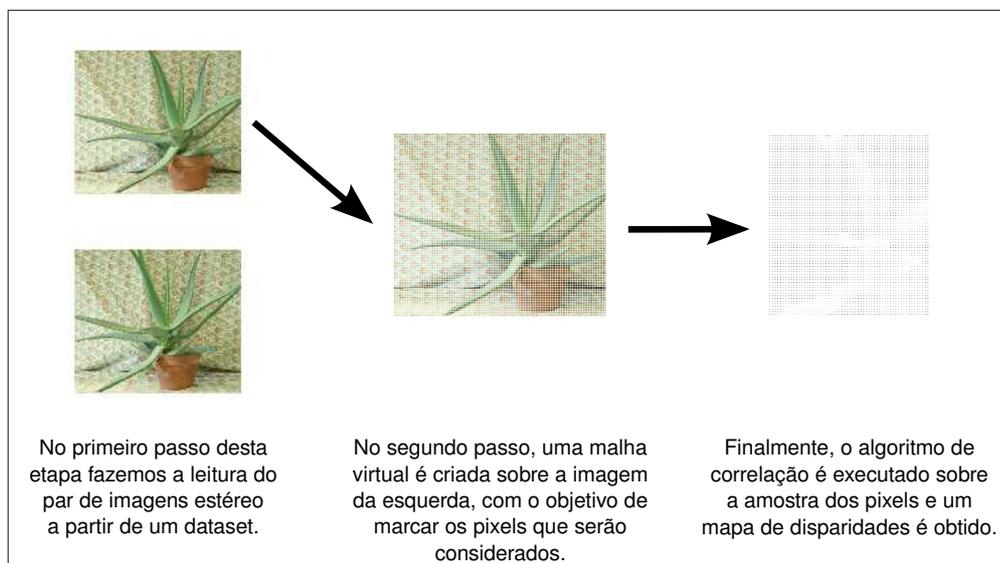


Figura 4.1: Primeira parte do diagrama do método proposto.

Nesta primeira etapa do método (Figura 4.1), observamos que um par de imagens é adquirida a partir de um dataset e carregada em memória, depois disto, é gerada uma ma-

lha virtual que sobrepõe esta imagem a fim de marcar os pontos que serão amostrados. Esta malha leva em consideração um parâmetro de salto para ajustar sua densidade, ou seja, quanto maior o salto, maior será o afastamento entre as linhas da malha. Posteriormente, serão considerados, para o algoritmo de correlação, apenas os pixels localizados nos pontos onde as linhas da malha se cruzam, fazendo com que tenhamos uma imagem de menor resolução para cálculo das disparidades, e finalmente, no último passo desta etapa, as disparidades são calculadas para estes pontos usando o algoritmo de correlação baseada em área.

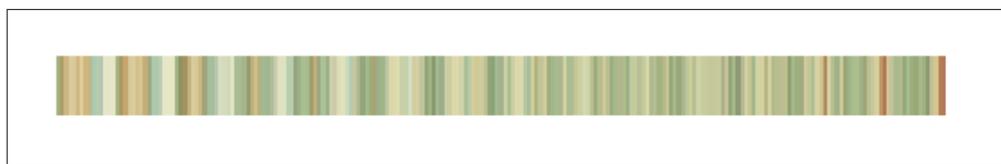


Figura 4.2: No algoritmo de correlação baseado em área, um vetor contendo todos os pixels vizinhos pertencentes a janela é considerado no momento da leitura do ponto.

Percebamos que embora uma amostra tenha sido retirada da imagem, para fins de cálculo das disparidades, todos os pontos estão sendo considerados, uma vez que o algoritmo de correlação baseado em área agrupa em um mesmo local todos os pixels vizinhos compreendidos na janela daquele ponto (Figura 4.2).

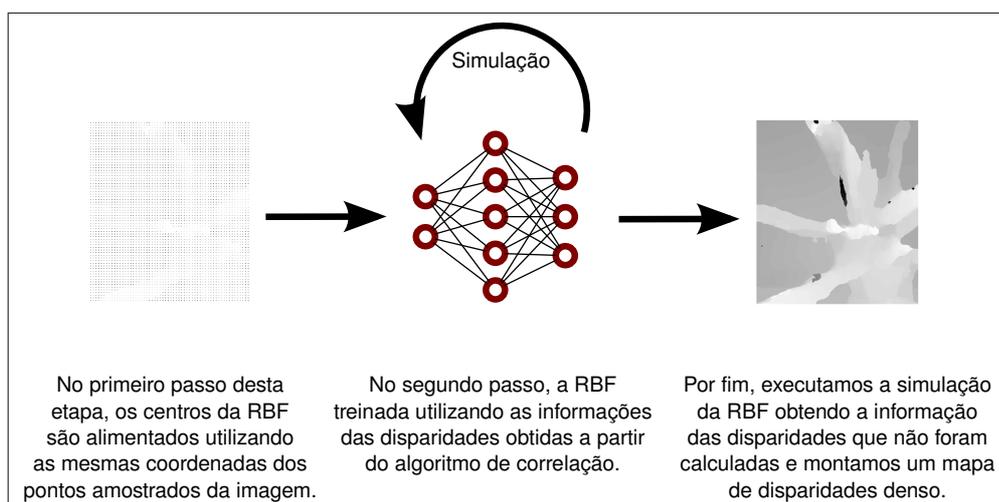


Figura 4.3: Segunda parte do diagrama do método proposto.

Na segunda etapa do método (Figura 4.3), necessitamos recuperar as informações que foram descartadas na etapa anterior para aumentar a velocidade de execução do algoritmo.

Para isso, usamos uma rede neural RBF como um interpolador dos dados. Inicialmente, a rede neural é carregada atribuindo os vetores das coordenadas dos pontos amostrados na primeira etapa aos pesos da camada escondida. Esses vetores correspondem aos centros das funções de base radial. Em seguida, treinamos a rede neural com o conjunto das disparidades que foram calculadas, fazendo assim uma preparação para o processo de interpolação posterior do mapa de disparidades completo.

