UM SISTEMA DE CONTROLE ROBÓTICO PARA INTEGRAÇÃO DE INFORMAÇÃO SENSORIAL MULTIMODO

Luiz Marcos Garcia Gonçalves

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:

Prof. Antonio Alberto Fernandes de Oliveira, D.Sc.

Prof. Cláudio Esperança, Ph.D.

Prof. Felipe Maia Galvão França, Ph.D.

Prof. Dibio Leandro Borges, Ph.D.

Prof. José Ricardo de Almeida Torreão, Ph.D.

RIO DE JANEIRO, RJ - BRASIL
NOVEMBRO DE 1999

GONÇALVES, LUIZ MARCOS GARCIA

Um Sistema de Controle Robótico para Integração de Informação Sensorial Multimodo [Rio de Janeiro] 1999

XII, 104 p., 29,7 cm, (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 1999)

Tese - Universidade Federal do Rio de Janeiro, COPPE

1 - Robótica, Controladores e Aprendizado Automático

2 - Integração de Informação Sensorial Multimodo

3 - Visão Robótica, Atenção e Categorização de Padrões

I. COPPE/UFRJ II. Título (série)

Dedicatória:

À Luciane, Daniel, Gabriel e Marcelly.

Agradecimentos:

Agradeço às seguintes pessoas e instituições que colaboraram direta ou indiretamento para a elaboração desta tese:

À Minha mãe, Marta Gonçalves, que deu o maior apoio, no inverno de 1998, quando do nascimento de meu filho Gabriel Garcia em North Hampton, MA, USA.

Ao Prof. Antonio Oliveira que me convidou para ingressar diretamente ao Doutorado, dando o maior incentivo também e pelo apoio à parte técnica e intelectual.

Ao Prof. Roderic Alan Grupen da UMASS que foi um dos principais responsáveis pela escolha do tema de tese, disponibilizando a cabeça estéreo do LPR para que eu pudesse trabalhar.

Aos Brasileiros da pequena Amherst e de outros lugares dos EUA, que em muito nos apoiaram e nos ajudaram a ter uma estadia super agradável: Ricardo, Vânia, Letícia e Renato (os Farias), Jefferson e Elizeth (os "Rabbits"), Mauricio, Hilcea, Bruna e Rafael (os Marengoni), Pilar, "Baiano", Bobi e Daniel (os Torres), Rich, Carla e Luana (os Goulet), Sérgião (da UFSC), Lucio (Tinoco), Mauricio e Cláudia (os Pinto), Tito, Janice, Vitória e Júlia (os Bonagamba), Jeanne Marie e Sérgio (o Mexicano), Maria Lúcia, Eduardo e Vanusa (os Souza), Kleber e Lilian, Magnólia (Santos), Fátima, Willian e Brian (os Meeks), Alexandra e Pedro (os Moreira), Ana Maria e Maria Alice, Murilo (da UFSC) e família, e outros.

Aos demais colegas do LPR, do departamento de CS, da UMASS e de Amherst: Manfred Huber, Justus Piater, David Wheeler, Patrick Deegan, Cosimo Distante, M.S. Raunak (que comprou minha "Chevi" por US$ 750), Kamal Souccar, e aos alunos das turmas de TA, pela companhia.

Aos professores, colegas e ex-colegas do LCG e COPPE: Cláudio Esperança, Ronaldo Marinho, José Pio, Marcelo Kallmann, Orlando (in memorian), Vitor Toso, Lúcio Fialho, Deise Ribeiro, Sara Nascimento, Sonja Meerbaum, José Brito, José Maria, Fernando Wagner, Gilson Giraldi, Antonio Lopes, Paulo Sérgio, Italo Matias, Mara Rios, Nelma Ribeiro, Carla Cristina, Carla Godinho, Moisés, Sérgio, Walter e outros.

Aos árbitros de todas as conferências e simpósios que aceitaram nossos "papers" para publicação, dando suporte à parte técnica desta tese.

Aos cartões de telefone "Alô Brasil", sem os quais nao poderíamos suportar a saudade do Brasil, quando estávamos nos EUA (a 14 centavos por minuto, dava para falar bastante com o Brasil).

Ao CNPQ, CAPES, NSF e UMASS (EUA), e FAPERJ que contribuiram com a parte do apoio financeiro.

Finalmente, agradeço À Deus por tudo.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

UM SISTEMA DE CONTROLE ROBÓTICO PARA INTEGRAÇÃO DE INFORMAÇÃO SENSORIAL MULTIMODO

Luiz Marcos Garcia Gonçalves

Novembro/1999

Orientador: Antonio Alberto Fernandes Oliveira

Programa: Engenharia de Sistemas e Computação

Este trabalho descreve a implementação de um sistema robótico para o controle da atenção e categorização de objetos baseado em informação sensorial multimodo. O sistema foi implementado em duas plataformas distintas: um robô simulado "Roger-the-Crab" que consiste em dois olhos e dois braços integrados numa única plataforma e um robô real que consiste em uma cabeça estéreo articulada com quatro graus de liberdade. Como resultado prático, o agente consegue selecionar uma região de interesse, executar mudanças do foco de atenção envolvendo movimentos sacádicos, executar uma extração eficiente de características e reconhecimento, construir incrementalmente um mapa do ambiente e manter este mapa consistente com uma percepção corrente do local. O sistema é capaz de analisar todas as regiões em seu ambiente, selecionadas de acordo com um mapa de saliências.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

A ROBOTIC CONTROL SYSTEM FOR INTEGRATION OF MULTI-MODAL SENSORY INFORMATION

Luiz Marcos Garcia Gonçalves

November/1999

Advisor: Antonio Alberto Fernandes Oliveira

Department: Computing and Systems Engineering

This work describes the implementation of a robotic system for control of attention and for pattern categorization based on multi-modal sensory information. The system was implemented in two distinct platforms: a simulated robot "Roger-the-Crab" consisting of two eyes and two arms integrated in a unique platform and a real robot consisting of an articulated stereo-head with four degrees of freedom (pan, tilt, left verge, and right verge). As a practical result of this work, the system can select a region of interest, perform attentional shifts involving saccadic movements, perform efficient feature extraction and recognition, incrementally construct a world map, and keep the map consistent with a current perception of the world. The system is capable of analyzing all regions of its world, selected according to a salience map.

Sumário

Lista de Figuras

Capítulo 1

Introdução

Esta tese propõe um modelo básico de arquitetura para um sistema robótico de cognição, o qual foi inspirado em resultados de neuro-psicologia e neuro-biologia. Para a implementação de tal sistema, são usadas ferramentas de inteligência artificial, processamento digital de imagens e teoria de controle em robótica. Assumimos que um sistema robótico para cognição envolve basicamente a construção de mecanismos que permitam ao robô mudar o seu foco de atenção e também categorizar ou identificar elementos a partir de informação sensorial. Estes mecanismos serão usados por um agente robótico na execução de tarefas em tempo-real. A arquitetura básica, que será proposta aqui, foi inicialmente desenvolvida usando um simulador robótico composto por dois olhos e dois braços (GONÇALVES *et al.* , 1998a; GONÇALVES & OLIVEIRA, 1999; GONÇALVES *et al.* , 1999b). Numa segunda fase, a mesma arquitetura ligeiramente adaptada foi implementada para controlar um robô real que consiste de uma cabeça estéreo com quatro graus de liberdade (movimentos de pan, de tilt, e de vergências esquerdo e direito) (GONÇALVES *et al.* , 1998b; GONÇALVES *et al.* , 1999a; GONÇALVES *et al.* , 1999c). Apesar de nas implementações efetuadas no hardware real terem sido realizados experimentos usando apenas informação visual (provida por câmeras), a arquitetura proposta é capaz de integrar informação multi-modo, obtida por intermédio de vários sensores de diferentes tipos, num sistema de comportamento ativo e cooperativo.

Este trabalho não tem intenção de sugerir ou descrever modelos matemáticos ou computacionais para sistemas biológicos, nem de explicar comportamentos ou a funcionalidade desses sistemas. A proposta se resume basicamente em dar um comportamento cognitivo, usando visão ativa e informações hápticas, a uma plataforma robótica. Entretanto, como diversas partes da arquitetura computacional construída são inspiradas em sistemas biológicos, terminologia relativa a partes e características de um sistema biológico são amplamente usadas no texto, referindo-se na realidade a partes e características equivalentes em um robô.

Apesar da metodologia aqui desenvolvida poder ser aplicada mais amplamente, nos ateremos apenas a duas modalidades de informação sensorial: háptica (incluin-

do propriocepção e tato) e visual. Uma maneira natural de associar os sistemas sensoriais háptico e visual é assumir que o primeiro é subordinado ao segundo, com os braços e mãos agindo baseados na informação visual disponível. A visão é sem dúvida o mais importante e poderoso sistema sensorial em organismos biológicos, e também o mais complexo. Mas, há situações em que o sistema háptico parece ser fundamental. Por exemplo, dadas duas peças, uma feita de madeira e outra de ferro e ambas com mesma pintura o que as torna exatamente iguais visualmente, se a tarefa é distinguir qual é a peça feita de ferro, a determinação do peso extraída pelo braço (informação háptica) definirá uma solução para o problema. Em situações como esta, a informação háptica serve para desambiguar a informação visual. Neste trabalho, a importância relativa dos sistemas visual e háptico (e também de outros sistemas sensoriais) é assumida ser altamente dependente do contexto. Os sistemas sensoriais trabalham em paralelo, provendo informação complementar ou redundante a um sistema tomador de decisões, o qual é responsável por prover respostas adaptativas (ações) aos estímulos ambientais. Estamos interessados num sistema que seja capaz de fazer vergir os olhos numa região de interesse, de subsequentemente mover os braços para alcançar e tocar um possível objeto existente naquela região e de escolher uma outra região, quando a corrente não for mais de interesse, mudando seu foco de atenção. Para validar tal sistema, uma tarefa envolvendo todos os procedimentos acima deve ser definida. Possíveis tarefas são o reconhecimento e identificação de objetos para fins de inspeção ou monitoração, orientação espacial do robô e, eventualmente, navegação (planejamento dos movimentos a serem realizados ou do caminho a ser seguido pelo robô). Um mapa do ambiente é construído de forma incremental e pode ser mudado dinamicamente. O robô tem ainda que lidar com instâncias novas de objetos e não apenas com as já conhecidas. Além da representação das características dos objetos ("object-indexes"), este mapa contém também informação sobre o seu posicionamento e orientação. Uma vez que este tipo de mapa esteja construído, o agente robótico pode executar tarefas específicas. Ao adotarmos uma estratégia comportamental ativa, estamos provendo uma maneira dinâmica para que o robô possa interagir com ambientes variáveis.

Na implementação final que valida a proposta deste trabalho, informação visual adquirida em tempo real é usada pela cabeça estéreo para prover uma resposta "on-line" para um estímulo ambiental. Como resultado temos um sistema com comportamento ativo que age de acordo com o seu estado perceptual. As tarefas essenciais para esse sistema de visão ativa são o controle da atenção e categorização (identificação e/ou reconhecimento) de objetos.

Basicamente, para controlar a atenção, nós usamos um mapa de saliências "bottom-up" (dirigido por estímulo ambiental). O direcionamento da atenção é feito selecionando-se uma região de interesse (ROI) a partir do mapa da saliências, computando e

executando movimentos sacádicos para os olhos e eventualmente movimentos de pescoço/pan e/ou tilt, para colocar a região de interesse na fóvea (parte central da retina). Então, um processo para a extração de características ("features") é executado, provendo mudanças no estado perceptual do robô. Uma memória associativa mapeia estas características no endereço de um padrão armazenado numa memória de longa duração ("long-term memory" ou MLT), permitindo ao sistema reconhecer/identificar uma possivel instância de um objeto ou descobrir novas categorias deles (objetos desconhecidos). Finalmente, a construção e manutenção de um mapa do ambiente que contenha todas as representações, de maneira eficiente, completa a arquitetura de tal sistema.

Duas questões principais surgem, quando se usa um modelo baseado em extração de características para identificação de objetos:

- Determinar um conjunto mínimo de características que seja suficiente para gerar uma descrição plausível do ambiente. Isto inclui caracterizar as propriedades naturais de todos os objetos de forma que se possa realizar uma segregação destes (criar categorias).

- Determinar uma estratégia global que permita definir em tempo real qual o sub-conjunto de características que deva ser usado, de acordo com a tarefa que esteja sendo realizada num dado instante.

Nós procuramos resolver estas questões empregando memórias associativa e de longa duração dinâmicas. A cada estágio do desenvolvimento elas devem possuir informação suficiente para lidar com o conjunto de objetos conhecidos correntemente. Para conseguir isso, o conhecimento sobre as características dos objetos é adquirido e aumentado automaticamente por um supervisor de aprendizado. Esse conhecimento deve ser pelo menos o suficiente para a execução de uma determinada tarefa (no caso, inspeção ou monitoração).

Outra questão relaciona-se a usar ou não processos de segmentação para separar o que é objeto do que é fundo. Para responder esta questão, é preciso caracterizar devidamente em cada contexto de aplicação o que é objeto. Note que a noção intuitiva de objeto sugere alguma coisa no ambiente que desperte interesse. Por outro lado, podemos afirmar categoricamente que o que seja objeto sob execução de uma tarefa pode não o ser quando da execução de alguma outra tarefa. Há ainda o problema de juntar partes disjuntas de objetos. Estas podem ser determinadas por motivos de oclusões. Estes problemas levaram-nos a abandonar, na implementação em ambiente real, o modelo de segmentação utilizado em ambiente de simulação. Na implementação feita para o ambiente real, uma região é uma única célula de uma grade regular relativa a uma dada escala, enquanto que na implementação para o

ambiente de simulação ela é considerada como uma estrutura que pode ocupar várias células, sendo seu tamanho definido em função de um processo de segmentação.

Num método "top-down" ideal, a segmentação é realizada pelo processo de atenção, usando filtros detectores de bordas para definir as regiões e/ou filtragem passa-banda para fazer o sistema sintonizar na faixa (de frequência) que realmente é de interesse. A determinação do valor desta faixa do espectro que seja de interesse pode ser feita por vários subprocessos, de acordo com a tarefa sendo executada. Já num processo "bottom-up", fica difícil definir um modelo de segmentação baseado em filtros passa-banda, uma vez que não se sabe a *priori* o tipo de informação que possa interessar. Conjecturamos que talvez usando aprendizado por reforço (Q-learning) onde o sistema recebe recompensas por ações executadas para cada faixa de frequência, de acordo com a tarefa, seja possível determinar faixas de maior interesse.

Neste trabalho, procuramos seguir um modelo visando buscar um equilíbrio entre divisão do trabalho e eficiência computacional. Em linhas gerais, nós propomos um sistema computacional de aprendizado híbrido, no qual usamos uma memória associativa que relembra as assinaturas visual e háptica dos objetos para reconhecimento e também aprendizado Q-learning para derivar políticas de controle ativas para os sistemas sensoriais. A memória associativa aprende usando características semi-invariantes (momentos normalizados) extraídas pelos processos visual e háptico. O processo de aprendizado Q-learning opera em alto-nível, orquestrando os controladores do sistema.

1.1 Organização do Trabalho

Além desta breve introdução, o restante desta dissertação está organizado da maneira apresentada a seguir.

- **Capítulo 2 - Cognição Robótica e Trabalhos Relacionados**

Neste capítulo, descrevemos o estado da arte em cognição robótica, bem como trabalhos de outras áreas relacionados com o tema desta dissertação tais como aprendizado de reforço (Q-learning), e visão estéreo. É também apresentado um apanhado dos métodos mais recentes em robótica e alguns desenvolvimentos das áreas de neuro-biologia e neuro-fisiologia que inspiraram nossa proposta.

- **Capítulo 3 - Arquitetura de Controle**

Este capítulo descreve a arquitetura básica de controle para um sistema sensorial multi-modo robótico genérico. São descritos os principais componentes de um sistema de controle com um comportamento ativo que foram usados tanto em simulação quanto na plataforma da cabeça estéreo.

- **Capítulo 4 - Ambientes de Implementação**

Aqui descrevemos os dois ambientes nos quais a arquitetura de controle desenvolvida foi implementada. A primeira seção descreve particularidades do ambiente de simulação "Roger-the-Crab" que foi desenvolvido por nós, bem como a implementação da arquitetura nele realizada. A segunda seção descreve particularidades da implementação realizada na plataforma de hardware (cabeça estéreo).

- **Capítulo 5 - Experimentos e Resultados**

Neste capítulo descrevemos os experimentos realizados tanto em simulação quanto na plataforma de hardware para testar a arquitetura proposta, mostrando também os resultados alcançados. Basicamente, são testadas tarefas de inspeção ou monitoração, onde o agente robótico deve construir um mapa de seu ambiente. Objetos são postados numa região restrita e o robô tem que percorrer todas as posições testanto o mecanismo de atenção e de identificação. O desempenho do sistema é discutido e apresentado em termos de tempo computacional gasto em cada uma das fases necessárias aos processos de atenção e identificação, e também em termos da eficiência destes processos.

- **Capítulo 6 - Conclusões, Discussões, e Trabalhos Futuros**

Neste último capítulo, são apresentadas algumas conclusões obtidas a partir dos resultados dos experimentos e são formalizados alguns melhoramentos possíveis de serem realizados na atual implementação.

- **Apêndice A - Redes Neurais do Tipo Back-propagation.**

Neste apêndice, é apresentado e discutido o algoritmo de aprendizado usando redes neurais do tipo back-propagation. Este algoritmo foi usado nas duas implementações realizadas neste trabalho.

- **Apêndice B - Sistema Visual Biológico.**

Particularidades do sistema visual biológico e sua funcionalidade são apresentadas, sendo feita uma descrição das principais estruturas e dos caminhos visuais que levam a informação ao cérebro (córtex visual).

Capítulo 2

Cognição Robótica e Trabalhos Relacionados

Neste trabalho usamos ferramentas de visão computacional, especialmente visão ativa, e de inteligência artificial para prover um comportamento ativo a um robô. O objetivo maior é prover políticas básicas de controle que possibilitem a um robô, a partir da informação provida por um servidor visual e háptico, tomar decisões acertadas de maneira automática frente a situações que ficam perfeitamente definidas pelo estado perceptual do robô. Este último, a um nível de aplicação mais geral, pode ser definido como o conjunto de informação provida por sensores robóticos (tais como câmeras, sonares, detectores de luz infra-vermelha, detectores de som, detectores de calor) e pela pose do robô, a qual inclui informação de natureza proprioceptiva e funcional. Um exemplo típico de decisão é a que deve ser tomada por um controlador em relação a mover um braço na direção de um objeto.

Neste capítulo, apresentamos informação de base e discutimos os principais métodos e modelos de visão computacional, robótica e aprendizado automático que foram essenciais para o desenvolvimento e a compreensão do sistema que será apresentado nos próximos capítulos. Este capítulo está organizado em três seções principais onde os seguintes tópicos são apresentados e discutidos: atenção e categorização, visão estéreo e disparidade, robótica e aprendizado automático. Ao final do capítulo vem uma seção de discussões e generalidades.

2.1 Atenção e Categorização: As Bases da Cognição

Ultimamente, muitos pesquisadores têm tentado imitar ou reproduzir comportamentos e sistemas biológicos para tentar modelar cognição em robótica. Neste trabalho, consideramos os processos (ou habilidades) de atenção, envolvendo a criação e manutenção de um mapa (atencional) do ambiente, e de categorização de padrões

como sendo as bases para a cognição robótica. Nesta seção, além de apresentar definições formais e descrever as principais formas de implementação dos processos citados acima, discutiremos também algumas metodologias mais gerais que possibilitam obter uma integração desses processos.