É importante ressaltar que a rede neural terá, na sua camada escondida, a mesma quantidade de neurônios que os pontos da amostra do mapa de disparidades. Desta forma, o que acontecerá é que sobre o vetor de suporte localizado em cada ponto da amostra, será construída uma função de base radial, sendo que no final do treinamento se obterá uma superfície composta pela soma de todas as funções de base radial. Essa superfície é justamente o mecanismo que será utilizado para a obtenção dos valores interpolados dos pontos que não fazem parte da amostra inicial. O passo final desta etapa é simular a rede neural, informando as coordenadas dos pontos no mapa de disparidades nos quais não foi calculado utilizando o algoritmo de correlação, e ainda, armazenar esse resultado na matriz completa das disparidades.

A razão de utilizarmos um método de interpolação baseado em redes neurais provém do fato de que as redes neurais, diferentemente dos métodos de interpolação lineares, oferecem um recurso mais preciso de detecção de características não lineares, sendo, portanto, um modelo de cálculo mais eficiente para o caso da interpolação das disparidades. Outra característica é que a rede RBF forma, naturalmente, uma malha de interligação entre os neurônios, tornando mais fácil a reconstrução tridimensional da superfície, uma vez que os pontos já estão estruturados.

## 4.4 Comentários finais

Percebemos através deste procedimento que houve um ganho relativo de velocidade, pois, o algoritmo de maior complexidade, que é justamente o de correlação das imagens, deixou de ser executado totalmente, sendo necessário apenas o cálculo para alguns pontos. Percebemos, ainda, que a rede neural de função de base radial tem características extremamente eficientes para interpolação, além de boa performance na estimativa e predição dos valores interpolados. Todos esses resultados, tabelas comparativas e materiais utilizados nos ensaios do procedimento, poderão ser consultados e analisados no próximo capítulo, que trata justamente dos resultados.

---

# Capítulo 5

## Resultados

---

No capítulo anterior, vimos as características do método sugerido neste trabalho. O objetivo deste método é, principalmente, otimizar a velocidade do cálculo das disparidades de um par de imagens estéreo através de um algoritmo tradicional de correlação baseado em área. Neste capítulo, apresentaremos os resultados obtidos através da aplicação deste método, assim como os comparativos de erro e de velocidades entre o método tradicional e o sugerido.

### 5.1 Conjunto de entrada

Para realizar as experiências com o método sugerido, utilizamos um conjunto de imagens estéreo pré-fabricadas [Scharstein e Pal. 2007] [Hirschmüller e Scharstein 2007], isso pelo simples fato de eliminar certas etapas prévias ao cálculo das disparidades, tais como a retificação e a calibração.

Na finalização desta pesquisa, com o objetivo de provar o correto funcionamento do método proposto, executamos o algoritmo implementado usando todas as imagens do conjunto de imagens citado anteriormente. Todavia, para ilustrar os resultados obtidos, selecionamos três imagens do conjunto total que demonstram perfeitamente os resultados encontrados. Este conjunto de entrada é exatamente os pares apresentados nas Figuras 5.1, 5.2, 5.3 e 5.4.

Todas as imagens foram utilizadas em baixa resolução, a imagem "Aloe" com resolução de  $427 \times 370$  pixels colorida, a imagem "Midd1" com resolução de  $465 \times 370$  pixels colorida, a imagem "Bowling1" com resolução de  $417 \times 370$  pixels colorida, e a imagem "Lampshade1" com resolução de  $433 \times 370$  pixels também colorida.



Figura 5.1: Par "Aloe" de imagens estéreo originais que servirão de entrada para o algoritmo.



Figura 5.2: Par "Midd1" de imagens estéreo originais que servirão de entrada para o algoritmo.



Figura 5.3: Par "Bowling1" de imagens estéreo originais que servirão de entrada para o algoritmo.



Figura 5.4: Par "Lampshade1" de imagens estéreo originais que servirão de entrada para o algoritmo.

## 5.2 Disparidades sem interpolação

Para utilizarmos como parâmetro de comparação para os experimentos com o método, executamos o algoritmo clássico de cálculo de disparidades nas três imagens citadas, porém, sem usar interpolação. Isto nos dá um parâmetro para comparar nosso método em relação a erro e a tempo de execução.

Os mapas de disparidades obtidos são apresentados nas Figuras 5.5, 5.6, 5.7 e 5.8.

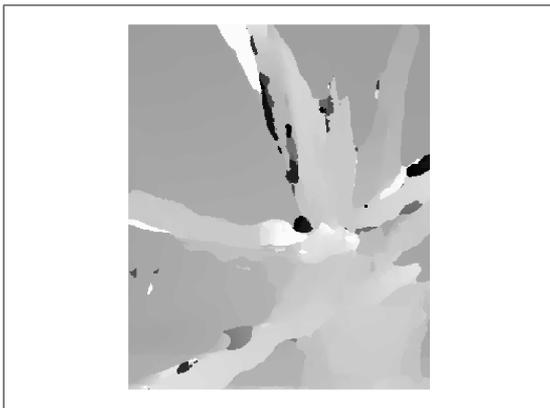


Figura 5.5: Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Aloe".



Figura 5.6: Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Midd1".



Figura 5.7: Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Bowling1".

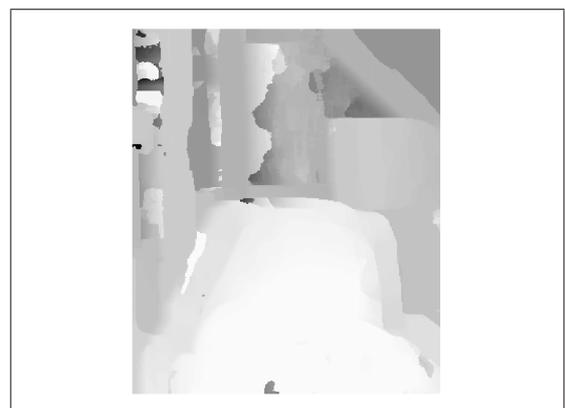


Figura 5.8: Mapa de disparidade sem interpolação, gerado a partir do par estéreo "Lampshade1".