2.1.1 Atenção

Formalmente, sob um ponto de vista mais geral, o processo de atenção pode ser definido como a habilidade de selecionar um tópico de interesse, ou um objetivo, e fazer com que os processos de cognição "sintonizem" neste objetivo, para extrair informação útil na execução de uma dada tarefa. Podemos ainda adicionar a esta definição a habilidade de mudar o interesse de um tópico para outro, quando for necessário. Note que, segundo esta definição, o tópico de interesse pode ser abstrato (virtual) ou real. Um exemplo de tópico de interesse abstrato se dá quando um indivíduo é flagrado olhando para o nada, pensando num problema ou formando uma imagem mental de uma situação (podemos dizer que neste caso a atenção está sendo guiada internamente). Neste trabalho lidamos exatamente com a situação oposta. Os nossos tópicos de interesse para o processo atencional relacionam-se a estímulos ambientais. Mais especificamente, são objetos existentes num ambiente real restrito no qual o robô se encontra. A atenção será empregada visando exatamente a interação do robô com o seu ambiente de trabalho.

Dentro deste contexto mais específico, um tópico de interesse passa a ser uma região de interesse. Selecionar uma região de interesse fica relativamente simples se construirmos um ou mais mapas de saliência contendo, para cada região, um conjunto de valores de ativação referentes a ela se tornar o foco da atenção segundo vários critérios ou características diferentes, dependentes da tarefa. A região cujo o somatório dos valores for maior é escolhida como sendo o alvo atencional. Estes valores são geralmente calculados usando-se filtros lineares aplicados diretamente aos dados sensoriais ou a mapas de características derivados daqueles. Uma vez que se tenha uma região objetivo definida, torna-se relativamente simples manter aquela região como alvo do processo de cognição até que a mesma não seja mais necessária. O problema principal é exatamente o de, dada uma tarefa específica, determinar os pesos que serão atribuídos às funções de transferência aplicadas a resposta de cada filtro. Assim, a complexidade do processo de atenção relaciona-se mais a mudança do foco de atenção propriamente dita do que a fixação em uma determinada região. Considerando a atenção sob esse modelo dinâmico, ela pode ser classificada segundo diferentes aspectos, apresentados a seguir.

- Quanto ao sentido, geralmente considera-se que a atenção pode ocorrer de duas formas diferentes: "bottom-up" e "top-down". Em organismos biológicos, a atenção "bottom-up" é explicada pelo fato dela ser originada por um estímulo

que provoca a atividade neural em níveis sucessivamente mais altos, até atingir o nível de decisão da mudança de atenção. Já na atenção top-down, o indivíduo é guiado internamente em direção a um estímulo particular, dependendo da tarefa ou da interpretação da tarefa a ser executada.

- Quanto a localização no mapa sensorial, a atenção pode ser encoberta ("covert") ou descoberta ("overt"). Na atenção descoberta, o objetivo é efetivamente colocado no centro do mapa sensorial, para que alguma informação seja extraída, sendo geralmente realizados movimentos físicos para isto. Na atenção encoberta (também chamada de falsa atenção), não são realizados movimentos físicos e o alvo atencional pode se situar em qualquer lugar no mapa sensorial em questão. Note que em se tratando de um sistema sensorial visual, a ocorrência do primeiro tipo de atenção (descoberta) seria uma situação ideal para o caso de processos cognitivos, pois naquela posição central a resolução é maior, a exemplo do que acontece na fóvea. Por outro lado, se a tarefa não exige uma grande resolução, a situação de atenção encoberta é preferível.

Neste trabalho consideramos um conjunto pequeno de características atencionais, suficiente para o processo atencional determinar onde colocar a janela de atenção. Isto ajuda a minimizar a quantidade de cálculos, minimizando também o tempo total necessário para mover a janela de atenção de uma posição a outra. Em organismos biológicos, o tempo requerido para atenção encoberta é de aproximadamente 30 a 50 milisegundos (JULESZ & SAARINEN, 1991). Já o tempo requerido para mudança da atenção que envolva um movimento sacádico dos olhos (atenção descoberta) varia de 80 milisegundos para sacádicos expressos a 200 milisegundos para sacádicos lentos (FISCHER *et al.* , 1993).

2.1.2 Categorização: Identificação e Reconhecimento

Identificação e reconhecimento de padrões têm sido estudados desde há muito tempo. Convém fazer um distinção entre os dois conceitos. No contexto deste trabalho, identificação significa determinar as características específicas de um determinado tipo de objeto que possam futuramente indicar que uma instância daquele tipo estará sendo observada a uma determinada fixação. Por exemplo, quando alguém cita o nome próprio de um sujeito, é porque ele já foi anteriormente identificado (ex: este é Bobi, o cachorro da Ana). Reconhecimento, em um contexto mais amplo, significa relacionar a informação sensorial com alguma classe de objeto, independentemente de uma instância específica. Assim, se alguem vê (ou toca) um determinado objeto, será possível dizer futuramente se já esteve em contato ou se já viu anteriormente algo parecido com esse objeto. Aqui, usamos o termo categorização para substituir os dois termos anteriores. Assim, podemos ter uma categoria específica

que determine uma única instância de objeto ou podemos ter uma categoria mais geral que determine uma classe de objetos.

Alguns aspectos devem ser levados em consideração quando se fala em categorização. Nós categorizamos objetos principalmente a partir de características relativas a forma dos mesmos. Também categorizamos objetos a partir de características baseadas em padrões de cor e textura, padrões de movimento, relações de adjacência entre as partes e outras características não triviais como funcionalidade e utilidade (uma porta pode ser reconhecida porque pessoas a usam para entrar e sair de uma sala). A seguir, descrevemos algumas situações em que o sistema visual biológico demonstra habilidades específicas no tocante a categorização, e ao final faremos uma análise sugerindo algumas características essenciais a um modelo para categorização.

- Objetos a diferentes distâncias e em diferentes localizações na retina:
  - o ângulo visual varia quando nós olhamos objetos a diferentes distâncias e quando há objetos do mesmo tipo, porém com tamanhos diferentes;
  - também temos habilidade de categorizar um objeto mesmo quando ele se encontra na faixa lateral do nosso campo visual.

  A habilidade de categorizar sob estas situações demonstra que deve haver provavelmente alguma transformação da informação sensorial em características que sejam invariantes no tocante ao deslocamento e a escala. Esta transformação permite um mapeamento do que é percebido visualmente a uma mesma representação interna da forma dos objetos. Mesmo que o objeto seja visto com tamanhos variados ou em diferentes posições na retina. Esta habilidade é denominada usualmente de constância perceptual.

- Variação na forma dos objetos. O sistema biológico consegue categorizar objetos mesmo que a forma não corresponda exatamente a forma de objetos da mesma categoria vistos anteriormente. Algumas situações em que isto ocorre são:
  - objetos são vistos de pontos de vista diferentes;
  - objetos possuem forma deformável;
  - a relação espacial entre as partes dos objetos varia;
  - partes são acrescentadas ou subtraídas aos objetos.

  Isto sugere uma generalização da forma, a partir de informações de partes principais vistas, bem como o estabelecimento de uma hierarquia entre as partes do objeto.

- Empobrecimento de informação visual:

- objetos estão parcialmente oclusos;
- imagem do objeto está degradada (mudança em textura, cor);
- objeto encontra-se muito próximo do observador.

Novamente uma generalização e/ou uma complementação ajuda a resolver o problema aqui. Mais ainda, a informação visual obtida a um dado instante deve ser mantida de forma que nova informação extraída possa completar a formação de um padrão perceptual que permita categorização da instância em questão.

- Objetos em cenas complexas. Podemos citar aqui pelo menos duas habilidades diferentes:
  - uma instância específica de um objeto é segregada entre vários objetos que distraem a atenção;
  - vários objetos podem ser notados numa única fixação.

  Encontrar uma explicação para estas últimas habilidades não é trivial. Para a primeira delas, um estudo neuro-fisiológico, isto é, (FARAH, 1990), sugere inclusive a existência de mecanismos específicos para identificação de faces e para leitura que não são usados em outras tarefas envolvendo categorização. Para a segunda, pode-se sugerir algum processamento em paralelo ou mesmo um processo de construção (ou cópia) de uma imagem mental (KOSSLYN, 1994) que seria mantida por algum tempo, mesmo após o desaparecimento do estímulo, numa memória de curta duração. Este recurso seria usado se a tarefa exigisse categorizar vários objetos numa única fixação.

Todas as situações e habilidades acima descritas demonstram a existência de invariabilidades, regularizações, generalizações e mesmo complementações ou formação de imagens mentais. Idealmente, as características de representação extraídas da informação sensorial não devem mudar para um mesmo objeto, mesmo que em situações diferentes, levando a um mesmo endereço de memória.

Sob a questão da invariabilidade, sistemas de representação invariantes podem ser sugeridos para armazenar informação relativa a forma dos objetos. Podemos classificar estes segundo dois paradigmas opostos:

- *(A)* - Representação centrada no objeto: um modelo geométrico único e completo do objeto possui todas as informações inerentes a sua forma. Note que a informação sensorial pode ter que sofrer uma transformação genérica para ser levada a representação correspondente armazenada e também que objetos complexos podem ser de representação difícil ou de alto custo segundo este modelo.

- *(B)* - Representação centrada no observador: segundo este modelo, várias representações, cada uma obtida de um ponto de vista diferente são armazenadas. Note que a informação sensorial não necessita transformação de rotação (no máximo translação e escala) para ser levada a representação armazenada. Porém, para um objeto complexo, inúmeras vistas de várias posições diferentes podem ser necessárias para bem representá-lo.

Quanto ao aspecto regularização, as teorias que prevalecem para reconhecimento e identificação podem ser classificadas em:

- *(A)* - Métodos que usam propriedades de invariância: assumem que certas propriedades simples se mantêm invariantes sob as transformações que o objeto possa sofrer. Isto leva às técnicas que usam espaços de características invariantes, técnicas de agrupamento ou "clustering", e técnicas de separação ou segregação.

- *(B)* - Métodos que usam decomposição em partes: baseiam-se na decomposição do objeto em partes inter-relacionadas. Isto leva às técnicas de descrição estrutural simbólica, hierarquia de características e reconhecimento de padrão sintático.

- *(C)* - Métodos que usam alinhamento do estado perceptual: a idéia nesta classe de métodos, sugerida por Ullmann em (ULLMAN, 1996), é introduzir uma compensação para as transformações que separam um objeto visualizado e sua representação correspondente armazenada, e então, compará-las.

Podemos dizer que nossa proposta, quanto ao sistema de representação, combina os dois opostos descritos acima (nem no observador e nem no objeto), uma vez que usamos um conjunto de características que suporta bem a transformação de rotação combinadas com invariância nas relações de adjacência entre as partes de um objeto. Quanto ao aspecto regularização, o método usa propriedades de invariância, como no modelo *(A)*.

2.1.3 Trabalhos Relacionados e Análise das Soluções

Uma boa referência no sentido de prover um modelo computacional que explique a neuro-fisiologia da atenção pode ser encontrada em (VAN DER LAAR *et al.* , 1995; VAN DER LAAR *et al.* , 1997). Nestes trabalhos, usando imagens estacionárias, uma extração de multi-características é executada, calculando-se vários mapas de características. Uma rede neural atencional recebe uma entrada dependente da tarefa sendo executada e direciona informação do mapa de características para o

mapa de saliência. Então, o próximo local no qual o sistema tem que prestar atenção é simplesmente dado pela posição mais "saliente" no mapa de saliências.

Usando um modelo similar ao acima, Itti et al. propõem em (ITTI *et al.* , 1997) um modelo para atenção que usa filtros lineares sintonizados em várias orientações e vários períodos para computar uma resposta linear ao estímulo visual independente de fase. Estes filtros lineares interagem através de combinações excitatórias e inibitórias não lineares. Um modelo de erros juntamente com uma estratégia de decisão são assumidos para relacionar a saída do método a dados psicofísicos. O mapa de saliências é calculado e um modelo estatístico determina a próxima janela de atenção.

O trabalho de Westelius (WESTELIUS, 1995) trata o problema de controle do foco de atenção para visão robótica, usando uma plataforma de simulação. Um método para determinar o foco de atenção é apresentado, baseado em uma filtragem denominada de convolução normalizada, desenvolvida inicialmente para filtragem de dados incompletos e com incerteza. O método torna invisíveis as partes da imagem que já tenham recebido atenção, permitindo ao sistema explorar novas áreas ou eventos. Em cancelando sinais conhecidos ou já modelados, a atenção do sistema é mudada para novos eventos ainda não descritos.

Juntando os tópicos atenção e identificação, Rybak (RYBAK *et al.* , 1998), tambem usando um modelo simples com imagens estacionárias monoculares, trata percepção e cognição como processos comportamentais. Estes incluem o processamento paralelo de fragmentos de imagem dentro da janela de atenção e uma varredura sequencial na imagem pela janela de atenção. Reconhecimento de padrões é codificado em memória como uma sequência de movimentos dos olhos com uma verificação esperada dos fragmentos espaciais de imagem nos locais. Os dois caminhos bem conhecidos "what" e "where" são codificados usando redes neurais. Um objeto é reconhecido sequencialmente se o padrão de movimento do olho e cada padrão de fragmento de imagem invariante correspondente são similares a uma representação daqueles armazenada em memória.

Kosslyn (KOSSLYN, 1994) sugere um bom modelo descritivo para explicar como identificação e reconhecimento acontecem. O modelo sugere que características extraídas diretamente das imagens são usadas juntamente com imagens mentais formadas pelo cérebro (por complementação das imagens dos objetos). Apesar da falta de explanações melhores e de dificuldades práticas, é intuitivamente atrativo desenvolver um sistema visual de acordo com aquela descrição.

Dois bons trabalhos provendo modelos de características para reconhecimento e identificação podem ser encontrados em (BALLARD, 1991; RAO & BALLARD, 1995). Esses trabalhos apresentam um conjunto de operadores baseados em derivadas gaussianas para extração de características. Os operadores são sugeridos como

sendo similares a operadores do modelo biológico.

Em nosso ponto de vista, exceto (KOSSLYN, 1994) que não é um sistema implementado mas apenas um modelo descritivo, todos os trabalhos acima apresentam falhas, recaindo em um ou mais dos seguintes aspectos:

1. O trabalho não provê um modelo básico que propicie cognição, que a princípio inclui pelo menos atenção (eventualmente envolvendo movimentos em hardware) e categorização de padrões.

2. O trabalho não considera um conjunto de características que seja suficiente para atenção e/ou categorização. No trabalho de Rybak, por exemplo, não são considerados padrões de movimento nem disparidade estéreo para ajudar no reconhecimento, mas apenas uma percepção planar sequencial. Segundo este modelo, poderia haver ambiguidades entre certas formas de objetos.

3. O trabalho foi desenvolvido em simulação, usando imagens estacionárias, não considerando aspectos temporais (como movimento) ou funcionais e comportamentais. Além do mais, não há uma preocupação em prover uma resposta eficiente aos estímulos ambientais em tempo real.

No presente trabalho, tanto para atenção quanto para categorização de padrões, nós preferimos adotar uma abordagem que usa características baseadas em intensidade e textura, além de outras como padrões de disparidade estéreo e de movimento. Essas características são calculadas a partir das respostas de filtros baseados em um conjunto modificado de derivadas gaussianas parciais (ordem 0, 1 e 2 em duas direções cada).

Para realizar atenção, nós usamos um modelo que lembra (VAN DER LAAR *et al.* , 1995; VAN DER LAAR *et al.* , 1997), usando como características atencionais básicas as magnitudes de cada par de respostas dos filtros acima, para cada ordem de derivada, em cada posição da imagem. Como nós trabalhamos com sequências de pares de imagens, padrões de movimento (em realidade as duas derivadas direcionais da diferença de quadros consecutivos) e padrões de disparidade estéreo são também levados em consideração para determinar a próxima região na qual o sistema tem que prestar atenção, ou seja, a geração de um mapa de saliências. Nosso modelo necessita de efetivamente promover movimentos sacádicos, envolvendo eventualmente também movimentos de pan e tilt, além do motor de vergência, para conseguir colocar a janela de atenção na fóvea (centro do quadro ou imagem corrente).

Para categorização, momentos locais são calculados a partir das respostas dos filtros gaussianos e, juntamente com outros momentos calculados a partir dos padrões de disparidade estéreo e padrões de movimento (ambos calculados pela parte de atenção) são usados como entrada para uma rede neural do tipo "perceptron"

em multi-camadas treinada com um algoritmo do tipo "back-propagation" (BP) (RUMELHART *et al.* , 1986; WERBOS, 1988; BRAUN & RIEDMILLER, 1993; BALLARD, 1997). Esta memória (rede BP) lembra o endereço de memória da representação considerada (um índice para cada representação ou objeto). Essas características (momentos) introduzem alguma invariância quanto a rotação, translação e escala (ver Capítulo 5 ou (RAO & BALLARD, 1995; BALLARD *et al.* , 1999; RAVELA & MANMATHA, 1997; RAVELA & MANMATHA, 1998)), devido ao que as denominamos de momentos semi-invariantes. Assim, podemos considerar o processo de identificação como sendo uma sobrejeção finita de $\Re^n$ (um conjunto de n características em $\Re$) em N^+ (índices dos objetos). Ou seja, permitimos que mais de uma percepção possa e deva levar a um mesmo objeto. De um modo menos informal, podemos dizer que uma percepção corrente ativa alguma informação na memória associativa, seja esta informação um endereço de memória (por exemplo, numa simples tabela) contendo mais informação a respeito dos objetos ou mesmo algo mais funcional como uma ordem para o acionamento de um controlador robótico. Após uma certa fase de treinamento, o sistema consegue identificar positivamente objetos, a partir de suas características e mesmo que nao tenhamos um objeto identificado em primeira instância, as representações mais ativadas conterão informação suficiente para "guiar" o robô em tempo real, na escolha da próxima ação que deverá buscar uma identificação positiva do objeto.

A maior diferença de nosso trabalho para com os acima sumarizados situa-se no fato de que nós realizamos tudo isto em um sistema de visão ativa num robô real, exigindo um processamento em tempo real, ao invés de imagens estacionárias. Nós desenvolvemos e temos funcionando um modelo cognitivo mais completo, envolvendo o controle de atenção e categorização de padrões.

2.2 Reconstrução Estéreo e Disparidade

Tradicionalmente, muitos trabalhos têm tentado desenvolver sistemas visuais construindo modelos geométricos completos para um dado ambiente e então planejando ações para robôs baseadas naqueles modelos. Em nossa abordagem, não há a necessidade de se construir um modelo completo da cena a priori. Os processos visuais e os atuadores robóticos operam baseados em correção de erros (disparidade visual) ou seja, aplicando os deslocamentos necessários para colocar uma região objetivo na fóvea. Desta forma, além da disparidade estéreo ser considerada diretamente pelo processo de atenção ela é também usada como uma das características que farão parte do vetor de entrada para o processo de identificação. A disparidade estéreo é ainda usada para manter ambos os olhos vergidos numa região de interesse. Aqui reduzimos o problema de reconstruir completamente a forma de um objeto,

não usando representações geométricas destes, mas sim representações que usam relações de disparidades entre as partes de um objeto. É claro, resta ainda o problema de determinar a disparidade. No restante desta seção, colocaremos o problema estéreo em si e discutiremos as soluções mais comuns encontradas na literatura a respeito.