### 5.3 Resultados do método

Conforme explicamos antes, um dos passos do método proposto é criar uma malha onde apenas alguns pontos são considerados. Neste experimento, criamos uma sequência de execuções do algoritmo para um mesmo par de imagens, aumentando a cada passo em 1 pixel o espaçamento da malha até o limite de 100 pixels de espaçamento, a fim de observar o comportamento do interpolador neural projetado.

O interpolador utilizado foi uma rede neural de função de base radial, que foi projetada para funcionar eficientemente para o nosso propósito, qual seja, interpolar dados que são similares em uma geometria tridimensional. A função de RBF usada foi a Multiquadrática, por apresentar melhor resultado de interpolação das arestas dos objetos 3D.

Apresentaremos a seguir os mapas de disparidades gerado pelo método, para os diferentes valores de espaçamento, utilizando uma representação tridimensional.

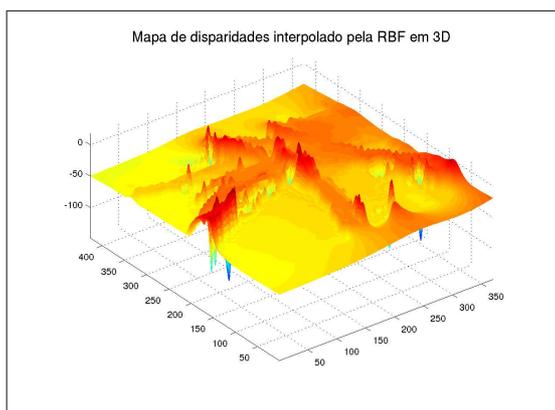


Figura 5.9: Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Aloe".

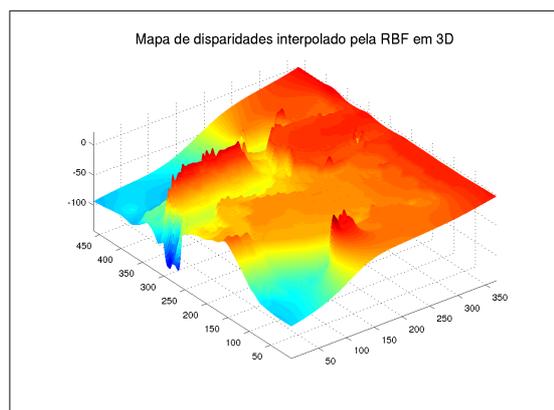


Figura 5.10: Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Midd1".

Notamos que a partir de uma certa quantidade de saltos, como esperado, a superfície gerada não mais retém as características da disparidade original (Figuras 5.21, 5.22, 5.23 e 5.24). Isto ocorre, principalmente, porque muito dado está sendo interpolado e os centros da RBF não mais são representativos para o conjunto total dos pontos da imagem. Isto faz com que o erro de aproximação aumente em relação ao mapa gerado originalmente, sem dados interpolados.

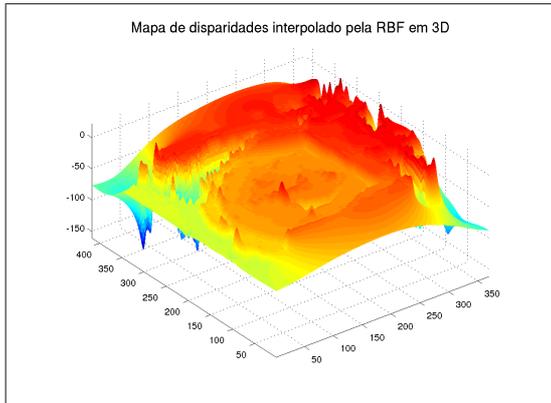


Figura 5.11: Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Bowling1".

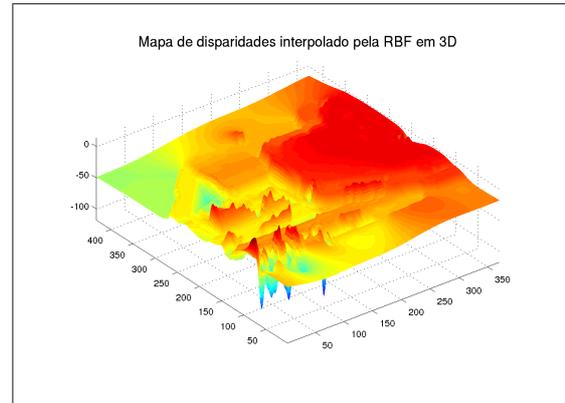


Figura 5.12: Mapa interpolado de disparidade usando espaçamento de 5 pixels, gerado a partir do par estéreo "Lampshade1".

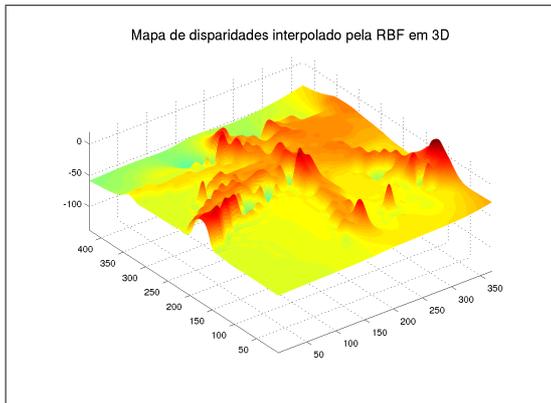


Figura 5.13: Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Aloe".

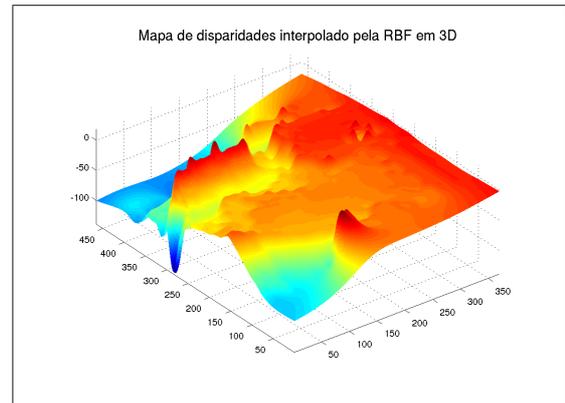


Figura 5.14: Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Midd1".