2.2.1 Reconstrução Estéreo

A técnica de reconstrução a partir de imagens estéreo encontra-se atualmente bem definida, com uma vasta bibliografia em livros textos como "Robot Vision" (HORN, 1986), "Vision" (MARR, 1982), "Computer Vision" (BALLARD & BROWN, 1982), "From Images to Surfaces" (GRIMSON, 1981), entre outros.

Na técnica tradicional de reconstrução estéreo, um modelo computacional da cena é construído, através de uma série de cálculos realizados sobre duas ou mais imagens digitais de uma mesma cena, tomadas de pontos de vista diferentes. A Figura 2.1 descreve um modelo simples não convergente (eixos em paralelo). Um ponto típico $P(x, y, z)$ é projetado em dois planos imagens (esquerdo=l e direito = r). Por conveniência de representação, os planos imagens estão rotacionados de um valor angular igual a π *radianos* em torno dos pontos focais.

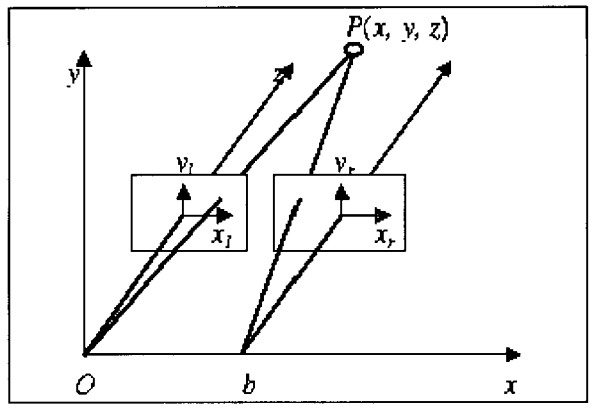


Figura 2.1: Modelo simples não convergente.

O parâmetro b, denominado linha de base (ou simplesmente base) é a distância entre os pontos focais ou centros de perspectiva das câmeras. O parâmetro f (distância focal), cujo valor é o mesmo para as duas câmeras, é a distância entre cada plano imagem e os pontos focais. Os sistemas de coordenadas de imagem esquerdo e direito são (x_l, y_l) e (x_r, y_r) e suas origens estão nas coordenadas (em sistema global) $(0, 0, f)$ e $(b, 0, f)$ respectivamente. Os pontos focais estão em $(0, 0, 0)$ e $(b, 0, 0)$. Usando similaridade entre triângulos, as seguintes Equações de projeção perspectiva são obtidas:

$$x_l = \frac{fx}{z}, x_l = \frac{f(x-b)}{z}, y_l = y_r = \frac{fy}{z} \tag{2.1}$$

Definindo a disparidade como $d = x_l - x_r$ (devido ao modelo ideal da Figura 2.1, esta só é definida para o eixo X), a partir das Equações 2.1 as seguintes Equações de inversão da projeção perspectiva podem ser obtidas:

$$x = \frac{bx_l}{d}, y = \frac{by_l}{d}, z = \frac{bf}{d} \tag{2.2}$$

Se tivermos a disparidade d determinada para todos os pontos da imagem, a profundidade em cada ponto pode ser determinada por simples similaridade entre triângulos, usando técnicas de integração numérica baseadas no cálculo variacional, tal como ocorre em (HORN, 1986).

2.2.2 O Problema da Correspondência Estéreo

Como nosso agente robótico atua baseado em correção de erros, como foi descrito no início da seção 2.2, na verdade, estamos em busca da disparidade d para cada píxel das duas imagens. Assim, podemos deixar de lado as Equações 2.1 e 2.2 e considerar apenas a Equação da disparidade $d = x_l - x_r$. Determinar a disparidade para cada píxel significa determinar a sua posição e a posição do seu correspondente na outra imagem. Isto pode ser formalizado da sequinte maneira: dadas duas estruturas numa imagem, e estabelecendo certas relações entre elas, o problema de correspondência estéreo consiste em se estabelecer estruturas correspondentes numa outra imagem (ou imagens) que podem estar transformadas por um movimento rígido (ZHANG, 1993).

O processo estéreo é considerado o gargalo em qualquer sistema de visão ativa e fatalmente envolve cálculos maciços de correlação, como nos trabalhos que seguem o modelo de Marr (MARR, 1982), ou então a aplicação de filtros locais direcionais (FREEMAN & ADELSON, 1991) como em (SANGER, 1988; FREEMAN & OHZAWA, 1990; WESTELIUS, 1995). Estes últimos trabalhos, baseados nos modelos de diferenças de fase local e de diferença de posição, envolvem cálculo de transformadas locais de Fourier. Estes e outros trabalhos envolvendo visão estéreo serão sumarizados na sub-seção seguinte.

2.2.3 Trabalhos Relacionados e Análise das Soluções

Um grande número de pesquisas em Visão Estéreo e Visão Ativa têm seguido o paradigma de Marr (MARR, 1982) e de Marr e Poggio (MARR & POGGIO, 1979). Veja por exemplo (NISHIHARA, 1984; NISHIHARA *et al.* , 1984; MATTHIES & BROWN, 1997; KOLLER *et al.* , 1994; DAVIS *et al.* , 1992; MATTHIES &

SHAFFER, 1987; RODRIGUEZ & AGGARWAS, 1990; MATTHIES *et al.* , 1995; WESSLER, 1996). Estes trabalhos, entre outros, propõem modelos que computam mapas de disparidade completos baseado em cálculo de correlação. Torna-se difícil projetar algoritmos para aplicações de tempo real em arquiteturas padrões (ou mesmo em arquiteturas pipeline) usando este paradigma.

Um algoritmo relativamente rápido usando correlação é descrito no trabalho de Huber e Kortenkamp (HUBER & KORTENKAMP, 1995). Usando uma arquitetura pipeline vetorial eles conseguem processamento a uma taxa máxima de alguns quadros por segundo (menos de cinco). Estudos em neuro-psicologia, isto é, (JULESZ & SAARINEN, 1991), revelam evidências de que em algumas tarefas o sistema biológico atinge processamentos com taxas de até uma dúzia de vezes mais (ou até 60 quadros por segundo).

Para ter uma melhor performance, poderia-se usar um modelo baseado no sistema biológico, oferecido por Sanger em (SANGER, 1988). Este modelo emprega filtros de Gabor, que são aproximações locais da transformada de Fourier, como base para o cálculo da disparidade (veja (FREEMAN & ADELSON, 1991) para mais detalhes sobre filtros direcionais). Este modelo usa o fato de que o deslocamento de uma função gera um deslocamento proporcional em sua transformada de Fourier. A disparidade binocular em cada posição é então proporcional a diferença de fase nos pedaços de superfície ("patches") considerados nas imagens esquerda e direita (FREEMAN & OHZAWA, 1990).

Pesquisas mais recentes (QIAN, 1994; QIAN & ZHU, 1997; FLEET *et al.* , 1997) propõem um novo modelo para a codificação neural da disparidade binocular. Os dados neuro-fisiológicos suportam dois modelos para a disparidade seletiva de células simples e complexas no córtex cerebral primário (FLEET *et al.* , 1997). Estes envolvem combinações de campos receptivos monocular que são deslocados em sua posição nas retinas (modelo de deslocamento de posição) ou deslocados em fase entre os olhos (modelo de deslocamento de fase). Os resultados mostram que estes modelos são uma aproximação computacional razoável para representar o cálculo da disparidade em modelos biológicos. Um trabalho prático (embora que em simulação) para o cálculo da disparidade usando diferenças de fase locais em modelo de multi-resolução pode ser encontrado em (WESTELIUS, 1995).

Na prática, os modelos acima descritos baseados em deslocamento de fase são tão caros computacionalmente quanto os modelos baseados em correlação (como por exemplo o usado em (HUBER & KORTENKAMP, 1995)). Isto é devido a quantidade de operações necessárias para representar os neurônios sensíveis a disparidade e também para calcular os filtros de Gabor nos modelos de deslocamento de posição e de fase. O projeto de um circuito integrado de propósito exclusivo seria necessário, envolvendo o controle das milhões de conexões necessárias num modelo biológico.

Note também que é impossível lidar com tal quantidade de informação nos processos da visão de mais alto-nível. Ao invés de uma reconstrução da forma 3D completa em alta resolução (em toda a imagem), nós realizamos medidas estéreo de forma adaptativa, num processo de várias escalas, onde cada nível provê uma estimativa da disparidade para o próximo nível. Além do mais, apenas um determinado nível é escolhido para ser processado, pelos processos de mais alto nível do sistema, provendo uma redução significativa de dados para a identificação e reconhecimento do padrão ou representação em consideração. Nosso modelo age de forma similar a (WESTELIUS, 1995) no tocante ao espaço de escalas, embora não usemos diferenças de fase, mas sim cálculos de valores de correlação.

2.3 Robótica, Controladores, Aprendizado Automático e Mapas Espaciais

Quando se fala em soluções para problemas de robótica, uma série de fatores devem ser ponderados. Primeiro, qualquer solução que tenha intenção de ser mais ou menos completa é de formulação complexa e torna-se computacionalmente cara. Ao contrário do que parece, é extremamente exaustivo desenvolver soluções que se imagina serem intuitivamente simples para organismos biológicos. Por exemplo, para fazer um braço robótico aprender como agarrar objetos apenas de uma determinada classe, um longo trabalho pode ser necessário, envolvendo modelagens físicas da cinemática e dinâmica do processo, bem como o desenvolvimento de técnicas de aprendizado com uso de ferramentas de Inteligência Artificial. Felizmente, soluções para problemas como este já existem. Atualmente, os problemas mais relevantes em robótica são voltados mais à integração de modelos e metodologias já existentes para contextos específicos do que propriamente o desenvolvimento de novas soluções (por exemplo, integrar "grasping" e visão). Além do mais, com o avanço de tecnologias de fabricação de chips VLSI ("very large scale of integration"), controladores podem ser implementados em hardware, tornando-se extremamente rápidos, sendo também possível encontrar equipamentos de robótica que já vêm com alguns procedimentos em software já desenvolvidos (planejador de movimento, controladores, etc).

Assim, neste trabalho, os problemas de robótica relacionam-se mais com o controle de decisão, tal como em (ARAUJO & GRUPEN, 1996; COELHO & GRUPEN, 1997; HUBER & GRUPEN, 1997), do que com a formulação de bases de controle (HUBER *et al.* , 1996; MACDONALD, 1996; SCHNACKERTZ & GRUPEN, 1995; SOUCCAR *et al.* , 1998) ou o desenvolvimento de controladores robóticos (GRUPEN, 1999). Nós tentamos modelar uma estrutura de controle segundo um contexto definido pelo estado perceptual do robô e pela tarefa sendo executada. Em nosso modelo, um controlador opera em ciclo, transformando entrada em saída para satis-

fazer uma estratégia de controle ou "política". A entrada é sempre relativa ao estado perceptual corrente, contendo informação sensorial e a pose do robô. A transformação que ele efetua pode ser uma ação física, tal como um movimento gerado pelos atuadores robóticos, ou outras ações não físicas, tais como a realização de simples cálculos ou a aplicação de operadores morfológicos a uma imagem. A saída, geralmente em forma de relatório, atualiza a pose do robô ou o estado perceptual. Uma vez que um controlador atinge uma condição de equilíbrio satisfazendo a estratégia de controle, variáveis de estado são atualizadas em um vetor de predicados (ou vetor de estados). Este vetor de predicados é compartilhado entre os vários controladores. Dependendo dos valores deste vetor e da tarefa sendo executada, um determinado controlador irá atuar, mudando o vetor de predicados novamente. Desta forma, uma política de controle ou programa comportamental é estabelecido, usando um conjunto de controladores e um conjunto de variáveis de estado representadas no vetor de predicados. Visando automatizar a busca de uma solução, é possível formular a tarefa de modo mais complexo como sendo um processo markoviano, e tentar usar técnicas de otimização ou de aprendizado para buscar uma solução. Neste sentido, o conceito de controlador (GRUPEN, 1999) torna-se mais complexo, envolvendo mais do que o cálculo de movimentos ou ações físicas dos braços ou outros componentes de uma arquitetura robótica. Dentro de uma política de controle mais global, um controlador poderia ser ainda composto por vários outros controladores que atuam num nível mais específico.

Um Processo de Decisão Markoviano (MDP) é um processo estocástico cujo passado não influi no estado futuro do processo, se o estado presente se encontra totalmente especificado (PAPOULIS, 1991). Em robótica, quando se modela um processo como um MDP, uma política para uma dada tarefa consiste na ativação sequencial de um ou mais controladores robóticos em um ciclo de controle, geralmente atendendo-se a um conjunto de restrições vinculado a tarefa que está sendo executada, para tentar atingir o objetivo da tarefa. Um ciclo de controle é estabelecido sobre a máquina de estados finitos que define o MDP. Em geral, definir uma política satisfatória para um MDP envolve o uso de programação dinâmica, podendo-se usar, em particular, técnicas de aprendizado por reforço para buscar essa solução.

Aprendizado por reforço (ou "reinforcement learning") e mais especificamente Q-learning (WATKINS, 1989; BALLARD, 1997; SUTTON & BARTO, 1998) é basicamente uma forma de resolver problemas de aprendizado de controle que possam ser modelados como um MDP usando programação dinâmica. Um espaço de estados e um conjunto de ações que determinam a passagem de um estado a outro são definidos numa tabela denominada "*Q-table*". As linhas desta tabela representam os estados e as colunas representam as ações. A idéia é dar recompensas para as melhores ações executadas de cada estado. Assim, após a execução de uma ação, o valor da

célula da *Q-table* determinada pelo par (estado-ação) é atualizado. Se cada ação fosse executada infinitas vezes para cada estado, cada valor (ou "*Q-value*") da *Q-table* convergiria com probabilidade 1 para valores que representassem uma solução ideal, onde a melhor recompensa é dada para cada par estado-ação (WATKINS, 1989; WATKINS & DAYAN, 1992). Na prática, não são necessárias infinitas tentativas, uma vez que após um certo número de ciclos de controle consegue-se uma estimativa das melhores ações a serem realizadas de cada estado. A seguinte função $Q(s,a)$ é usada em cada passo, pelo processo de treinamento para atualizar os *Q-values*:

$$Q(s,a) = Q(s,a) + \alpha(r + \gamma \max_{a' \in A} Q(s',a') - Q(s,a)) \tag{2.3}$$

As variáveis $< s, a, r, s' >$ sumarizam uma transição simples no ambiente. Aqui, s é o estado do agente antes da transição, a é a ação escolhida, r a recompensa instantânea que ele recebe e s' o estado resultante. A variável a' representa todas as ações possíveis de serem escolhidas do conjunto A no estado resultante s'. O parâmetro $\boldsymbol{\alpha}$ é a taxa de aprendizado desejada para o sistema, e $\boldsymbol{\gamma}$ é um fator de desconto para o aprendizado verificado no passo anterior. Valores típicos para estes parâmetros são geralmente 0.9 para α e 0.5 para γ.

Geralmente, para selecionar uma ação nesta fase de treinamento usa-se uma função que se baseia em exploração ou explotação (ARAUJO & GRUPEN, 1996), dependendo do estágio em que o treinamento se encontra. Nos experimentos realizados neste trabalho, nos atemos a exploração, usando um selecionador de ação estocástico que inclui a distribuição de probabilidades de Boltzman. Segundo esta distribuição, a probabilidade de selecionar a ação a no estado s é dada por:

$$p(a|s,T) = \frac{e^{Q(s,a)/T}}{\sum_{a' \in A} e^{Q(s,a')/T}}, \tag{2.4}$$

onde T é uma temperatura, $Q(s,a)$ e $Q(s,a')$ são funções de avaliação para os pares de estado-ação (s,a) e (s,a'), como definido acima, para qualquer a ou $a' \in A$, sendo A o conjunto de ações. Num dado estado s, o percentual de exploração é determinado pela temperatura T e pelo grau de distinção da função de avaliação para as diferentes ações deste estado. Assim, um aspecto chave neste método é a escolha da temperatura inicial e o seu fator de decaimento.

O algoritmo Q-learning, melhor descrito em (WATKINS, 1989), pode ser resumido como a seguir:

Algoritmo Q-learning:

1. *Definir o estado corrente* $\boldsymbol{s}$ *decodificando a informação sensorial disponível;*

2. *Usar o selecionador estocástico de ação (Equação 2.4) para determinar uma ação* $\boldsymbol{a}$;
3. *Executar ação* $\boldsymbol{a}$, *gerando um novo estado* $\boldsymbol{s'}$ *e uma recompensa* $\boldsymbol{r}$;
4. *Calcular o erro de diferença temporal* $\hat{\boldsymbol{r}}$:

$$\hat{r} = r + \gamma \max_{a' \in A}(Q(s', a')) - Q(s, a); \tag{2.5}$$

5. *Ajustar o Q-value do par estado-ação* $(\boldsymbol{s}, \boldsymbol{a})$:

$$Q(s, a) = Q(s, a) + \beta \hat{r};$$

6. *Retorne ao passo 1.*

Ultimamente, *Q-learning* tem sido muito empregado para encontrar soluções satisfatórias para problemas que envolvam combinações de alternativas comportamentais diversas, como o descrito acima. Em (ARAUJO & GRUPEN, 1996), *Q-learning* foi usado para definir uma estratégia comportamental do agente (um sapo) frente a uma tarefa de sobrevivência e alimentação num ambiente silvestre. Em (COELHO & GRUPEN, 1997), *Q-learning* é empregado em tarefas de "grasping" e em (HUBER & GRUPEN, 1997) ele é usado em tarefas onde um agente robótico com várias pernas aprende como se locomover no seu ambiente (HUBER & GRUPEN, 1997).

2.3.1 Mapeamento Topológico do Ambiente

Em se tratando do sistema biológico, não há uma comprovação formal de que seja mantida uma representação interna do ambiente, mas há evidências, a partir de estudos neuro-fisiológicos (ver por exemplo (COLLET *et al.* , 1986; REDISH & TOURETZKY, 1997)), da existência de uma representação do ambiente, registrada topograficamente, na área do hipocampo. Este mapa seria usado para orientação e locomoção. Note que não queremos aqui abordar em profundidade questões relativas a existência ou não de áreas do cérebro que contenham mapas topológicos do ambiente em modelos biológicos, mas sim usar este paralelo para atentar para a importância computacional dos mesmos. Embora manter uma representação do ambiente possa não ser necessária no sistema biológico, conjecturamos que ela é essencial em se tratando de sistemas robóticos. Claro, há maneiras de se evitar a manutenção de um mapa do ambiente, mas isto pode se tornar caro computacionalmente.

A questão principal a ser respondida a seguir é determinar o porque da necessidade de se manter um mapa interno. No modelo adotado neste trabalho, os controladores robóticos de movimento operam em ciclo, executando movimentos diferenciais a cada intervalo de tempo em que esteja operando. Cada vez que um

controlador opera, erros diferenciais de posicionamento são recalculados e uma correção (diferencal) é aplicada, corrigindo o movimento do robô. Em outras palavras, o torque e velocidade necessários para levar o robô ao objetivo são recalculados a cada passo. Assim, é necessário manter um referencial no ambiente para poder definir a posição do objetivo em relação a posição corrente, sem as quais não seria possível calcular o deslocamento. Esta posição de referência em relação a posição corrente do robô fica exatamente determinada se construirmos um mapa do ambiente, contendo ambas. Note que um modelo da região do objetivo poderia ser empregado para definir a posição do objetivo a cada passo, calculando-se então o deslocamento. Porém, este processo envolveria estabelecer a correspondência entre o modelo e um certo número de regiões nos mapas sensoriais, o que exige algum esforço computacional, sendo aparentemente inviável. Assim, com a construção de um mapa do ambiente, uma simples checagem é suficiente para definir o deslocamento a cada passo. Note que, ao final, o robô poderá eventualmente não estar exatamente sobre o objetivo, mas tão próximo dele quanto a precisão do seu sistema de odometria o permitir.

Além dos argumentos acima, no tocante a eficiência computacional, outro fator relativo a agilizar a mudança do foco de atenção nos levou a optar pela existência de mapas atencionais (ou mapas de saliência). Assim, não temos formalmente um mapa do ambiente, mas sim mapas atencionais que servem também para esta função de mapeamento topológico. Esses mapas serão vistos em maiores detalhes durante o decorrer do texto.

2.3.2 Trabalhos Relacionados

Aproximação e apreensão ("reaching and grasping") por braços robóticos são outros tópicos muito estudados na última década. Estes problemas de robótica também têm tido soluções inspiradas na biologia. Muitos trabalhos têm tentado imitar ou reproduzir habilidades de infantes para aproximar e tocar ou agarrar um objeto apresentado ou responder em forma de ações ou movimentos a um estímulo apresentado. Um trabalho importante estudando os sistemas biológicos da visão e tato em infantes e suas inter-relações é o livro de Streri (STRERI, 1993). O ato de aproximação a um objeto ("reaching") em recém-nascidos é sugerido ser um movimento balístico, estimulado (ou disparado) por percepção visual ou auditiva, sem controle algum durante a aproximação, enquanto que em bebês de 4 a 5 meses a aproximação é sugerida ser suavemente guiada, inclusive com uma eventual mudança de trajetória durante a aproximação. Isto pode sugerir o processo de aproximação como uma habilidade que já nasce com o sujeito, que desenvolve aprendendo de acôrdo com o desenvolvimento motor. Em (CLIFTON *et al.* , 1994; BERTHIER, 1996b; BERTHIER, 1996a), um trabalho teórico produziu um modelo matemático para o desenvolvimento do processo de aproximação. O modelo sugere que infantes

estão aprendendo constantemente sobre as capacidades correntes de seus sistemas motores e adaptando estratégias de aproximação de acordo com seu nível de controle motor corrente. Isto pode sugerir que o processo de aproximação não é uma habilidade de nascença, mas que pode ser aprendida, num modelo de estímulo e resposta. Os trabalhos de implementação do processo de aproximação em robótica estão recentemente emergindo para este lado biológico de aprendizado.

As propostas que não se baseiam em neuro-biologia geralmente modelam movimentos como funções diferenciáveis contínuas e tentam uma solução no cálculo integral ou por processos iterativos. Em (CONNOLY & GRUPEN, 1993), são usadas funções harmônicas para o cálculo do caminho a ser seguido por um robô ("path planning"). O método descrito, além de fornecer o melhor caminho, também permite ao robô desviar de obstáculos que já sejam conhecidos. No trabalho de Coelho (COELHO & GRUPEN, 1997), informações proprioceptivas e táteis são usadas para prover uma estratégia robusta para apreensão robótica. A ferramenta desenvolvida parece ser útil não apenas para a solução do problema de apreensão robótica, mas também para resolver outras tarefas complexas que possam ser modeladas como um POMDP (Processo de Decisão Markoviano com estados Parcialmente Observáveis (PAPOULIS, 1991)). Baseado em experiências de apreensão registradas anteriormente, um modelo de performance esperada é derivado. Num dado momento, baseado no estado atual e na experiência prévia, o controlador pode selecionar, on-line, a política mais efetiva para o contexto.

Um modelo interessante envolvendo mapeamento visual e óculo-motor pode ser encontrado em (FERRELL, 1998). No trabalho em questão, são usados mapas multimodais, registrados e topograficamente organizados, do espaço sensorial-motor para orientar um robô (denominado de COG) na presença de estímulo ambiental. Um algoritmo usando aprendizado automático é usado para treinar o robô. Basicamente, para aprender as conexões entre os mapas oculo-motor e visual, estes são inicialmente conectados um ao outro através de campos receptivos com uma larga sobreposição. Então, o algoritmo tenta ajustar as conexões diminuindo suas vizinhanças. As conexões são atualizadas de acordo com uma função de aprendizado que usa uma distância-erro entre uma posição no mapa de movimento e uma posição alvo dada.

2.4 Discussões e Generalidades

Neste trabalho tentamos buscar uma solução que seja suficiente para o que chamamos de bases para a cognição em robótica: mudança do foco de atenção e categorização. Este problema é complexo e é muito dependente da tarefa que esteja sendo executada, além de depender de forma crucial da solução adotada para ou-

tros processos tais como a criação e manutenção de mapas espaciais ou reconstrução estéreo. Existem muitas soluções parciais (das quais algumas foram descritas neste capítulo), e outras, que podem ser consideradas relativamente completas mas que usam imagens estacionárias. Não encontramos na literatura a respeito nenhuma solução provendo uma solução tão completa como a que propomos aqui. Uma solução assim fatalmente deve integrar ferramentas diversas de áreas disjuntas de pesquisa, sendo um processo custoso.

Não é trivial integrar informação sensorial multi-modo num sistema de processamento único. Não se pode garantir que o sistema irá funcionar apenas juntando as soluções parciais, encontradas separadamente, ou estendendo a solução de sistemas desenvolvidos para um tipo de sensor para que suporte outros. Olhando esta integração por um outro lado, podemos afirmar categoricamente que, não importando o modelo sugerido matematicamente, é computacionalmente melhor para o sistema cognitivo usar em conjunto a informação sensorial provida pelos sistemas da visão, propriocepção incluindo tato, e provavelmente audição para aprender, integradamente aos sistemas motores, do que tentar desenvolver sub-sistemas independentes. Neste sentido, os sistemas sensoriais e motores podem operar como componentes coordenados na execução de uma dada tarefa. Se, por exemplo, a tarefa é pegar algum objeto percebido visualmente, erros no processo de vergência estéreo, no processo de aproximação e no processo de apreensão (resposta tátil) servem como retro-alimentação para o sistema integrado.

Capítulo 3

Arquitetura de Controle Multi-modo para Integração de Visão e Tato

Uma descrição funcional da arquitetura de um sistema para integrar o processamento das informações sensoriais providas pela visão e pelo sistema háptico (incluindo propriocepção e tato) é apresentada neste capítulo. A Figura 3.1 mostra os aspectos funcionais da arquitetura. Em resumo, a informação de entrada provida pelos dois sistemas sensoriais é organizada em áreas indexadas espacialmente, o "buffer" visual e o "buffer" háptico, os quais constituem juntos o que denominamos de "buffer" perceptual. A informação é agrupada de tal maneira que indicadores de natureza topológica e espacial obtidos a partir de informação sensorial múltipla são usados para formar um vetor (ou mapa) de características com representações de propriedades tais como intensidade, textura, forma, tamanho e peso. Este vetor de características é usado como um padrão para ativar uma memória associativa central. Esta memória associativa, de fato uma rede neural do tipo "back-propagation" (rede BP) agindo como um aproximador para uma função, estabelece a correspondência entre esse conjunto de propriedades e o índice de um registro em uma memória de longa duração (MLT ou "long-term memory"). Nesta última, além das propriedades citadas, cada registro possui também outros tipos de informação a respeito dos objetos, tais como fatos (nome, utilidade, funcionalidade) e o seu histórico de interação com o ambiente. Um mecanismo de controle da atenção (Controle Atencional) é necessário para mudar o foco de atenção de uma região para outra, baseado em informação provida pelo processo de pré-atenção e também em informação contida nos mapas topológicos espaciais (mapas atencionais). Estes últimos além de influírem no processo de atenção, armazenam características necessárias à detecção de mudanças no ambiente e também podem ser úteis na execução de outras tarefas como navegação e orientação. O supervisor de aprendizado automático armazena informação na MLT e treina novamente a rede BP toda vez que uma representação

ainda desconhecida for encontrada.

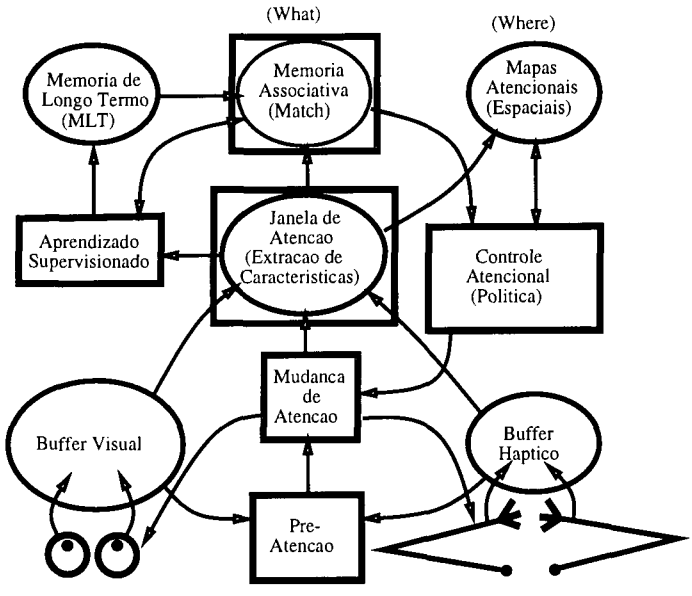


Figura 3.1: Arquitetura de controle para um sistema multi-modo.

Apesar do enfoque principal deste trabalho ser o contrôle da atenção e a categorização de objetos, a arquitetura proposta pode ser usada em outras tarefas dos mais diversos tipos. As partes componentes da arquitetura serão melhor detalhadas nas seções seguintes, com um enfoque maior nas tarefas de atenção e categorização.

3.1 Estado Perceptual ("Buffers" Visual e Háptico)

Estudos neuro-fisiológicos revelam que as imagens produzidas pelas retinas são projetadas primariamente em áreas do córtex visual localizadas na parte posterior do cérebro. (Veja o Apêndice B para mais detalhes do sistema visual biológico). Algumas pesquisas (MARR, 1982; MARR & POGGIO, 1979; GRIMSON, 1981; KOSSLYN, 1994; JULESZ, 1971) sugerem que estas áreas constituem uma estrutura chamada de "buffer" visual. Esta estrutura parece ser como um mapa em multi-resolução (tipo uma pirâmide) da cena percebida, armazenado não necessariamente em áreas corticais contíguas. A informação contida em um ou mais níveis de resolução flui do "buffer" visual para um processamento em mais alto-nível. O fato da presença desse "buffer" visual proporcionar uma maior eficiência computacional no processamento de mais alto nível justifica a sua existência, segundo os paradigmas de Marr (MARR, 1982) e de Nishihara (NISHIHARA, 1991). Segundo estes trabalhos, uma metodologia para estudar e definir modelos computacionais

para o sistema biológico deve levar em consideração alguns aspectos para justificar a existência das estruturas propostas para compor tal sistema:

- *A)* O primeiro aspecto refere-se à eficiência computacional do sistema com a inclusão de uma estrutura para o processamento específico de uma dada tarefa. Se o sistema operar com mais eficiência, em teoria, a estrutura deve existir.

- *B)* O segundo aspecto refere-se à divisão do trabalho. Se uma tarefa pode ser subdividida em duas outras tarefas que possam ser executadas mais rapidamente (geralmente de forma paralela) por duas estruturas ao invés de usar apenas uma estrutura (em forma sequencial), em teoria, as duas estruturas devem substituir a primeira.

Assim, neste trabalho a existência do "buffer" visual pode ser justificada pelo aspecto da eficiência computacional assumindo que um pequeno conjunto de características perceptuais é necessário e suficiente para que o processamento de mais alto nível possa se dar de forma efetiva e computacionalmente eficiente. Muita da informação provida pelos sensores é ambígua e desnecessária. A informação realmente útil é aquela relacionada com o objeto que é o foco de atenção corrente. Regiões com características similares ou relacionadas de alguma forma podem ser agrupadas de maneira que as características dos objetos: forma, textura, cor, localização espacial, orientação e tamanho possam ser extraídas facilmente e sistemas perceptuais cognitivos possam ser orquestrados de modo eficiente.

O "buffer" háptico contém informação sensorial do braço e das mãos, mais especificamente, informação proprioceptiva e táctil. A informação proprioceptiva relaciona-se com a pose do robô, ou seja, as configurações das juntas e ligações, incluindo os seus posicionamentos relativo, torque e velocidade correntes. Assim, propriedades dos objetos tais como tamanho, peso, rugosidade e propriedades espaciais como posição e orientação podem ser extraídas do "buffer" háptico. Em sistemas biológicos, o "buffer" do braço e mãos é mapeado na região somato-sensorial do córtex cerebral, onde a informação é armazenada segundo uma certa ordem topológica que facilita também a localização da informação proprioceptiva e táctil recebida.

3.2 Controle da Atenção

Em nosso modelo, denominamos a atenção "bottom-up" simplesmente de involuntária (causada por estímulo visual) e a atenção "top-down" de voluntária (guiada internamente). Atenção involuntária é ativada por um mecanismo de pré-atenção e ocorre quando um estímulo com forte ativação (alta intensidade de cor, movendo-se rapidamente, com um tamanho visual grande, ou mesmo estímulo táctil inesperado) requer que a janela de atenção seja posicionada numa dada região. A atenção

voluntária é ativada por um mecanismo de decisão que basicamente força a escolha de uma região, nem sempre a mais ativada, de uma mapa de saliências. Em linhas gerais, este mecanismo age da seguinte maneira:

- A atenção pode ser mantida na mesma região; esta decisão é tomada a partir de um padrão de ativação fornecido pela memória associativa que define o quanto uma região ainda necessita ser o foco de atenção para que uma identificação positiva possa ocorrer.

- A informação contida na memória de longa duração direciona a atenção a uma determinada região com o objetivo de buscar um detalhe que confirme ou descarte uma representação ("zoom" numa determinada região, procura por uma característica que confirme uma hipótese espacial, ou a procura em regiões sabidas ser de alto interesse, como bordas por exemplo) ou ainda, esta informação da MLT aciona um movimento do braço para tocar e mover um objeto melhorando a sua visibilidade, ou para pegar o objeto, extraindo o peso ou informação táctil. O endereço dessa informação contida na MLT é determinado pela memória associativa através do padrão de representação mais saliente, mas cujo valor de ativação ainda esteja abaixo de um dado limiar. Este padrão se refere a um potencial objeto mas cuja identificação ainda não está confirmada.

- Uma região do ambiente ainda não visitada ou já visitada há bastante tempo, o que pode ser determinado por uma simples inspeção nos mapas topológicos ou atencionais, pode requerer que a atenção seja voltada para ela.

- O braço toca algum objeto inesperadamente, e exige que a atenção se volte na direção daquele objeto.

Em outras tarefas mais gerais, não estudadas neste trabalho, a atenção pode ser direcionada por um mecanismo de controle atencional, de acordo com uma política definida especificamente para a tarefa. Um exemplo pode ser a busca de um determinado estímulo visual que possua propriedades similares a um dado modelo, o que ressalta o aspecto voluntário. Outro exemplo se dá quando o robô sabe que um estímulo pode aparecer numa determinada região do ambiente, então a janela de atenção pode ser direcionada para aquela região por um mecanismo de pré-atenção, esperando pelo aparecimento do estímulo (por exemplo, o robô pode aprender que pessoas geralmente aparecem e desaparecem de um ambiente pela porta de entrada ao mesmo). Noutro caso, pode ser necessário mudar a janela de atenção para um objeto que se situe numa dada direção já conhecida antes do movimento. Para ilustrar esta situação, citamos a tarefa de leitura, em que a janela de atenção é movimentada voluntariamente da esquerda para a direita e para baixo ao final de cada linha. Já

ao fazer a contagem de um conjunto pequeno de objetos, em geral a atenção evolui continuamente, isto é, passa de um elemento para o vizinho, tentando-se evitar repetições e também fazendo com que ao final todos os objetos do conjunto tenham sido observados (contados).

3.2.1 Janela de Atenção

A janela de atenção é uma estrutura que permite a extração de informação contida numa das regiões organizadas espacialmente no "buffer" perceptual. Sua posição não é fixa dentro desse "buffer". Entretanto, devemos notar que a informação mais útil, isto é, a profundidade estéreo fornecendo a forma, pode ser extraída se o foco de atenção se encontrar no centro da interseção dos campos visuais dos dois olhos, local este conhecido como o "horopter". Idealmente, o tamanho e a forma da janela de atenção deve variar dinamicamente, de acordo com a forma e tamanho visual dessas unidades perceptuais (regiões de interesse). Isto foi possível na implementação que fizemos em simulação como veremos mais adiante, mas na prática, na outra implementação realizada na plataforma de hardware nós não usamos este modelo de janela de atenção (ROI) por dificuldades práticas em implementar um método para segmentação das imagens em regiões de interesse que funcione em tempo real. Vários trabalhos propõem boas soluções para o problema de segmentação de imagens, em imagens estacionárias, mas provavelmente nenhum provê soluções compatíveis com o processamento em tempo real. As soluções mais gerais para imagens estacionárias podem usar desde simples detecção de bordas ou uso de operadores morfológicos (BEVERIDGE *et al.* , 1989; HARALICK *et al.* , 1987; JAIN, 1989; GONZALES *et al.* , 1992; HANSON *et al.* , 1998) até o uso de T-snakes e baloons (KASS *et al.* , 1988; COHEN, 1991; MCINERNEY & TERZOPOULOS, 1995; MCINERNEY, 1997) ou snakes e T-snakes duais (GUNN & NIXON, 1997; GIRALDI & OLIVEIRA, 1999).

3.2.2 Pré-atenção e Mudança de Atenção

O mecanismo de pré-atenção opera em baixo nível calculando valores de ativação para cada região de interesse, em função dos quais é estabelecida uma prioridade dela se tornar o foco (janela) de atenção. Assim, o mecanismo de atenção seleciona entre várias regiões a mais saliente, decidindo onde colocar a atenção. Esse mecanismo muda efetivamente a janela de atenção, desvinculando-a da posição corrente e vinculando-a numa nova posição nos "buffers" visual ou háptico. Os olhos executam movimentos sacádicos para mudar o foco de atenção corrente. Em organismos biológicos, considera-se que o processo de mudança da atenção possui dois componentes básicos: um que efetivamente move o corpo, a cabeça, os olhos e/ou a janela de atenção e outro que ao mesmo tempo codifica e mantém uma representação em

memória, aprimorando-a gradativamente enquanto o foco de atenção for mantido na região em questão ou próximo dela. De fato, enquanto o foco de atenção se mantiver num mesmo objeto, a informação que chega é cada vez mais facilmente incorporada a essa representação.

Note que os braços podem operar também na mudança do foco de atenção. Se um braço está se movendo e eventualmente toca algum objeto, a janela de atenção pode ser redirecionada para ele. Em algumas situações, os olhos se voltam efetivamente para este estímulo inesperado para reconhecê-lo ou identificá-lo, ou ainda para colocar o sistema em um estado funcional que lhe permita lidar com esse evento táctil.