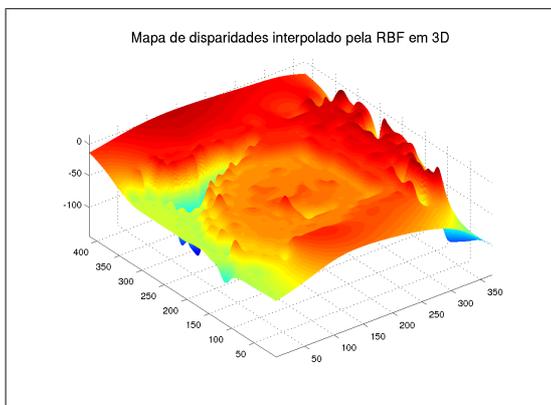


Figura 5.15: Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Bowling1".

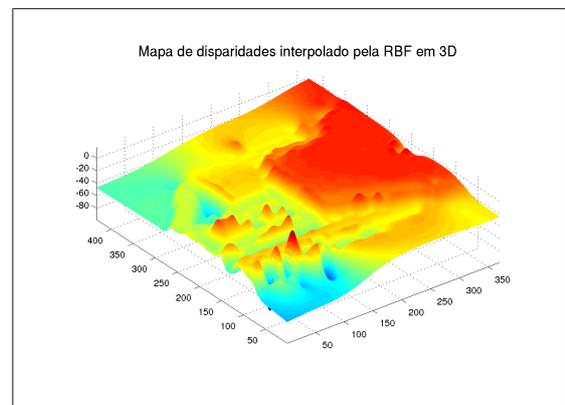


Figura 5.16: Mapa interpolado de disparidade usando espaçamento de 10 pixels, gerado a partir do par estéreo "Lampshade1".

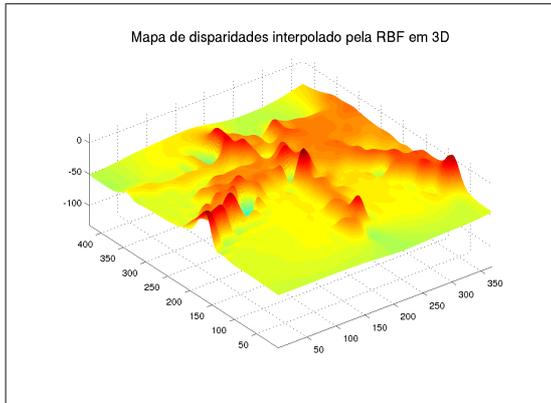


Figura 5.17: Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Aloe".

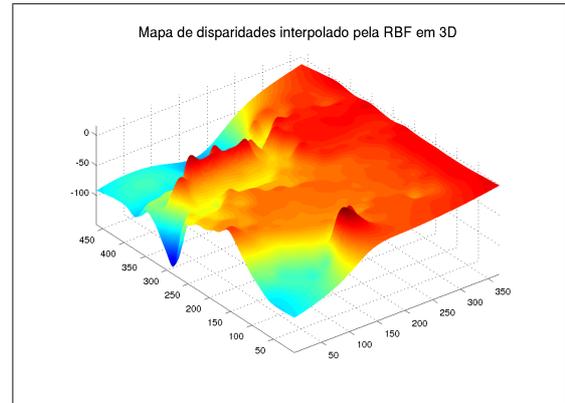


Figura 5.18: Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Midd1".

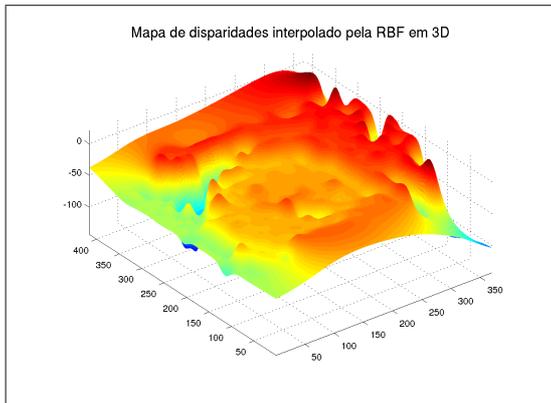


Figura 5.19: Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Bowling1".

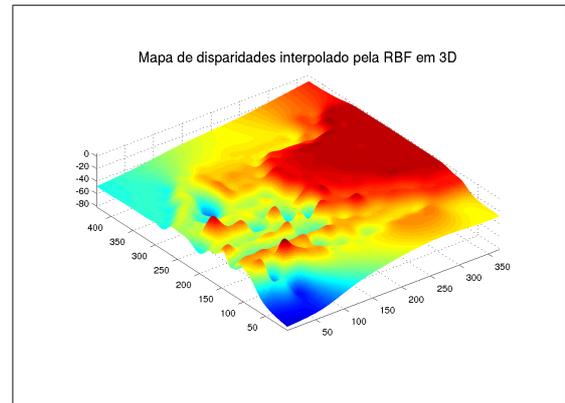


Figura 5.20: Mapa interpolado de disparidade usando espaçamento de 15 pixels, gerado a partir do par estéreo "Lampshade1".

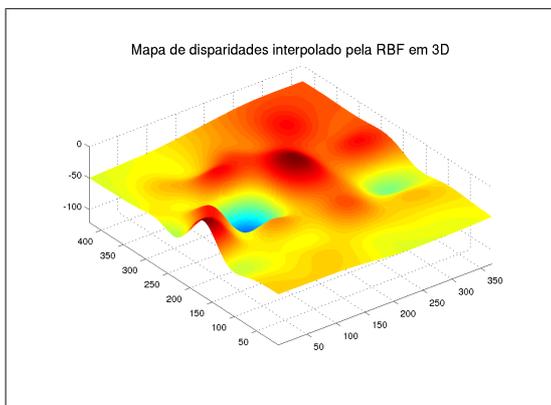


Figura 5.21: Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Aloe".