3.2.3 Comportamento de Inspeção ou de Monitoração

Numa situação em que todas as regiões de interesse já tenham sido visitadas e em consequência um mapa do ambiente tenha sido construído, um comportamento de inspeção ou monitoração deve ser adotado. Neste comportamento, o robô age de forma ativa, mudando seu foco de atenção de uma região a outra, visando detectar possíveis mudanças que venham a ocorrer no ambiente. Na prática, para manter o sistema neste estado de inspeção ou de monitoração nós adicionamos uma variável de ativação extra (*interesse*) para cada região detectada no ambiente. Toda vez que uma região é visitada, o valor zero é atribuído à variável *interesse* correspondente. Então, a cada ciclo de controle, esse valor sofre um incremento e, em função do tempo decorrido, a um dado momento o valor será alto o suficiente para forçar o sistema a escolher uma região que já tenha sido visitada há bastante tempo e que eventualmente possuirá o maior valor do somatório dos valores de ativação, para ser o foco de atenção novamente. Assim, o sistema é guiado internamente ou forçado a focalizar em alguma região previamente visitada. Isto permite detectar mudanças ocorridas no ambiente e coloca o robô num estado comportamental ativo. Deve-se observar que, em geral, o sistema não irá visitar a mesma regão consecutivamente, mas que essa região poderá ser observada de novo antes de o sistema varrer de forma completa o ambiente. Isto depende da função usada para determinar a região mais saliente do ambiente.

Sem este mecanismo, quando o ambiente não tivesse mais regiões a serem mapeadas, o robô permaneceria com seus olhos estáticos, vergidos na última região visitada até que ocorresse uma mudança dentro do campo visual do robô. Apesar de não permitir o robô detectar mudanças fora do campo visual, este último modelo também é útil para ambientes muito dinâmicos. Assim, o sistema pode usar ambos os modelos e decidir em tempo de execução qual será aplicado, dependendo da tarefa.

3.3 Memória Associativa (Categorização)

A categorização de objetos e outras tarefas requerem que informação sensorial de diferente natureza seja associada à uma mesma representação em memória. O endereço desta representação é fornecido por uma memória associativa. A representação que contenha propriedades mais similares àquelas do padrão de entrada corrente se torna a mais ativada. Se o valor desta ativação estiver acima de um limiar, considera-se que o padrão perceptual da ROI corrente foi categorizado. Caso contrário, mais ou melhor informação deve ser provida. Se após um certo número de tentativas visando melhorar a informação de entrada, o limiar acima não tiver sido atingido ainda, considera-se que um objeto ainda desconhecido foi descoberto e nova informação e fatos relativos a ele devem ser armazenados na memória de longa duração. Então, um supervisor de aprendizado é automaticamente acionado para "aprender" algumas coisas sobre a nova representação. Aqui, "aprender" significa inserir as novas características e outras informações como por exemplo um nome ou índice para a representação na MLT e treinar novamente a memória associativa já com o novo padrão incorporado.

A memória associativa especifica, a partir da representação do objeto correntemente em foco, o endereço do padrão mais próximo a ela contido na MLT. Note que representações do tamanho efetivo do objeto (este não é o tamanho visual, mas sim uma característica invariante) e das outras propriedades como intensidade, textura, forma, e um nome (ou um índice identificador) são também armazenadas. Uma possível implementação para essa memória assiciativa é através de uma rede neural. De fato, nós usamos uma rede do tipo "perceptron" com multi-camadas, treinada com um algoritmo do tipo "back-propagation" (BPNN) (RUMELHART *et al.* , 1986; WERBOS, 1988; BRAUN & RIEDMILLER, 1993; BALLARD, 1997), com um mecanismo do tipo "winner-take-all" para escolher o endereço do padrão mais ativado. (Veja o Apêndice A para maiores detalhes do funcionamento de redes neurais do tipo BPNN.)

Para o treinamento da rede são necessários pares de entrada e saída associados (conhecidos). Assumindo que o robô a princípio não possui nenhum conhecimento, a cada padrão diferente encontrado no ambiente, um par de características de entrada e seu índice associado na saída da rede é formado. Assim, a medida que novas representações vão sendo descobertas no ambiente pelo agente robótico, a última camada da rede BP, inicialmente com dois objetos fictícios, é acrescida de um nó, sendo a topologia da rede remodelada (nós são também acrescidos na camada intermediária) e novamente treinada. Um epoch, ou período completo na fase de treinamento, é dado por uma passagem completa por todos os pares entrada/saída conhecidos, em ordem aleatória. Geralmente, centenas de "epochs" são necessários para que se obtenha uma convergência satisfatória do processo de treinamento. Após este treinamento

(cada novo objeto inserido), determina-se um limiar (κ) a ser usado pelo processo de identificação para definir se um padrão percebido é desconhecido ou se já possui uma representação na MLT. Este limiar é dado por:

$$\kappa = 1 - \omega_1 \varepsilon_{max} - \omega_2 \varepsilon_{min}, \tag{3.1}$$

onde, $\omega_1 + \omega_2 = 1$ e, para todos os pares de vetores de entrada-saída (X_i, Y_i) usados para o treinamento, sendo y_{ij} a saída desejada para cada nó j da última camada e o_{ij} a ativação calculada pela rede para estes nós, os erros ε_{max} e ε_{min} são medidos como:

$$\varepsilon_{max} = Max(|y_{ij} - o_{ij}|) \tag{3.2}$$

$$\varepsilon_{min} = Min(|y_{ij} - o_{ij}|). \tag{3.3}$$

A equação 3.1 acima representa uma função ponderada simples dos erros mínimo e máximo experimentados na última camada da rede para um epoch completo. Os pesos ω_1 e ω_2 podem ser ajustados de acordo com a maior ou menor rigorosidade desejada para essa decisão.

Um outro classificador poderia ser usado aqui, ao invés da rede BP. Por exemplo, os usados em (VIOLA, 1996; COELHO *et al.* , 1999; MARENGONI *et al.* , 1999; PIATER & GRUPEN, 1999; KOHONEN, 1990). O três primeiros trabalhos citados (VIOLA, 1996; COELHO *et al.* , 1999; MARENGONI *et al.* , 1999) usam uma rede bayesiana, e baseiam-se em modelos probabilísticos para extração de características das imagens e a consequente identificação. Em (PIATER & GRUPEN, 1999), baseado em um conjunto mínimo de características ("texels" e "edgels"), o sistema aprende a escolher a melhor combinação topológica delas que lhe permita discriminar visualmente um objeto dentre um conjunto corrente deles (o sistema necessita de um supervisor que lhe informe quando uma nova instância de objeto ocorrer). Em (KOHONEN, 1990), o uso de mapas auto-organizáveis ("Self-Organizing Maps" ou SOM) que exploram o espaço de características através de agrupamentos ("clustering"), provê um excelente modelo para categorização a partir de mapas de características. Nossa opção pela rede BP baseou-se no fato de que ela produz resultados aceitáveis em tarefas de identificação e também porque ela retorna um valor da ativação para todos os nós da camada de saída, um valor normalizado entre 0 e 1. Este valor de ativação será usado para determinar se uma representação é desconhecida ou para guiar a atenção. Tarefas atencionais do tipo top-down podem usar este valor para manter a atenção numa mesma área.

3.4 Mapas Espaciais

Neste trabalho não é definido um mapa espacial global específico, no qual os objetos descobertos vão sendo postados. Os próprios mapas atencionais (ou mapas de saliência), compostos de um mapa visual para cada olho e de dois mapas hápticos para cada braço, são usados para fins de mapeamento. Assim, mapear uma representação traduz-se mais especificamente em colocar o valor do seu "status de mapeamento" nos mapas atencionais para zero. Isto informa que uma região está efetivamente mapeada, ou seja, que o sistema já visitou aquela região. Além dos valores de ativação de cada região de interesse informarem se a região já se encontra mapeada, os mapas atencionais contêm também informação necessária ao processo de pré-atenção. Um conjunto de características atencionais fica armazenado nestes mapas, que seja suficiente para definir se houve alguma mudança na região, desde a última vez que ela foi visitada.

Nós usamos mapas discretos em coordenadas angulares para representar os mapas atencionais visuais e hápticos. Uma região de interesse é representada em coordenadas do espaço de configurações dos olhos ou dos braços. Note que coordenadas angulares são uma escolha natural para codificação do espaço em um sistema de percepção ativa. Além de as coordenadas angulares poderem ser mapeadas diretamente em torque e velocidade (coordenadas de motor) que são usados pelos controladores posicionais derivativos (GRUPEN, 1999), elas também suportam facilmente o controle do planejamento de movimento (CONNOLY & GRUPEN, 1993).

Os mapas visuais e hápticos usados serão melhor detalhados no próximo capítulo (4), que trata das plataformas de simulação e de hardware.

Capítulo 4

Implementações da Arquitetura

Neste Capítulo discutiremos detalhes das implementações da arquitetura apresentada no Capítulo anterior, realizadas em duas plataformas diferentes. A primeira plataforma, discutida na seção seguinte, é um simulador ("Roger the Crab") desenvolvido paralelamente à arquitetura, composto por duas câmeras e dois braços robóticos. A segunda implementação foi realizada em uma plataforma de hardware composta por uma cabeça estéreo com duas câmeras conectada a um dispositivo processador de imagens dedicado.

4.1 A Plataforma de Simulação

Para validar computacionalmente a arquitetura apresentada no Capítulo anterior, nós desenvolvemos uma plataforma de simulação robótica e essa arquitetura foi implementada inicialmente nesta plataforma. O robô simulado "Roger-the-Crab", cuja interface gráfica é apresentada na Figura 4.1, possui 5 controladores robóticos que são usados para controlar os movimentos do pescoço (pan), de dois olhos (em particular no que se refere a vergência) e de dois bracos, tudo integrado numa única plataforma.

A Figura 4.2 mostra a dinâmica do simulador. Cada braço possui dois graus de liberdade, ou duas juntas, sendo uma equivalente ao ombro e a outra ao cotovelo. Cada um destes graus de liberdade permite um movimento angular discretizado de 256 unidades polares (ou "counts") num plano, cobrindo uma circunferência completa. Este espaço (ou mapa) de conFigurações permite ao Roger alcançar um objeto que esteja a uma distância d a partir da origem de cada braço, tal que $L_1 - L_2 < d < L_1 + L_2$, sendo L_1 e L_2 ($L_1 \geq L_2$) os comprimento das ligações correspondentes ao ante-braço e ao braço respectivamente. Isto define uma coroa círcular cuja área é discretizada num espaço de 256^2 células. Note que todas as células desta coroa, exceto as de suas bordas, podem ser alcançadas com as ligações em duas conFigurações diferentes. A cabeça de Roger possui 3 graus de liberdade, sendo estes relativos aos movimentos de pan e aos movimentos de vergência

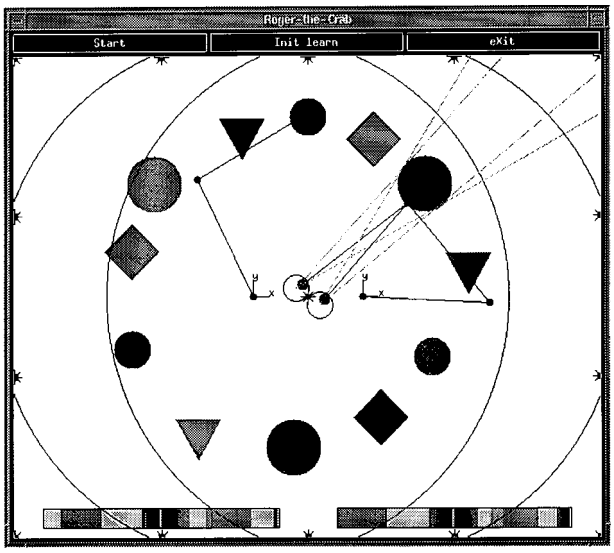


Figura 4.1: Interface do "Roger-the-Crab". A parte visual é composta de uma cabeça com duas câmeras, possuindo tres graus de liberdade, ou seja, movimentos de pan e de vergência esquerdo e direito (sem movimento de tilt). A parte háptica é provida por dois braços, com dois graus de liberdade, ou seja, com duas ligações (ou "links") cada um.

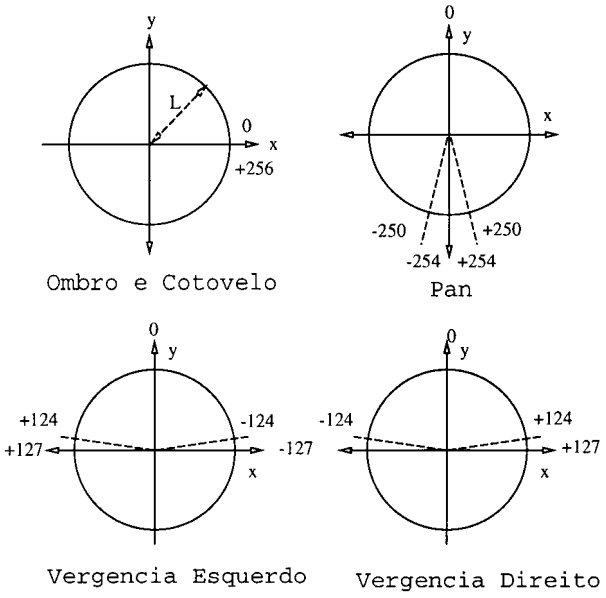


Figura 4.2: Dinâmica do simulador Roger. São mostrados os espaços de conFigurações de cada grau de liberdade e também os limites para estes determinados pelo hardware (linhas sólidas) e pelo software (linhas pontilhadas).

de cada câmera (direito e esquerdo). O espaço de conFigurações do movimento de pan cobre uma circunferência completa e está discretizado em 509 conFigurações possíveis. Como pode ser visto na Figura 4.2, Roger pode mover o seu pescoço (pan) meia circunferência para cada lado, a partir da origem. Movimentos para a esquerda correspondem a valores negativos entre 0 e -254. Movimentos para a direita correspondem a valores positivos entre 0 e +254. O espaço de conFigurações do movimento de vergência é discretizado em 255 unidades polares e cobre no total meia circunferência. Isto permite ao Roger mover os olhos no máximo 1/4 de circunferência para cada lado, sendo -127 referente a vergência interna ou mínima (olhos convergindo um para o outro) e +127 referente a vergência externa ou máxima (olhos divergindo um do outro). Além desses limites estabelecidos em hardware, os controladores de movimento do Roger também possuem limites para os movimentos, determinados por software. Estes limites estão representados na Figura 4.2 pelas linhas pontilhadas e servem para evitar colisões no limite de hardware simulado (numa situação real isto poderia causar danos).

A interface mostrada na Figura 4.1 permite construir e modificar o ambiente do Roger, incluindo fontes de luz, objetos de diferentes tipos e tamanhos, e paredes. Os objetos podem ser círculos, triângulos, quadrados, elipses e outros polígonos regulares. Propriedades naturais dos objetos tais como intensidade de cor (ou nível de cinza), peso, e tamanho podem ser também especificadas e manipuladas. Baseado nestas propriedades, serão realizados os cálculos necessários para simular os dados que os sensores do Roger captam. O processo de iluminação implementado usa um modelo de Phong (ROGERS, 1985; ROGERS & ADAMS, 1990) com até 16 fontes pontuais de luz artificial internas ao ambiente e/ou uma fonte de luz solar para simular a distribuição de luz na retina de Roger. (Veja (NEWMAN & SPROULL, 1979; ROGERS, 1985; ROGERS & ADAMS, 1990; FOLEY *et al.* , 1990) para uma melhor descrição sobre modelos de iluminação). Em adição, este processo pode inserir também um ruído gaussiano no cálculo da intensidade em cada pixel na retina. Esta intensidade é calculada tendo por base o valor especificado para a região da superfície correspondente no ambiente. A aplicação à imagem na retina de filtros que aproximam Laplacianos de Gaussianas ($L \circ G$) com 3 variâncias diferentes constituem o processamento de sinal de baixo nível. Veja (PAPOULIS, 1991) para detalhes sobre distribuições gaussianas. O núcleo desses filtros representa uma discretização da função dada por:

$$\nabla^2 G(r) = (\frac{r^2}{\sigma^2} - 1)\frac{e^{-\frac{r^2}{2\sigma^2}}}{\sigma^3\sqrt{2\pi}} \tag{4.1}$$

Compensações para a gravidade e outros efeitos ambientais são introduzidas e as Equações básicas de dinâmica e cinemática (e suas inversas) são definidas (GRU-

PEN, 1999) para serem usadas pelos servos dos controladores de movimento dos braços e olhos. Para simular o peso de um objeto, utiliza-se uma função que mapeia sua massa, um valor arbitrário atribuído quando da construção do ambiente, no torque e velocidade angular de cada junta (a informação proprioceptiva) necessários para que o braço possa levantar o objeto. Considera-se ainda a conFiguração e a massa das ligações (ou partes) componentes do braço.

Os controladores de movimento (cujos graus de liberdade são especificados na Figura 4.2) são independentes e operam concorrentemente para cada olho e para cada braço. Coordenar todos os controladores consiste em definir basicamente quais estarão operando a um dado momento, de acordo com uma política pré-definida. Os mecanismos de Roger que são responsáveis pela atualização da informação sensorial corrente operam de forma transparente, em baixo nível, transformando a informação de entrada em informação organizada no buffer perceptual, de modo que este último possa representar de forma eficiente uma percepção atualizada do ambiente. Operando com um relógio ("clock") simulado, estes mecanismos mapeiam a geometria externa do ambiente em informação visual, proprioceptiva e tátil. Após a convergência de um conjunto de controladores de movimento, procedimentos de alto-nível podem planejar e promover mudanças no espaço de estados do sistema, atualizar um mapa do ambiente, decidir quais as mudanças que serão necessárias no próximo ciclo de controle e que ações executar, de acordo com a tarefa que esteja sendo executada, a qual referir-nos-emos por "a política de controle adotada". Foram derivadas duas políticas de controle para um supervisor que opera em alto-nível. Em ambas as políticas, a função objetivo é a construção e manutenção de um mapa do ambiente. Na primeira política de controle, que será apresentada na subseção 4.1.6, uma estratégia simples e direta foi desenvolvida baseada na arquitetura mostrada na Figura 3.1, descrita no Capítulo anterior. A seguir, uma outra política de controle, que será apresentada na subseção 4.1.7 deste Capítulo, foi derivada usando a técnica de aprendizado "Q-learning". Esta última política de controle baseia-se em um espaço de estados e um conjunto de ações mostrados na máquina de estados finitos apresentada na Figura 4.5, vista adiante. No restante desta seção, nós iremos descrever como a arquitetura funciona na plataforma de simulação, antes das particularidades dessas políticas de controle.

4.1.1 Pré-atenção e Mudança de Atenção

Pré-atenção é um mecanismo de passo único que opera após a convergência dos controladores de movimento dos braços e dos olhos. Sua função é extrair bordas, delimitando regiões de interesse (ROI) e também calcular valores de ativação (involuntários) para cada região de interesse. Estes valores de ativação são funções normalizadas, ou seja, no intervalo $[0, 1]$, das seguintes variáveis: tamanho na re-

tina, média de intensidade, padrão de movimento visual, resposta tátil e status de mapeamento ("mapping") da ROI. Este último é uma variável que informa se a região encontra-se efetivamente mapeada. No caso, o valor inicial desta variável é 1 para todas as ROIs detectadas inicialmente no ambiente pelo processo de segmentação (ainda não visitadas, portanto não mapeadas), sendo ajustado para zero se a ROI recebe a atenção.

O mecanismo de mudança da atenção também opera somente nos estados onde há convergência dos controladores. Ele determina a ROI mais saliente no momento usando os valores de ativação calculados pelo processo de pré-atenção e atualizados a cada tentativa de se estabelecer uma correspondência com um objeto já representado na memória associativa. A seguir, ele muda a janela de atenção corrente para esta ROI vencedora do processo de atenção, o que determina que os controladores de movimento operem até atingir um novo estado de convergência, estabelecido em função dessa mudança. Esta convergência é atingida quando a região de interesse é colocada na fóvea ou seja, no centro das retinas. A decisão de qual a ROI que Roger deve prestar atenção é feita por um mecanismo do tipo "winner-take-all". Note que o foco de atenção é determinado por um dos olhos ou ainda por um dos braços. Doravante, este será referido simplesmente como olho ou braço dominante.

4.1.2 Fazendo os Olhos e os Braços Convergirem numa ROI

Os controladores de movimento da cabeça de Roger, compostos pelo controlador de pan (pescoço) e pelos controladores de vergência de cada olho, operam de modo diferente do que os dos braços. O controlador de vergência do olho dominante toma a janela de atenção corrente e calcula o deslocamento necessário para colocá-la no centro da imagem. Concorrentemente, o olho não-dominante calcula o deslocamento angular necessário para maximizar a correlação entre os centros das imagens. O controlador do pescoço (pan), operando com um ganho menor, tenta manter seu eixo principal no horópter (cruzamento dos eixos óticos), calculando o deslocamento até este. Após cada vez que os controladores de pan e de vergência calculam os próximos movimentos, os servos correspondentes, que são controladores posicionais derivativos (GRUPEN, 1999), atualizam o torque e velocidade angular correntes. Estes últimos seriam transformados em última instância em corrente, acionando os motores responsáveis pelos respectivos movimentos. Já os movimentos dos braços são guiados visualmente. O objetivo posicional é o horópter. Se um dos controladores dos braços deve agir, o planejador de movimento calcula um caminho livre de colisão da posição corrente do braço à posição alvo, baseado na informação corrente contida em dois mapas de conFiguração: um mapa de potencial e um mapa de fronteiras. Este caminho é calculado resolvendo-se uma função harmônica por uma estratégia

de gradiente descendente (GRUPEN, 1999). Basicamente, considerando-se alto o potencial na posição atual e nas posições onde se tenha obstáculos (fornecidas pelo mapa de fronteiras), usa-se uma função de vizinhança conhecida como "Manhatan Neighborhood" para espalhar esses potenciais às outras posições, de baixo potencial, mantendo-se o potencial no objetivo como zero. Ao final de um processo iterativo, em cada posição, o gradiente mínimo determina a direção do melhor caminho (livre de obstáculos). (Ver (CONNOLY & GRUPEN, 1993) para melhores detalhes sobre esta estratégia e sua convergência).

4.1.3 Extração de Características

Intensidade e textura são computadas como média e variância locais, respectivamente, usando diretamente as respostas L_{ij} do filtro $L \circ G$ definido acima pela Equação 4.1. A disparidade estéreo é calculada através de cálculos de correlação obtidos também a partir dessas respostas. Uma vez que são usados núcleos com 3 escalas (σ) diferentes para esses filtros, também se têm 3 níveis (portanto, um vetor 3D) para intensidade, textura e disparidade estéreo. Esta extração de características tem resolução espacial adaptativa, dependendo do tamanho visual do estímulo. Da resposta obtida para um filtro $L \circ G$, um mesmo número de píxels N_i é aproveitado para a extração de características, independente do tamanho visual da região de interesse. Assim, em cada nível $i = 1, 2, 3$ de resolução, a intensidade I_i (ver Equação 4.2) é calculada como uma média local das respostas L_{ij} do filtro $L \circ G_i$ obtidas em N_i pixels, normalizadas por um valor máximo arbitrário L_{Max} previamente calculado. Cada componente do vetor textura T_i (ver Equação 4.3) é calculada como um momento de segunda ordem (uma variância) das respostas L_{ij} do filtro $L \circ G_i$, também normalizadas pelo valor máximo L_{Max}. A representação para a forma (ou disparidade estéreo) também é um vetor variância, dado pela Equação 4.4 abaixo, das medidas de disparidade estéreo S_{ij} mais uma vez normalizadas por um valor máximo S_{Max}. Reconhecemos que esta escolha de parâmetros pode não ser a melhor representação para a forma, mas ela se revelou suficiente para diferenciar um objeto dentre um conjunto razoável deles com características de forma semelhantes. O tamanho D (Equação 4.5) de um objeto também extraído das medidas de disparidade é normalizado entre zero e um tamanho máximo o qual é arbitrado como uma função da geometria das lentes e da câmera. O mesmo ocorre com o peso W (Equação 4.6) extraído a partir de informação proprioceptiva dos braços. Este peso é normalizado pelo peso máximo que um braço consegue levantar.

$$I_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \frac{L_{ij}}{L_{Max}} \tag{4.2}$$

Objeto	Intensidade	Tamanho	Peso
Círculo	99	30	15
escuro	0.83	0.45	0.72
	79	30	10
	0.65	0.43	0.51
	69	30	10
	0.56	0.44	0.47
Círculo	59	30	5
claro	0.47	0.44	0.24

Tabela 4.1: Valores de propriedades arbitrados para os objetos e valores de características normalizadas calculadas pelos processos de simulação do Roger (sensores). Os objetos em questão são todos do mesmo tipo (círculo), com intensidades e peso variando.

$$T_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \left(\frac{L_{ij}}{L_{Max}} - \frac{1}{N_i} \sum_{k=1}^{N_i} \frac{L_{ik}}{L_{Max}} \right)^2 \tag{4.3}$$

$$S_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \left(\frac{S_{ij}}{S_{Max}} - \frac{1}{N_i} \sum_{k=1}^{N_i} \frac{S_{ik}}{S_{Max}} \right)^2 \tag{4.4}$$

$$D = \frac{d}{D_{Max}} \tag{4.5}$$

$$W = \frac{w}{W_{Max}} \tag{4.6}$$

A Tabela 4.1 lista algumas propriedades atribuídas (linhas ímpares) a alguns objetos e também os valores de características normalizados (linhas pares), após a simulação realizada pelos sensores de Roger e os cálculos realizados pelos processos de reconstrução estéreo e outros. Formalmente, os valores das características não representam as propriedades naturais dos objetos. Essas características são obtidas por expressões matemáticas, então não podem ser usadas como medida das propriedades naturais associadas a elas. No modelo biológico, o conceito intuitivo de peso como sensação é invariante, independente da conFiguração do objeto e/ou da pose do agente. Por exemplo, não há como medir o peso a partir da informação proprioceptiva fornecida pelos sensores de Roger se o braço estiver numa pose em que essa não pode ser determinada exatamente. Isto ocorre, por exemplo, para o braço em posição vertical totalmente estendido.

Outros tipos de objetos podem ser vistos na Figura 4.1. Na pose ilustrada, o controlador do braço direito e os controladores dos movimentos de pan e de vergência encontram-se já convergidos. Informação a respeito do objeto em foco (um círculo) está sendo extraída e a correspondência na memória associativa está sendo realizada, para tentar encontrar uma representação já existente similar.

Note que idealmente o horópter deve estar no centro da janela de atenção, para que a tentativa de correspondência se realize. Para promover a vergência para esse ponto, tentamos maximizar uma correlação simples entre as imagens no centro de cada retina. O aprendizado Q-learning pode também ser usado para derivar uma política na qual as ações realizadas pelos controladores que levam a esse estado de vergência são recompensadas (PIATER *et al.* , 1999). A quantidade de cálculos realizados e a resolução espacial das medidas estéreo, em particular, podem ser estabelecidas em função da performance (precisão e tempo de resposta) necessária para realizar uma tarefa. Uma fixação rápida pode permitir que apenas um nível de menor precisão para a disparidade, intensidade e textura seja calculado, enquanto que uma fixação mais demorada pode permitir realizar este cálculo com uma precisão/resolução mais altas.

4.1.4 Estabelecendo a Correspondência das Características

Uma vez que a forma, tamanho, intensidade, textura e o peso são calculados, estes são usados como um padrão de entrada para tentar estabelecer a correspondência com os padrões já existentes na memória. Isto permitirá identificar uma instância de uma representação. O lay-out da rede BP usada neste ambiente de simulação é representado na Figura 4.3. O número de nós da primeira camada é igual ao número de características extraídas, no caso 11: 3 formas, 3 texturas, 3 intensidades, 1 tamanho e 1 peso. O número de nós da segunda e da última camada (da direita) cresce de forma dinâmica, de acordo com o número de objetos conhecidos num dado momento.

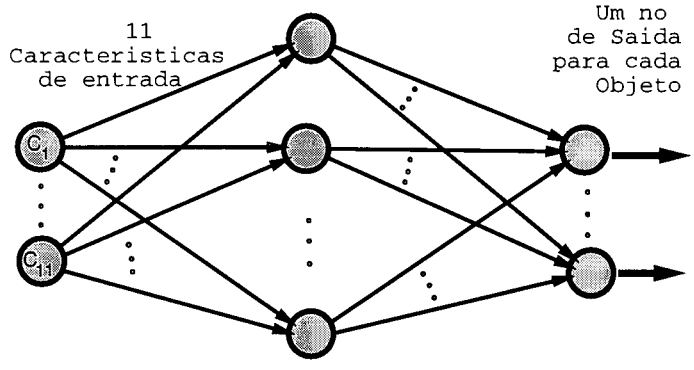


Figura 4.3: Rede neural "back-propagation" usada em simulação.

Se uma identificação positiva não ocorre numa dada tentativa de correspondência na memória associativa porque o maior dos valores de ativação obtidos para um nó na camada de saída da rede BP encontra-se ainda abaixo do limiar dado pela Equação 3.1, um processo em que se tenta melhorar a qualidade das características visuais poderá ser executado. Isto permitirá confirmar ou descartar a representação associada a esse nó (mais ativada). Se esta tentativa de melhora da informação visual também falha, uma outra tentativa é realizada fazendo-se o braço tocar ou mover o objeto para melhorar ainda mais as propriedades extraídas. Se, após a realização dessa última tentativa, o padrão permanece ainda sem identificação, isto significa que um objeto desconhecido está sendo detectado e a memória deve ser incrementada com suas propriedades, desconhecidas até então. Como já visto, isto é feito evocando-se de forma automática o supervisor de aprendizado.

4.1.5 Mapeando Topologicamente uma Representação

Uma vez que as regiões de interesse vão sendo identificadas, um mapa de topologia dos objetos contidos no ambiente pode ser então construído de forma incremental, colocando-se neste mapa um objeto (ou ROI) a cada vez. Esses são os mesmos mapas "atencionais" usados pelo processo de pré-atenção, onde as ROIs encontram-se segmentadas. Este mapeamento topológico é efetivamente completado ajustando-se para zero o valor da variável de estado da ROI corrente, a qual nos referimos anteriormente na seção 4.1.1 como "status de mapeamento". Isto, além de informar ao sistema que uma região já se encontra mapeada, permite ao mecanismo de atenção mudar a janela de atenção para outra região, uma vez que a referida variável não mais contribuirá no processo de decidir qual a região mais saliente (esta última é dada pela ROI cujo o valor do somatório dos valores de ativação seja mais alto). Note que em caso de uma identificação negativa, mas com alguma ativação na memória, o status de mapeamento da ROI corrente pode ainda continuar com valor alto, permitindo que a atenção permaneça na mesma região.

O espaço de conFigurações das câmeras e dos braços do "Roger" é usado para definir um sistema de coordenadas para a construção dos mapas atencionais. Para a parte visual, uma região de interesse é representada em coordenadas polares discretas, como um intervalo inteiro $[a, b]$. O movimento de vergência define o espaço de conFigurações de cada olho, ou seja, 508 unidades perfazendo uma circunferência completa, sobre o qual uma ROI pode estar definida.

Assim, uma ROI nada mais é do que uma estrutura de dados contendo o intervalo espacial o qual ela cobre no ambiente, representado em coordenadas do espaço de conFiguração dos olhos, acrescida de outros dados como os valores de ativação anteriormente definidos (inclusive o dos braços) necessários ao processo de atenção e também as 11 características relativas à região (formas, texturas, intensidades,

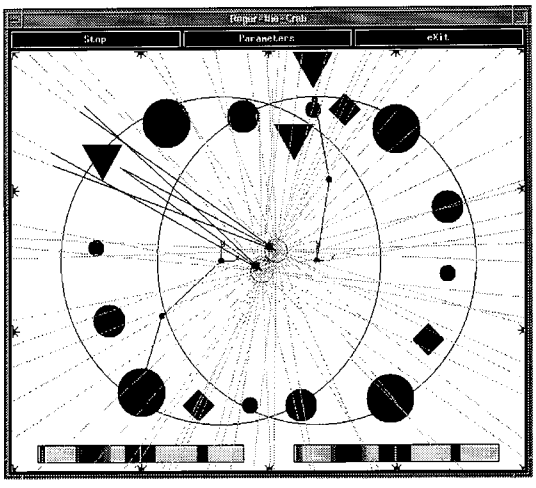


Figura 4.4: Regiões de interesse segmentadas para um dos ambientes.

tamanho e peso). Como citado anteriormente, estas últimas ficam guardadas para possibilitar detectar alguma mudança que possa eventualmente vir a ocorrer no ambiente. Cada ROI contém também ponteiros para as ROIs vizinhas, para que o processo atencional possa percorrer os mapas visuais e definir qual a região mais saliente. Uma segmentação do ambiente em regiões de interesse, detectadas a um dado momento num dos experimentos, pode ser vista na Figura 4.4. Neste experimento, o ambiente já foi totalmente coberto (todas as regiões visitadas). As linhas finas delimitam os cones associados a cada região, representando porções do ambiente percebidas como unidades perceptuais, para cada olho. Para as regiões que se encontram dentro do campo visual de Roger, essa delimitação é efetuada usando-se um detector de arestas sobre o resultado do filtro $L \circ G_3$ (o que opera com resolução mais alta).

Note que fora do campo de vista, as ROIs delimitadas na Figura não reproduzem exatamente a geometria do ambiente, podendo haver distorções. Lembre-se que o espaço de conFigurações sobre o qual as ROIs são definidas é determinado pelo movimento de vergência, portanto a origem de cada cone é posicionada em cada olho no momento em que a ROI é detectada. Ao se realizar um movimento de pan, a posição dos olhos muda, mas não são aplicadas as transformações geométricas inerentes a essa modificação às origens dos cones (isto não é efetivamente necessário aos controladores robóticos, talvez apenas para mostrar visualmente na interface). Note que se Roger olhar novamente para a região à sua retaguarda, esta distorção ou efeito geométrico na interface desaparecerá para este novo lado, e ela ocorrerá para o outro lado.

Usando este conceito de região de interesse fica muito rápido realizar uma checagem no ambiente para determinar em qual região prestar atenção ou então detectar mudanças ocorridas, bastando comparar os dados de cada ROI correntemente de-

tectada com os dados das ROIs que constam dos mapas atencionais.

Do mesmo modo que o espaço de conFigurações dos olhos serve para dimensionar os mapas visuais, uma grade regular 2D representa um espaço de conFigurações discreto para o braço. Assim, os mesmos graus de liberdade descritos na seção 4.1 definem a dimensão dos mapas do braço: 256×256 posições. Como visto na subseção 4.1.2, os dois mapas de cada um dos braços (um mapa de fronteiras contendo todos os objetos ou obstáculos e um mapa de potencial) serão usados pelo planejador de movimentos, para calcular um caminho livre de obstáculos.

4.1.6 Um algoritmo Simples para Controle

Para testar inicialmente a funcionalidade da arquitetura, nós implementamos uma estratégia de controle simples e direta para controlá-la, tendo por base os aspectos funcionais descritos na Figura 3.1. No algoritmo apresentado a seguir, assumimos que o mecanismo de pré-atenção opera imediatamente após cada movimento (convergência dos controladores) e também que a extração de características e tentativa de se estabelecer uma correspondência na memória associativa são automaticamente executadas após uma mudança no estado perceptual (ou convergência dos controladores de movimento). Inicialmente, a memória associativa contém representação para dois padrões: um com todas as características em 0 e outro com todas as características em 1.

Passo zero ou de inicialização: inicializar os pesos da rede BP (memória associativa), a função que estabelece a correspondência, e os controladores concorrentes dos braços, do movimento de pan e dos movimentos de vergência.

Ciclo de Controle:

1. *Redirecionar a atenção; o mecanismo de atenção poderá mover a janela de atenção corrente ou mantê-la na mesma região.*
2. *Se a correspondência na memória associativa determinar uma ativação para uma representação, acima do limiar, para a ROI corrente, atualizar o mapa atencional colocando o seu valor de ativação "status de mapeamento" em zero e retornar ao passo* 1*; caso contrário tentar uma melhoria das informações visuais.*
3. *Se após esta tentativa de melhoria, uma representação é ativada acima do limiar, atualizar o mapa atencional e retornar ao passo* 1*; Caso contrário, tentar uma melhoria usando o braço.*
4. *Se uma representação for ativada após esta tentativa do braço, atualizar o mapa atencional e retornar ao passo* 1*; caso contrário, acionar o aprendizado supervisionado para armazenar o novo conjunto de características na MLT e retreinar a rede BP; após isto, atualizar o mapa atencional e retornar ao passo* 1.

Observe que um movimento do braço imediatamente após o redirecionamento da atenção pode ser mais eficiente que a tentativa de melhoria das informações visuais para que uma identificação positiva possa ocorrer.

4.1.7 Uma Política de Controle usando Q-learning

Além da estratégia simples de controle detalhada na seção anterior (4.1.6), nós derivamos uma política para controle da atenção usando Q-learning. Consideramos as tarefas de mudança de atenção e categorização como sub-tarefas de uma tarefa de inspeção, definida como um processo markoviano (MDP), cuja máquina de estados finitos (FSM) é mostrada na Figura 4.5. Então, uma solução pode ser buscada para este MDP usando Q-learning (SUTTON & BARTO, 1998). Nessa Figura, cada estado é caracterizado por uma sequência de bits, cada um relativo a uma propriedade da ROI corrente, as quais denominamos *novelty*, *identity* e *mapping*. *Novelty* e *identity* são colocadas em 0 logo após uma mudança de atenção, antes de uma tentativa de encontrar correspondência na memória associativa. *Identity* é colocada em 1 no caso de uma correspondência ser detectada na memória e *novelty* é colocada em 1 no caso da não ocorrer essa correspondência após todas as tentativas possíveis. Neste caso, um *novo* objeto (ou com características desconhecidas até então) é encontrado. Logo após a mudança do foco de atenção, não dá para afirmar se uma ROI é nova ou se já possui uma representação na memória. Isto justifica porque *identity* e *novelty* são colocadas em 0. De forma mais precisa, após uma tentativa de correspondência, três casos possíveis podem ocorrer:

1. A ROI possui uma representação. Neste caso, *identity* é 1 e *novelty* é 0.

2. Uma nova representação é descoberta. Neste caso, *identity* é 0 e *novelty* é 1.

3. Não é possível concluir se a representação é nova ou não. Neste caso, ambas variáveis de estado permanecem em 0. Isto determinará que um movimento de braço ou uma melhoria das informações sejam tentados para conseguir mais e melhores características.

Como as variáveis *identity* e *novelty* não podem ser 1 ao mesmo tempo, o número de estados da FSM reduz-se a 6, como mostrado na Figura 4.5. O último estado *mapping* pode ser conseguido por um simples exame nos mapas atencionais.