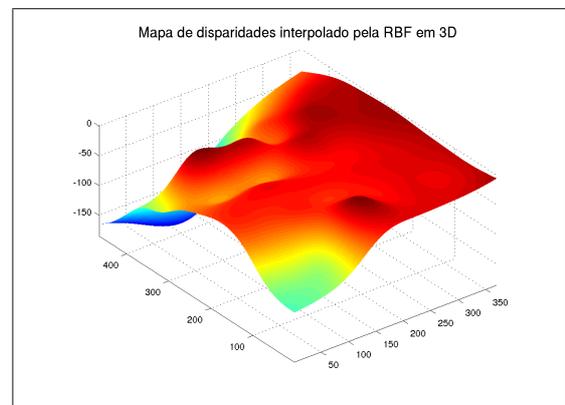


Figura 5.22: Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Midd1".

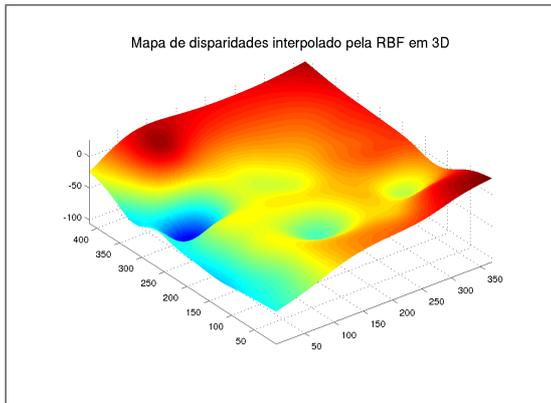


Figura 5.23: Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Bowling1".

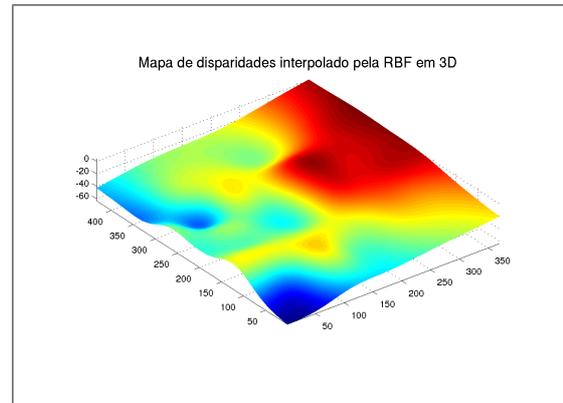


Figura 5.24: Mapa interpolado de disparidade usando espaçamento de 50 pixels, gerado a partir do par estéreo "Lampshade1".

## 5.4 Relação Erro-Tempo × Espaçamento da malha

Para observarmos como o espaçamento da malha contribui para o aumento do erro, vejamos as curvas plotadas nas Figuras 5.25, 5.26, 5.27 e 5.28. Apesar de um certo ruído no processo, o erro sempre aumenta com o aumento do espaçamento, no entanto, observamos, também, que o tempo de execução do algoritmo cai. Isto nos mostra que a inclusão do fator de espaçamento no método contribui significativamente para o aumento da velocidade de execução.

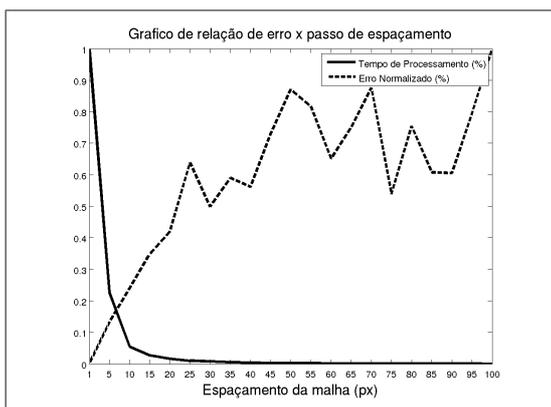


Figura 5.25: Gráfico do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Aloe".

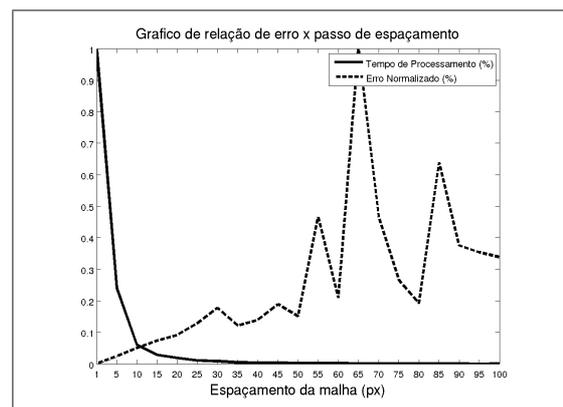


Figura 5.26: Gráfico do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Midd1".

Como podemos observar nas Tabelas 5.1, 5.2, 5.3 e 5.4, existe um ganho de velocidade no tempo de execução do algoritmo. Observamos, ainda, que a taxa de erro, embora cresça com a interpolação, ainda é bastante pequena.

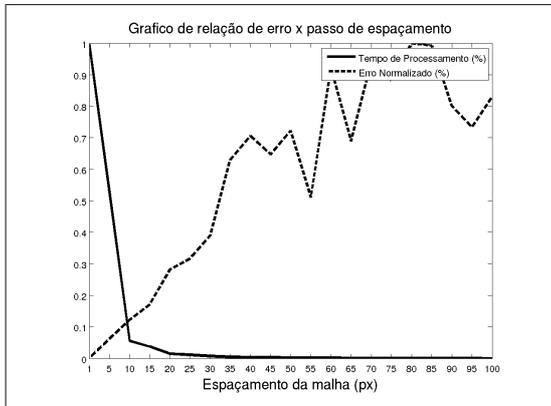


Figura 5.27: Gráfico do Erro-Tempo  $\times$  Espaçamento da malha, gerado a partir do par estéreo "Bowling1".

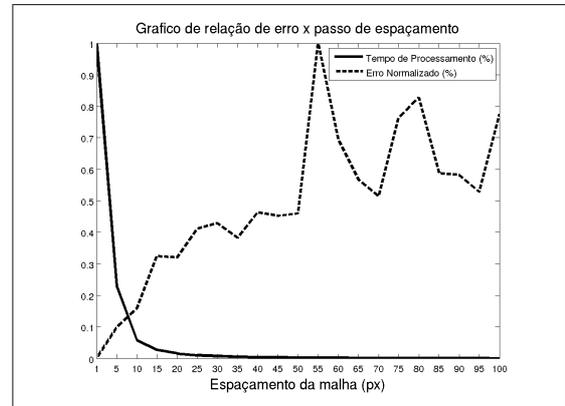


Figura 5.28: Gráfico do Erro-Tempo  $\times$  Espaçamento da malha, gerado a partir do par estéreo "Lampshade1".

Espaçamento	Tempo Relativo	Erro Relativo
1 px	100 %	0 %
5 px	18,27 %	7,95 %
10 px	4,34 %	16,07 %
15 px	2,11 %	26,52 %
50 px	0,19 %	58,23 %

Tabela 5.1: Tabela dos valores do Erro-Tempo  $\times$  Espaçamento da malha, gerado a partir do par estéreo "Aloe".

Espaçamento	Tempo Relativo	Erro Relativo
1 px	100 %	0 %
5 px	21,80 %	2,12 %
10 px	6,09 %	4,59 %
15 px	3,61 %	6,96 %
50 px	1,36 %	26,16 %

Tabela 5.2: Tabela dos valores do Erro-Tempo  $\times$  Espaçamento da malha, gerado a partir do par estéreo "Midd1".

Espaçamento	Tempo Relativo	Erro Relativo
1 px	100 %	0 %
5 px	18,44 %	5,29 %
10 px	5,94 %	10,39 %
15 px	3,13 %	14,56 %
50 px	0,16 %	61,46 %

Tabela 5.3: Tabela dos valores do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Bowling1".

Espaçamento	Tempo Relativo	Erro Relativo
1 px	100 %	0 %
5 px	24,34 %	8,13 %
10 px	7,11 %	13,73 %
15 px	5,53 %	19,39 %
50 px	0,34 %	45,24 %

Tabela 5.4: Tabela dos valores do Erro-Tempo × Espaçamento da malha, gerado a partir do par estéreo "Lampshade1".

Outro comparativo importante é verificar o desempenho da rede RBF em relação a um interpolador linear, no tocante ao tempo, nas Figuras 5.29, 5.30, 5.31 e 5.32, e quanto ao erro nas Figuras 5.33, 5.34, 5.35 e 5.36.

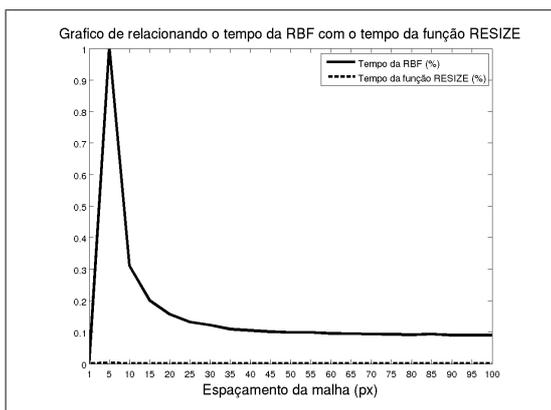


Figura 5.29: Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Aloe".

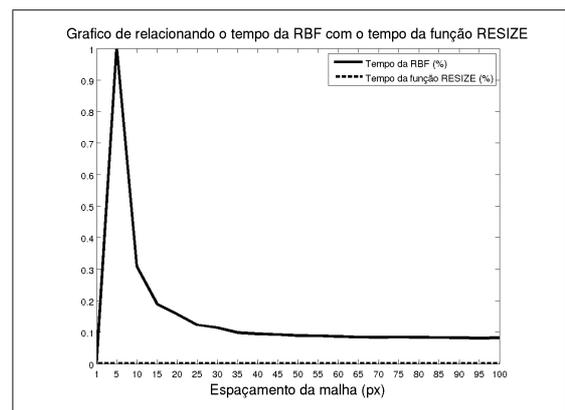


Figura 5.30: Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Midd1".

É interessante notar que, apesar da rede RBF ter um tempo de execução um pouco su-

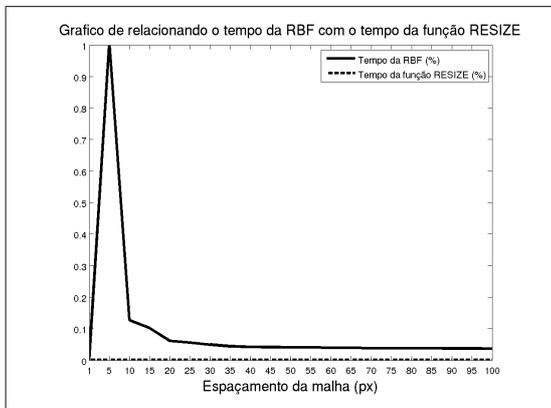


Figura 5.31: Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Bowling1".

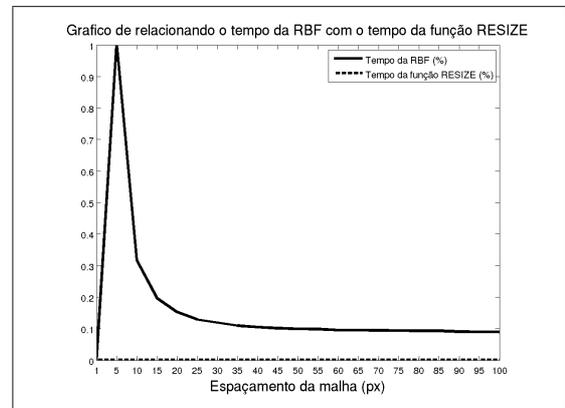


Figura 5.32: Gráfico comparativo entre tempo de execução da RBF e o tempo do interpolador linear, gerado a partir do par estéreo "Lampshade1".