O conjunto de ações que determina a transferência de um estado a outro referem-se às ações físicas (ou movimentos) dos olhos e braços e a operação de *mapeamento* (ou *map-updating*) de uma região nos mapas atencionais. Esta última é executada após uma representação ser classificada como nova ou já identificada pela memória associativa. As ações físicas são: uma mudanca completa da janela de atenção (ou

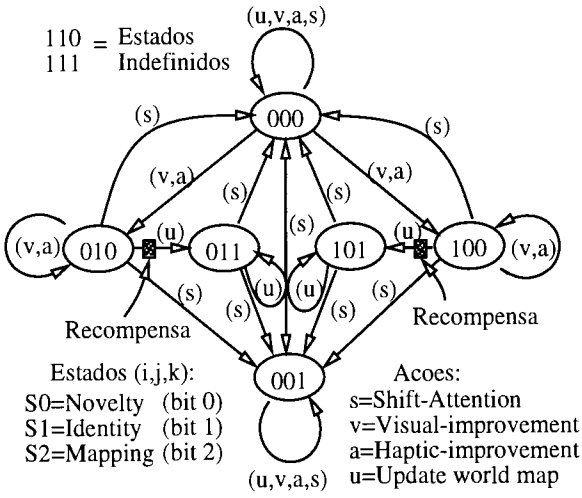


Figura 4.5: Máquina de Estados Finitos (base para o Q-learning)

shift-attention), melhoria da informação visual (ou *visual-improvement*), e melhoria da informação háptica (ou *haptic-improvement*). Um *visual-improvement* pode ser um ajustamento mais fino do processo de vergência ou a procura por alguma característica individual. Num *haptic-improvement*, o braço pode empurrar um objeto para conseguir uma melhor visualização ou pode tentar pegar esse objeto para avaliar o seu peso. No processo de Q-learning, as recompensas são atribuídas às transições entre estados (ações), e não a estados. Isto evita que se obtenha uma política em que o sistema permanece sempre executando ações que levem a um estado com recompensa. Como pode ser visto na Figura 4.5, no modelo adotado neste trabalho as recompensas são dadas à ação de *map-updating*, determinando uma transição do estado 100 para 101 e 010 para 011.

4.2 A Plataforma de Hardware

O robô usado neste trabalho consiste da Cabeça Estéreo mostrada na Figura 4.6. Esta plataforma possui duas câmeras montadas no topo de uma cabeça TRC Bisight, a qual possui os quatro graus de liberdade mostrados na Figura 4.7. Algumas restrições aos movimentos relativos a esses graus de liberdade também são indicadas nessa Figura. As linhas cheias indicam limites do hardware e as linhas pontilhadas indicam limites colocados em software, que se mostraram suficientes para as finalidades a que nos propomos e previnem a ocorrência de colisões que poderiam eventualmente ocorrer se adotássemos os limites de hardware. Comandos para os controladores de movimento podem ser enviados via uma interface de controle PMAC/Delta TAU, a qual controla diretamente os movimentos de pan (rotação lateral ou horizontal do conjunto, semelhante a rotação do pescoço), tilt (inclinação

frontal da cabeça) e vergência para as duas câmeras. Estas últimas também possuem foco e zoom controláveis (ambos são independentes) pelo controlador de movimentos da interface PMAC, não usados neste trabalho.

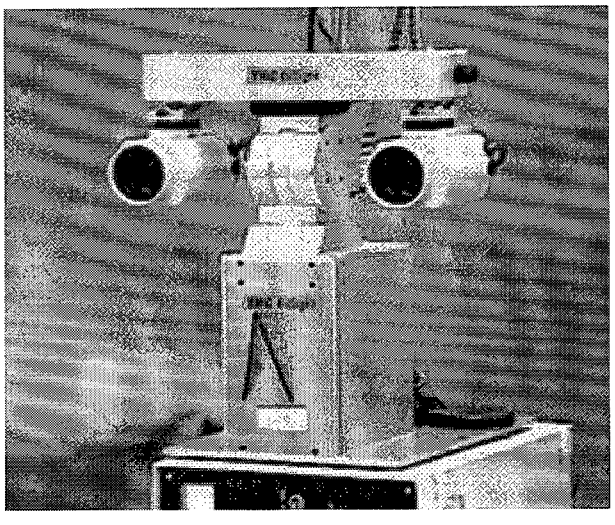


Figura 4.6: Plataforma da Cabeça Estéreo, consistindo de duas câmeras montadas numa cabeça mecânica com 4 graus de liberdade: pan, tilt, vergência direito e vergência esquerdo.

As imagens adquiridas de cada câmera servem como entrada para o processador de imagens dedicado "Datacube". Este consiste de vários dispositivos que implementam em hardware operadores morfológicos e filtros de tipos diferentes, todos integrados numa arquitetura única, que utiliza um modelo de processamento vetorial pipeline. Esta arquitetura permite o processamento das imagens adquiridas pelas duas câmeras em tempo-real a uma taxa de até 30 quadros por segundo. O processador de imagens usa os conceitos de "superfície" de processamento, "pipe" e "PAT", vistos a seguir.

Cada *superfície* é um vetor com valores numéricos inteiros (uma imagem) que pode ser armazenada em um dentre os 6 dispositivos de armazenamento (memórias com capacidade de no máximo 4 Mb cada), para cada uma das duas placas que compõem a atual arquitetura (uma MV-200 e outra MV-250).

Um "pipe" consiste em se tomar uma ou mais superfícies fontes, realizar uma ou mais operação de processamento de imagem, e armazenar a superfície (ou superfícies) resultante no dispositivo de armazenamento indicado. Pode-se ter até 4 pipes operando em paralelo, desde que não compartilhem os mesmos dispositivos de processamento ou caminhos. Um pipe pode operar da seguinte maneira:

- O pipe pode ser de execução contínua, sem sincronização com outros pipes e sem a intervenção do software de controle (Image-Flow). Este último apenas

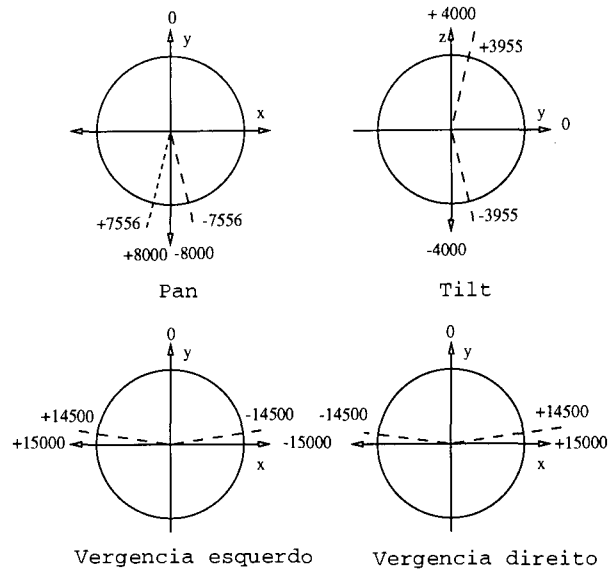


Figura 4.7: Dinâmica da cabeça estéreo. São mostrados os espaços de conFigurações de cada grau de liberdade e também os limites para estes determinados pelo hardware (linhas sólidas) e pelo software (linhas pontilhadas).

inicializa a execução do pipe. Pipes deste tipo geralmente são usados na aquisição ou na visualização de dados, uma vez que, em geral, estas operações não influem nos outros pipes, possuindo caminhos dedicados dentro da arquitetura.

- O pipe pode ser de execução unica, como um pipe "one-shot", engatilhado para ser disparado ou executado de forma sincronizada em toda ocorrência de um evento que pode ser simulado por uma instrução do software Image-Flow ou que pode ser gerado por um outro pipe.

No caso de um pipe ser de execução única, um PAT (Path Alternating Thread) é necessário para sincronizar um ou mais pipes ou para executar pipes que dependam da execução prévia de outros pipes. Neste caso, o pipe em questão só pode ser disparado quando o anterior tiver terminado. O primeiro pipe gera então o evento associado a execução do segundo. Usando PATs, a execução de um pipe pode ser programada para começar após a ocorrência de um evento específico. Assim, a aplicação de controle rodando no computador host pode por exemplo solicitar dados através da simulação de um evento que disparará uma série de PATs.

Um ciclo de controle para a arquitetura inclui uma combinação de pipes de execução contínua e de execução única. Geralmente a finalidade é a transformação dos dados de entrada para prover uma boa abstração (ou compactação de dados). A última operação efetuada no ciclo de controle será eventualmente a transferência desta informação abstrata do "Datacube" para a aplicação de controle que roda no computador host. Esta aplicação decidirá em última instância quais as ações de

alto-nível a executar, envolvendo eventualmente movimentos empregando os graus de liberdade da cabeça estéreo. Alguns exemplos de operações que podem ser executadas dentro da arquitetura "Datacube" são: convolução por filtros genéricos que podem ser definidos pelo programa de aplicação, cálculos de correlação, gradientes de Sobel, transformadas de Hough, operações estatísticas, operações envolvendo duas ou mais imagens como soma, subtração, ou multiplicação, operações envolvendo uma única imagem, como "tresholding" ou outras transformações usando tabelas do tipo "look-up", e ainda extração de certas características da imagem. O dispositivo que realiza a multiplicação e a soma dentro de uma vizinhança (convolutor NMAC) permite a pré-definição de um conjunto de máscaras para seu núcleo com diâmetros de até 8 pixels. Então, um destes núcleos pode ser selecionado e acoplado ao dispositivo NMAC, usando um PAT, em tempo de execução, a uma velocidade muito rápida, permitindo alto desempenho na aplicação de vários operadores às superfícies.

4.2.1 Controladores e a Arquitetura

Uma mesma arquitetura, similar à do sistema multi-modo implementado na plataforma de simulação, foi implementada na plataforma de hardware. Um modelo do tipo "controllers oriented" (ver seção 2.3) é usado aqui para a implementação do sistema de visão ativa.

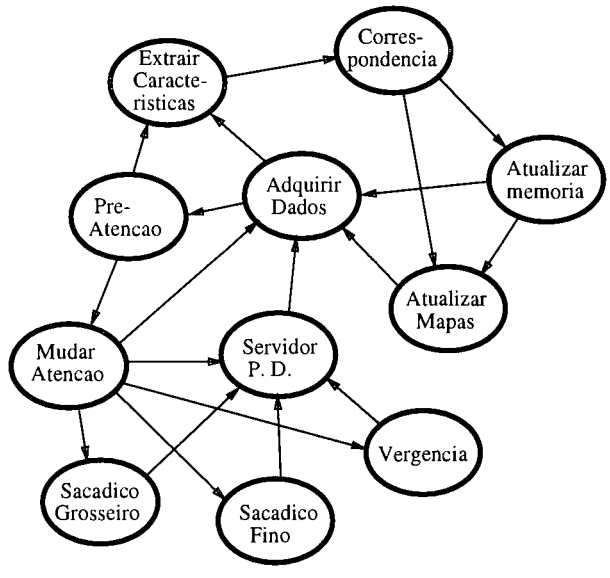


Figura 4.8: Programa Comportamental desenvolvido especificamente para tarefas de atenção e categorização.

Na implementação do sistema na plataforma de hardware, a estratégia que foi utilizada é a constituída pelo programa comportamental representado na Figura 4.8.

Cada controlador é executado automaticamente após a convergência do controlador operando anteriormente. A parte superior e à direita da Figura mostra os controladores relacionados com o processo de categorização (Adquirir dados, Extrair características, Correspondência, Atualizar memória, Atualizar mapas) e a parte inferior e à esquerda estão os controladores relacionados com o processo de atenção, envolvendo os movimentos físicos da cabeça estéreo (Adquirir dados, Pré-atenção, Mudar atenção, Sacádico grosseiro, Sacádico fino, Vergência e Servidor PD).

4.2.2 Retina em Multi-resolução (Buffer Visual)

Uma representação em forma de imagens em multi-resolução, é usada neste trabalho para codificar a informação visual. Um espaço de escalas é necessário para proporcionar uma redução de dados necessária para se obter um processamento em tempo real, enquanto que uma multi-imagem (várias versões de filtros aplicados a uma imagem) é necessária para extração de características diversas que serão usadas pelos processos comportamentais de atenção e categorização.

Um modelo biológico para a geração das imagens em multi-escala pode se basear no uso de derivadas gaussianas com núcleos de diferentes tamanhos (RAO & BALLARD, 1995). Neste caso, pode-se calcular as derivadas diretamente das imagens originais e amostrar os resultados em diferentes resoluções apenas dentro da área de escopo de cada nível. Alternativamente, um filtro com desvio padrão constante poderia ser usado num processo de cascata, calculando-se cada nível a partir do nível anterior, como o é feito em (WESTELIUS, 1995). É mostrado em (WESTIN, 1994) que um desvio gaussiano de $\frac{\pi}{\sqrt{2}}$ é ideal para a geração da pirâmide segundo esta metodologia. Nós arguímos que o mesmo resultado que é obtido usando os modelos acima e com a mesma complexidade computacional podem ser alcançados usando a metodologia adotada no presente trabalho, a qual é descrita a seguir. A redução de dados alcançada deve permitir que o computador possa executar outras operações necessárias para completar o processamento de alto-nível em tempo-real (isto será discutido mais à frente, quando mostrarmos os resultados). Nos experimentos realizados, o processador de imagens "Datacube" pode gerar as 8 imagens em multi-escala a uma taxa de 15 quadros por segundo. Esta taxa obviamente decai quando somado o tempo gasto no processamento de alto-nível, executado pela aplicação de controle que roda no computador host. Mesmo assim, o resultado permite alcançar uma performance compatível com as necessidades do sistemas de visão ativa implementado na cabeça estéreo.

A representação resultante para a nossa retina em multi-resolução pode ser vista na Figura 4.9. Cada uma das 8 imagens em multi-escala possui 4 níveis de resolução diferindo um do outro por um fator de escala 2. Das 8 imagens, 6 são derivadas gaussianas direcionais de intensidade, modificadas para facilitar a implementação e

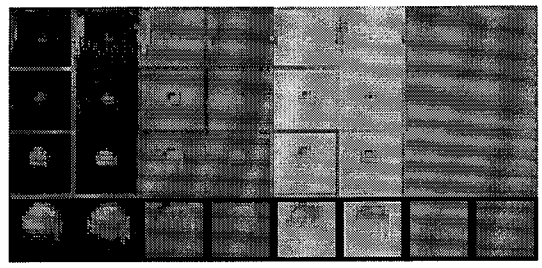


Figura 4.9: Matrizes de características (imagens em multi-escalas) geradas pelo "Datacube" para servir de base aos comportamentos (ou processos) de identificação e de atenção. Cada coluna representa uma imagem em 4 resoluções diferentes. Os três primeiros pares de colunas referem-se à derivadas gaussianas de ordem 0, 1, e 2, respectivamente, dadas pelos filtros definidos pelas Equações 4.7, 4.8 e 4.9, vistas adiante, enquanto que o último par de colunas à direita refere-se às imagens de movimento.

os cálculos. As duas imagens em multi-escala restantes são as derivadas direcionais de primeira ordem das diferenças entre dois quadros consecutivos (nas direções X e Y para cada imagem), representando movimento. Então, ao todo são geradas 8 imagens derivadas parciais em multi-escala, sendo as 6 primeiras derivadas da intensidade (duas de ordem zero, duas de ordem 1 e duas de ordem 2, duas direções para cada ordem), mais duas derivadas de ordem 1 para as imagens representando movimento. Às seis primeiras imagens, chamaremos simplesmente de imagens gaussianas ou parte gaussiana da retina e às duas últimas chamaremos simplesmente de imagens de movimento, embora estas últimas não representem propriamente o movimento, como será discutido mais à frente. Esta transformação de dados se dá em duas fases para ambas as partes (gaussiana e imagens de movimento). A primeira fase é a geração de imagens em multi-escala e a segunda fase é o cálculo das derivadas propriamente.

4.2.3 Geração das Imagens em Multi-escalas

O tamanho das imagens originais capturadas pelas câmeras estéreo é de 512×480 pixels. Estas imagens originais são usadas diretamente para a geração das duas imagens em multi-escala (uma para cada olho) que serão base para o cálculo das imagens gaussianas. Para as imagens de movimento é calculada primeiramente a diferença entre dois quadros consecutivos, um dos quais está sendo adquirido correntemente e outro quadro que é previamente adquirido e mantido armazenado numa das memórias do "Datacube". Cada nível é então gerado, para ambas imagens gaussiana e de movimento, pela aplicação de um filtro média, com núcleos de diferentes tamanhos, na vizinhança de cada píxel daquelas imagens e amostrando-se com um certo fator de resolução. O intervalo de amostragem é determinado em função de ca-

da nível que esteja sendo gerado, bem como também o são o diâmetro da vizinhança e as regiões de escopo das imagens de entrada que serão afetadas pelo processo de filtragem. Para o nível inicial (o de menor resolução), o diâmetro do filtro é de 8 píxels, sendo este aplicado em toda a imagem e amostrado a cada 8×8 pixels. Para o último nível (o de maior resolução), o diâmetro do filtro é de 1 píxel e a região da imagem na qual ele é aplicado é composta pelos 64×60 píxels centrais. Então, neste último nível, uma simples transferência da superfície de origem é executada, sem que se faça uma convolução propriamente dita. A este ponto, o que se têm são quatro imagens em multi-resolução: duas imagens de intensidade e duas imagens de movimento (na realidade estas últimas são imagens diferença), uma imagem de cada tipo para cada olho, onde cada nível tem dimensões de 64×60 píxels.

4.2.4 Computando as Derivadas

Numa segunda fase, são computadas as derivadas parciais para cada nível, para extrair (ou realçar) as características que serão usadas para atenção e categorização. Para gerar a parte gaussiana, cada imagem de intensidade em multi-escala gerada na fase anterior (uma de cada olho) é convoluída com um conjunto de seis núcleos gaussianos (derivadas direcionais da distribuição gaussiana) adaptados, em duas direções cada. As máscaras dos núcleos gaussianos são dadas por:

$$\left.\begin{array}{l} G_x^{(0)}(x,y) = ke^{ax^2} \\ G_y^{(0)}(x,y) = ke^{ay^2} \end{array}\right\} \tag{4.7}$$

$$\left.\begin{array}{l} G_x^{(1)}(x,y) = 2ake^{a(x^2+y^2)}x \\ G_y^{(1)}(x,y) = 2ake^{a(x^2+y^2)}y \end{array}\right\} \tag{4.8}$$

$$\left.\begin{array}{l} G_x^{(2)}(x,y) = 2ake^{a(x^2+y^2)}(2ax^2+1) \\ G_y^{(2)}(x,y) = 2ake^{a(x^2+y^2)}(2ay^2+1) \end{array}\right\} \tag{4.9}$$

$$\forall (x,y) \in [(-s,+s),(-s,+s)];$$

onde $a = \frac{-1}{2\sigma^2}$, $k = \frac{1}{\sigma\sqrt{2\pi}}$, $s = 3$, e $\sigma = 1.7$.