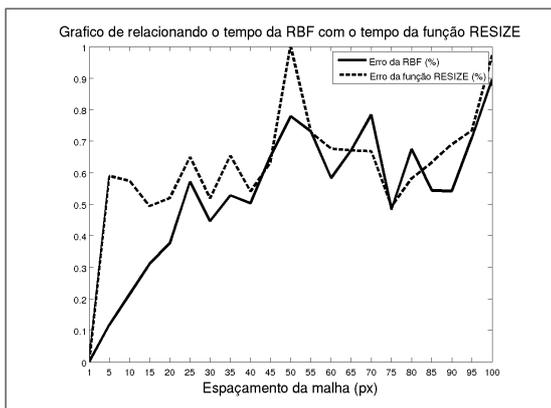


Figura 5.33: Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Aloe".

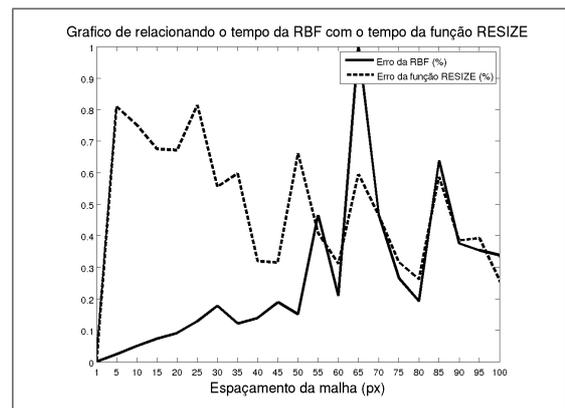


Figura 5.34: Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Midd1".

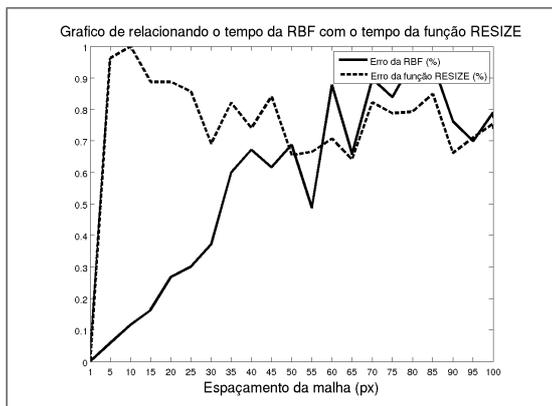


Figura 5.35: Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Bowling1".

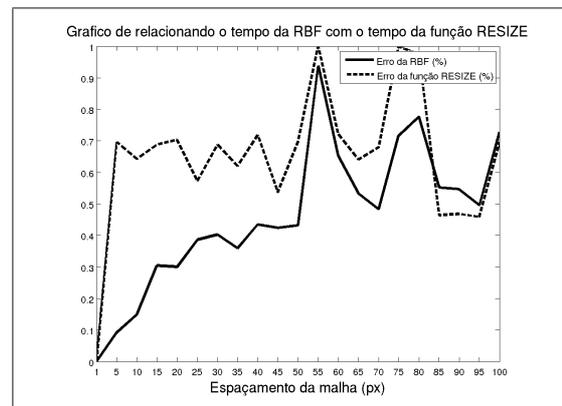


Figura 5.36: Gráfico comparativo entre erro gerado pela interpolação da RBF e o erro do interpolador linear, gerado a partir do par estéreo "Lampshade1".

perior ao interpolador linear, o resultado apresentado pela RBF é bem superior, sobretudo quando a malha de espaçamento ainda está pequena.

## 5.5 Comentários finais

Diante dos resultados obtidos, verificamos que conseguiu-se uma melhoria significativa no desempenho do algoritmo clássico de correlação de imagens estéreo. Essencialmente percebeu-se, através dos experimentos, que o fator de espaçamento reduziu consideravelmente a quantidade de iterações do algoritmo de casamento, sem comprometer, no entanto, o resultado final do mapa de disparidades.

No próximo capítulo faremos os comentários finais sobre esse trabalho, desafios encontrados e proposições de trabalhos futuros.

---

# Capítulo 6

## Conclusões

---

Concluimos que o algoritmo proposto, neste trabalho, conseguiu produzir os mapas de disparidade em um tempo menor e com um erro pequeno, isso comprovado através dos resultados apresentados no capítulo anterior.

Em tempo, gostaríamos de tecer alguns comentários sobre os principais desafios enfrentados e fatos peculiares que ocorreram no decorrer da pesquisa. Sabemos que todo trabalho de pesquisa sempre passa por situações inadvertidas, mas, também sabemos que todas elas servem para enriquecer ainda mais o estudo e nos ajudar a contornar problemas futuros. Na nossa opinião, o valor de um trabalho como o presente vai além de sua publicação e sua contribuição científica, não se podendo aqui mensurar a importância das experiências vividas durante a sua execução. Tomamos a liberdade, neste capítulo, de fugir do protocolo e comentar livremente sobre algumas questões-chave na elaboração desta dissertação.

A primeira descoberta inesperada no decorrer do desenvolvimento deste trabalho foi perceber que a área da visão computacional oferece muitos desafios interessantes para a exploração, a experimentação e a pesquisa. Por se tratar de uma área relativamente jovem, são vastas as possibilidades de contribuição. Um aspecto que nos chamou a atenção diz respeito ao fato de que os modelos relacionados com a visão computacional geralmente estão ligados a conceitos biológicos, e a natureza se torna a principal fonte de inspiração para o desenvolvimento dos algoritmos presentes na área. Isto nos dá um sentido totalmente novo da criação da natureza, e nos revela como tudo foi cuidadosamente calculado antes de ser criado.

Outro aspecto interessante com o qual nos deparamos se refere ao fato de que, para simular processamentos que normalmente apenas o cérebro é capaz de calcular com eficiência, o computador se mostra uma ferramenta extremamente limitada. Para os procedimentos utilizados, verificou-se que a memória facilmente se esgotava ou demorava horas para terminar o processamento. Isso nos mostrou o quanto ainda estamos atrasados no

desenvolvimento da tecnologia dos processadores e memórias. Mesmo com toda vantagem de se utilizar um software matemático de simulação, o Matlab<sup>TM</sup>, nosso constante desafio era fazer pequenos ajustes nos algoritmos para evitar que houvesse "estouro" da memória, ou então, fazer com que o processamento ocorresse mais rapidamente. Algumas simulações deste trabalho levaram dias para serem executadas completamente.

Resolvidos os problemas de memória e processamento citados anteriormente, enfrentamos outro, que não foi menor do que o primeiro, principalmente, devido a simplicidade do algoritmo de casamento estéreo adotado. Esse problema consistiu no ajuste dos parâmetros do algoritmo a fim de que se produzisse um bom resultado de disparidade. Estes parâmetros de ajuste, se mal sintonizados, produziam mapas de disparidade muito distantes dos reais e pouco coerentes. Para resolver isso, recorremos a valores empíricos e um conjunto sucessivo de simulações até chegar a valores de janelamento e de espaço de busca satisfatórios para o cálculo das disparidades. No entanto, não nos preocupamos em implementar uma forma automática de aferição desses parâmetros (ver Perspectivas).

## 6.1 Perspectivas

Identificamos vários pontos durante a nossa pesquisa que tiveram que ser postos de lado para não atrasar a conclusão desta dissertação. Deixamos estes pontos para trabalhos futuros e revisões posteriores, contudo, comentaremos alguns que consideramos os mais importantes.

A primeira perspectiva deste trabalho é, justamente, a de avançar as pesquisas para conseguir produzir os mapas de disparidade em tempo real, tornando assim viável a implementação deste algoritmo em cenários reais.

Outra igualmente importante, considerando que os parâmetros do algoritmo de correlação baseado em área são arbitrariamente estipulados, seria de poder estimar esses parâmetros de uma forma automática, descartando a necessidade de simulações, e habilitando o sistema para processar disparidades de imagens reais com maior eficiência.

Por último, sabendo-se que as redes neurais de função de base radial possuem, inerentemente, a característica de formar uma malha tridimensional com os neurônios treinados, esta malha, que também pode ser entendida como a superfície de interpolação, poderia ser a entrada de um sistema de reconstrução tridimensional, o que nos daria uma visão mais aproximada da cena real e não somente um mapa de disparidades.

Sem dúvidas, outros interpoladores, que não o neural baseado em RBF, poderiam ser testados, e seus desempenhos comparados com o objetivo de concluir sobre qual o melhor interpolador para este cenário. Por outro lado, outros aspectos aqui não identificados tam-

bém poderão suceder este estudo, bastando para isso que uma mente de um pesquisador os explorem e os concretizem, como fizemos com este trabalho.

---

# Referências Bibliográficas

---

- Arbib, M. A. (1981), *Perceptual structures and distributed motor control*, in *Handbook of Physiology - The Nervous System*, V. B. Brooks, Bethesda, MD.
- Baumgart, B. G. (1974), *Geometric modeling for computer vision*, Stanford University, Department of Computer Science.
- Binford, T. O. (1984), *Stereo vision: complexity and constraints*, Symp. on Robotics Research, MIT Press.
- Fernandes, R. G., R. W. Silveira e A. D. Dória Neto (2004), *On Disparity Matching in Stereo Vision via a Neural Networks Framework*, Departamento de Engenharia de Computação e Automação - UFRN.
- Gonzalez, R. C. e R. E. Woods (1992), *Processamento de Imagens Digitais*, Ed. Edgard Blücher.
- Goulermas, J. Y. e P. Liatsis (1999), *Feature Based Stereo Matching via Coevolution of Epipolar Subproblems*, vol. 1ª edição, Seventh International Conference on Image Processing And Its Applications.
- Grimson, W. E. L. (1981), *A computer implementation of a theory of human stereo vision*, 217-253, 1981ª edição, Philosophical Transactions of the Royal Society of London.
- Haykin, S. (2001), *Neural Networks: A comprehensive foundation*, Pearson Education.
- Hirschmüller, H. e D. Scharstein (2007), *Evaluation of cost functions for stereo matching*, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007).
- Julesz, B. (1960), *Binocular depth perception of computer-generated patterns*, 39(5):1125-1162ª edição, The BellSystem Technical Journal.
- Li, G. e Y. He (2002), *A hierarchical combined feature and area-based stereo matching algorithm*, 2ª edição, Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium.

- Marapane, S. B. e M. M. Trivedi (1989), *Region-based stereo analysis for robotic applications*, Systems, Man and Cybernetics, IEEE Transactions.
- Marr, D. (1982), *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, Henry Holt and Co., Inc., New York, NY, USA.
- Marr, D. e T. Poggio (1979), *A computational theory of human stereo vision*, Proceedings of the Royal Society of London, UK.
- Petrou, M. e P. Bosdogianni (1999), *Image Processing: The Fundamentals*, John Wiley & Sons, UK.
- Rahmatulloh, F. (2009), *Radial Basis Function Networks for Modeling Option*, <http://statistikawan.org/radial-basis-function-networks-for-modeling-option.html>.
- Scharstein, D. e C. Pal. (2007), *Learning conditional random fields for stereo*, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007).
- Shapiro, L. G. e G. C. Stockman (2001), *Computer Vision*, Prentice Hall, Upper Sadle River, NJ.
- Sunyoto, H., W. V. D. Mark e D. M. Gavrila (2004), *A comparative study of fast dense stereo vision algorithms*, Intelligent Vehicles Symposium, 2004 IEEE.
- Wang, J. H. e C. P. Hsiao (1999), *On Disparity Matching in Stereo Vision via a Neural Networks Framework*, Natl. Sci. Counc. ROC.