Sendo $d = 0, 1$, representando as duas direções dos filtros (X e Y), e $k = 0, 1, 2$, a Equação que define a convolução realizada para uma imagem I adquirida no instante t é dada por:

$$g_d^{(k)} = G_d^{(k)} * I_t \tag{4.10}$$

Ocorre ainda dentro do "Datacube" uma redução das dimensões das imagens de cada nível que antes era de 64×60 píxels para 16×15 pixels. Para isto, é feita uma amostragem a cada 4×4 píxels. Assim, em cada nível das imagens em multi-escala finais, a área coberta é representada por 16×15 píxels, e de um nível para outro a razão entre as áreas representadas é de 4 vezes. Note que isto permite uma redução de $512 \times 480 = 245760$ píxels na imagem original para uma representação final nas imagens em multi-escalas de $16 \times 15 \times 4 \times 8 = 7680$ píxels (ou seja, uma redução de $32 : 1$). A Figura 4.2.4 mostra uma das imagens em multi-escalas de uma bola de tênis, extraída a partir de uma das colunas da retina mostrada na Figura 4.9. Cada imagem é composta de 16×15 píxels, embora tenham sido ampliadas na Figura para representar as áreas que cobrem na imagem original.

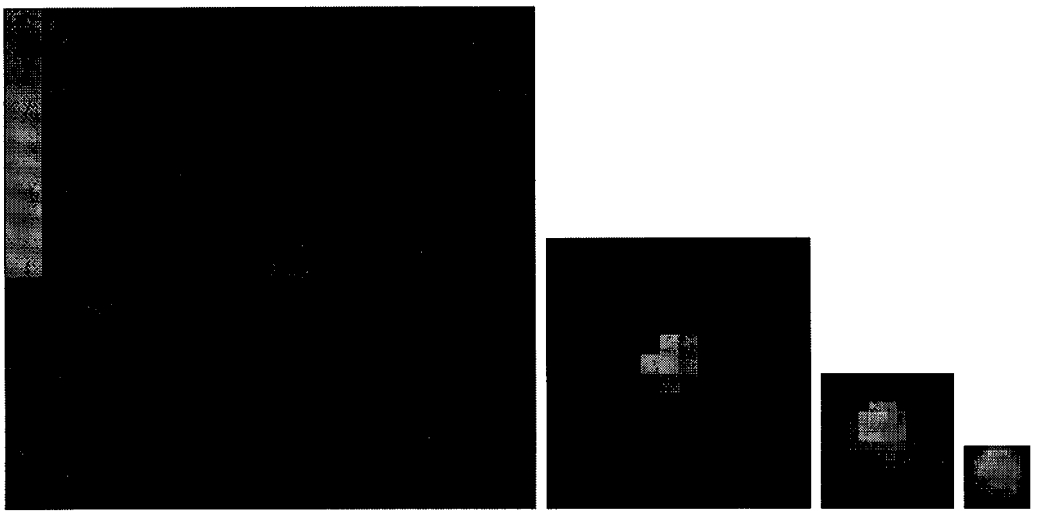


Figura 4.10: Imagem em multi-resolução de uma esfera. Cada imagem é constituída de 16×15 píxels. Quando se passa de uma representação para outra à sua direita, a área coberta diminui 4 vezes.

As derivadas para a parte da retina relativa a movimento são também computadas da mesma maneira. As derivadas primeiras em duas direções são calculadas gerando as imagens finais em multi- escala representando movimento (duas para cada olho). Sendo $d = 0, 1$, representando as duas direções dos filtro de movimento, a Equação que define a convolução realizada para calcular a imagem de movimento, a ser aplicada à diferença entre duas imagens, uma adquirida no instante t e a outra em $t - 1$ é dada por:

$$m_d = G_d^{(1)} * [I_t - I_{t-1}]. \tag{4.11}$$

Note que o mesmo núcleo gaussiano (Equação 4.8) é usado para derivar a parte referente a movimento. Isto ajuda na redução da quantidade de ruídos das imagens.

Um modelo ideal poderia efetivamente calcular o campo de movimento, usando técnicas de relaxação ou outros métodos iterativos. Porém, calcular o campo de movimento não é necessário para a parte de atenção e para a parte de identificação/reconhecimento tal modelo poderia ser muito caro computacionalmente.

4.2.5 Computando a Disparidade Estéreo

Um mapa de disparidade é calculado na memória computador, após a transferência das retinas em multi-escalas. Um modelo simples é usado, maximizando medidas de correlação usando as imagens correspondentes às derivadas de segunda ordem. Um modelo alternativo poderia usar informação de frequência espacial para calcular estéreo diretamente das imagens de entrada, dentro da arquitetura "Datacube", usando o modelo biológico citado anteriormente, como em (WESTELIUS, 1995). A Figura 4.11 mostra uma metodologia em "cascata" usada por nós neste trabalho para computar a disparidade (ver (GONÇALVES & OLIVEIRA, 1998) para um método similar). Uma vez que usamos imagens em multi-escala, os resultados de um nível são usados para estimar a disparidade no próximo nível. Para o nível inicial, como o movimento de vergência possui restrições (a serem vistas na subseção 4.2.9), a disparidade é também limitada. Este processo de cascata proporciona uma redução substancial na quantidade de cálculos necessários para estabelecer pontos correspondentes nas imagens. Outra restrição usada é dada pela simetria relativa entre as imagens com respeito ao eixo ciclopiano. Este último é dado pela reta definida pelo ponto central entre os olhos (câmeras) e pelo horopter, o ponto em que os eixos óticos se interceptam. Uma vez que o nosso sistema possui uma geometria controlada (sem disparidade em y), são realizadas medidas estéreo apenas na direção X.

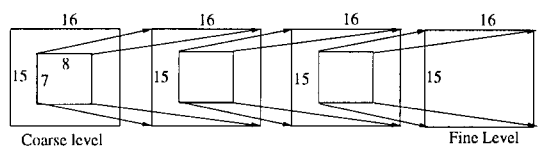


Figura 4.11: Cálculo do processo estéreo em cascata. Cada nível provê uma estimativa da disparidade para o próximo.

4.2.6 Controle do Comportamento de Atenção

Basicamente, espera-se que o comportamento de atenção escolha a região mais saliente no ambiente e mova o foco (ou janela) de atenção para esta região. Isto envolve o cômputo de dois mapas de saliências, um para cada olho (câmera), onde as posições possuem um nível ou valor de ativação baseado na informação fornecida

pelo estímulo visual e também num fator de atenção determinado por processos do sistema, que representa a quantidade de atenção ainda necessária a uma região para que a tarefa seja executada. Então, movimentos sacádicos podem ser calculados e executados pelos controladores de movimento da cabeça estéreo para colocar a região vencedora (mais ativada) no centro da fóvea, por uma simples inspeção nesses mapas. Eventualmente, movimentos de pan e tilt podem ser necessários além dos movimentos de vergência. Os mapas de saliências são calculados usando-se os dados abstratos fornecidos pelo "Datacube" (que são os estímulos visuais ou sinais perceptuais) e também verificando a informação contida num mapa atencional do ambiente em construção (fator de atenção). Apenas os estímulos visuais poderiam ser considerados para definir a próxima região de interesse. Porém, neste caso não há uma garantia de que o sistema cobrirá todo o ambiente. As localizações efetivamente observadas devem estar representadas nesse mapa atencional com um valor de ativação baixo, para que não se tornem novamente o foco de atenção, informando ao sistema que elas já foram previamente visitadas. Este mapa atencional também codifica outro tipo de informação prévia, uma vez que uma checagem rápida é necessária para detectar mudanças eventualmente ocorridas no ambiente.

4.2.7 Definindo um Objetivo (Pré-atenção)

A geração dos mapas de saliências se dá numa fase de pré-atenção. Este mapa, tal qual as retinas também possui uma representação em multi-resolução. Começando da escala de menor resolução, um valor de ativação é calculado para cada posição por intermédio de uma função de pesos normalizada. Esta função transfere ativação a partir do estímulo visual e do mapa atencional para o mapa de saliências. Esta função é dependente da tarefa e pode ser aprendida usando uma rede neural como em (VAN DER LAAR *et al.* , 1995; VAN DER LAAR *et al.* , 1997) ou usando aprendizado de reforço ("Q-learning") como em (GONÇALVES *et al.* , 1999b; GONÇALVES *et al.* , 1999a).

As características consideradas para atenção são o vetor calculado de disparidade estéreo (D_{ij}), os três vetores de intensidade (quadrado da magnitude) $I_{ij}^{(0)}$, $I_{ij}^{(1)}$, e $I_{ij}^{(2)}$ de cada par de imagens gaussianas $g_d^{(k)}$, dados respectivamente pelas Equações 4.13, 4.14, e 4.15 e também o vetor intensidade das imagens movimento M_{ij} dado pela Equação 4.12. No caso desta implementação em hardware, os pesos w_M, $w_{G^{(0)}}$, $w_{G^{(1)}}$, e $w_{G^{(2)}}$ foram definidos experimentalmente, em função da tarefa de inspeção ou monitoração desejada para o sistema, e não usando aprendizado automático ou "Q-learning".

$$M_{ij} = w_M \left[(m_{x,ij}^{(1)})^2 + (m_{y,ij}^{(1)})^2 \right] \tag{4.12}$$

$$I_{ij}^{(0)} = w_{G^{(0)}} \left[(g_{x,ij}^{(0)})^2 + (g_{y,ij}^{(0)})^2 \right] \tag{4.13}$$

$$I_{ij}^{(1)} = w_{G^{(1)}} \left[(g_{x,ij}^{(1)})^2 + (g_{y,ij}^{(1)})^2 \right] \tag{4.14}$$

$$I_{ij}^{(2)} = w_{G^{(2)}} \left[(g_{x,ij}^{(2)})^2 + (g_{y,ij}^{(2)})^2 \right] \tag{4.15}$$

Após estas características atencionais terem sido calculadas, a Equação a seguir é usada para computar o mapa de saliências:

$$S_{ij} = E_{ij} + P_{ij} + D_{ij} + M_{ij} + I_{ij}^{(0)} + I_{ij}^{(1)} + I_{ij}^{(2)}. \tag{4.16}$$

Essa Equação (4.16) é uma simples soma dos valores de ativação dados acima, uma vez que os pesos w_M, $w_{G^{(0)}}$, $w_{G^{(1)}}$, e $w_{G^{(2)}}$ são aplicados previamente no cálculo das intensidades. Note que aparecem dois fatores novos na Equação 4.16: E_{ij} e P_{ij} que são explicados a seguir. O fator E_{ij} é a variável *interesse* como definida na seção 3.2.3, recuperada a partir do procedimento de pré-atenção (mapa atencional). Ao fator P_{ij} que também aparece em 4.16 chamamos de *proximidade*, sendo que o seu valor é inversamente proporcional à distância entre a posição considerada no mapa de saliências e a região da fóvea. Sua finalidade é evitar a geração de movimentos sacádicos muito largos. Ele prioriza a região da fóvea, uma vez que seu valor é maior para esta.

4.2.8 Mudando a Janela de Atenção (Sacádico Grosseiro).

Mudar o foco de atenção envolve tomar a região mais ativa entre todos os níveis nos mapas de saliência e efetivamente direcionar os olhos (câmeras) para o centro desta região. Um movimento sacádico (grosseiro) é calculado para ambos os olhos. Uma vez que temos um mapa de saliências para cada olho, o conceito de olho dominante é aplicado aqui como sendo aquele cujo mapa de saliências contém a região mais ativa. Assim, para o olho dominante, o deslocamento é simplesmente dado pela diferença de coordenadas na imagem entre a posição corrente e a posição vencedora. Para o olho não dominante, a disparidade estéreo é somada ao deslocamento calculado para o olho dominante. Uma vez que ambos objetivos são definidos (ou o deslocamento necessário para se chegar a estes), adquire-se um modelo do objetivo representado por um conjunto de características, para o olho dominante. Essas características são tomadas em uma janela ao redor do objetivo para todos os níveis de resolução. Esse modelo pode ser dado pelo padrão perceptual correntemente na retina (se o objetivo estiver dentro do campo visual) ou pode ser extraído dos mapas atencionais, para objetivos fora do campo visual. Uma vez que poderão eventualmente ocorrer erros

na geração do movimento sacádico grosseiro, esse modelo extraído poderá ser usado para executar correções, dando origem ao que denominamos movimentos sacádicos finos, descritos na próxima subseção.

Completando a geração do movimento sacádico grosseiro, os deslocamentos a serem aplicados aos graus de liberdade da cabeça estéreo são determinados a partir dos deslocamentos de imagem calculados para os objetivos. Os primeiros deslocamentos encontram-se determinados em coordenadas de imagem (na realidade um sistema de coordenadas em multi-níveis, centradas nos olhos). Assim, esses deslocamentos devem ser transformados em unidades de pan, tilt e vergência, conforme especificados na Figura 4.7, valores os quais serão enviados aos controladores de movimento (servos) da interface PMAC/Delta Tau. Esta transformação é uma função dependente também do nível (n) que determinou a atenção e algumas restrições são colocadas aos graus de liberdade da cabeça estéreo. O ângulo ciclopiano, definido como o ângulo entre o eixo perpendicular ao ponto central do segmento de reta que une os olhos (ou limbo) e o eixo ciclopiano (como definido anteriormente) não deve ser maior que um limite que foi determinado experimentalmente como sendo de 15 graus. Sendo $P(d_1, d_2, n)$ uma função como descrita acima, que transforma os deslocamentos d_1 e d_2 calculados em unidades de coordenadas de imagem no nível n para unidades do movimento de pan, este como especificado na Figura 4.7, este limite é dado simplesmente por:

$$|P(d_1, d_2, n)| < 667. \tag{4.17}$$

Dentro do limite acima, as características providas por ambos os olhos ainda produzem bons resultados em tarefas de categorização. Assim, quando o oposto ocorre (ângulo ciclopiano maior que 15 graus ou 667 "counts"), um movimento de pan (como um movimento horizontal do pescoço) deve ser realizado para colocar o ângulo ciclopiano dentro daquele limite novamente.

Outra restrição imposta é que o ângulo entre os eixos de vergência seja também limitado. A posição de abertura máxima (vergência externa) é aquela em que os eixos são paralelos e a de máximo fechamento (vergência interna) é determinada por um ângulo interno de -45 graus em relação um ao outro. Neste caso, para satisfazer esta restrição, uma correção é aplicada ao olho não dominante. Sendo $V(d_i, n)$, $i = 0, 1$, uma função que transforma o deslocamento expresso em coordenadas de imagem no nível n para unidades do movimento de vergência, como especificado na Figura 4.7, a restrição imposta a este grau de liberdade é dada por:

$$-7500 \leq [V(d_0, n) + V(d_1, n)] \leq 0 \tag{4.18}$$

O movimento de Tilt é computado como uma transformação direta do deslocamento vertical na imagem determinado para o olho dominante no nível n escolhido pelo processo de atenção. Como a cabeça estéreo possui câmeras com uma geometria controlada, ambos os olhos possuem o mesmo tilt. A única restrição, que também se aplica aos outros movimentos calculados, é que devem ser obedecidos os limites em software, mostrados com linhas tracejadas na Figura 4.7.

Uma vez que as novas posições estejam calculadas em unidades de movimento dos controladores, para todos os graus de liberdade, estas são passadas à interface PMAC. Nesta, os controladores de movimento (servos PD) são acionados de forma concorrente, direcionando efetivemente a plataforma de hardware para atender a nova posição.

4.2.9 Ajustando Atenção (Sacádico Fino e Vergência)

Devido a erros, após um movimento sacádico grosseiro, os olhos podem não estar vergidos na posição do objetivo especificado, embora em geral estejam muito próximos dele. Então, movimentos sacádicos finos são gerados por um processo iterativo, que visa maximizar a correlação entre o centro da imagem do olho dominante e o modelo da região do objetivo adquirido antes do movimento sacádico grosseiro. Caso a posição que determine o maior valor de correlação não esteja a uma certa distância (um limiar) do centro da imagem, novos movimentos são calculados para o hardware, corrigindo este erro. Este processo iterativo vai do nível de menor para o de maior resolução e converge quando no nível no qual foi determinada uma mudança de atenção atende a restrição acima. Ao mesmo tempo, ou seja, a cada ciclo de controle, em que um sacádico fino é gerado para o olho dominante, o mecanismo de vergência atua sobre o olho não dominante. São calculados deslocamentos para o eixo de vergência desse olho, que procuram trazê-lo a uma posição em que os centros das imagens dos dois olhos possua um valor máximo de correlação. Um limiar é usado para evitar situações em que não há correspondência dentro do campo de vista. Deste modo, os olhos vergem ao mesmo tempo ou imediatamente após a convergência do processo de sacádico fino.

4.2.10 Identificação

Uma vez que ambos os olhos (ou um deles, em caso de oclusão) tenham convergido para uma região de interesse, a identificação ou categorização do padrão perceptual pode ocorrer. Da mesma forma que em simulação, esta identificação é determinada usando-se uma memória associativa implementada através de uma rede neural do tipo "back-propagation" (BP). Esta rede BP, cuja estrutura é mostrada na Figura 4.12, também possui três camadas ou níveis. A memória associativa relaciona

as características extraídas das imagens fornecidas pelo "Datacube" a uma representação armazenada na memória de longo termo (MLT). Características obtidas de um nível arbitrário das retinas podem ser usadas para estabelecer esta correspondência. Em uma situação mais geral, o nível de resolução no qual as características devem ser calculadas depende da tarefa sendo executada (por exemplo, se a atenção é top-down ou bottom-up), do tempo disponível e das próprias imagens. Como vimos, é uma função do mecanismo atencional definir qual nível será usado.

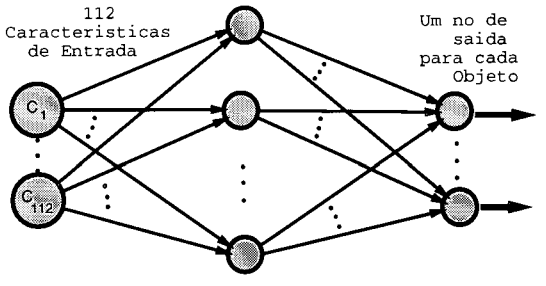


Figura 4.12: Rede neural de "back-propagation" usada na aplicação para a plataforma de hardware. O número de nós da segunda e da última camada (da direita) cresce de forma dinâmica.

Na camada de entrada, a rede neural usada possui um nó para cada característica. Na camada de saída, inicialmente com dois nós, um novo nó é criado para cada nova representação detectada. Uma função ponderada dos erros mínimo e máximo determinados pelo processo de treinamento da rede neural é usada como limiar para definir se uma representação é nova (Equação 3.1). A camada intermediária ("hidden") possui um número de nós determinado empiricamente. Nos experimentos que realizamos, fazendo esse número igual a 1.5 vezes o número de nós da última camada, bons resultados foram obtidos. O treinamento é feito da mesma forma que a descrita na seção 3.3, e uma vez treinada, o cálculo da ativação para uma representação se dá também de forma semelhante, notando que o número de nós na camada de entrada é diferente, como será explicado a seguir.

4.2.11 Extração de Características

Experimentos usando as imagens das retinas em multi-resolução diretamente como entrada para a memória associativa deram bons resultados. Nestes experimentos, foram usadas as 8 matrizes ($2G_0 + 2G_1 + 2G_2 + 1Motion + 1Estereo$) de um mesmo nível (determinado pela atenção), representando as características das imagens obtidas para cada olho. Como cada uma possui dimensões de 16×15, esta entrada para a BPNN, composta por um total de 3840 valores, resultou numa quantidade considerável de processamento. Isso determinou que alguma forma de compactação tivesse que ser realizada. Existem alguns modelos que poderiam resultar numa boa

compactação dos dados de entrada para a BPNN. Por exemplo, o trabalho descrito em (RAVELA & MANMATHA, 1997) usa relações de invariância irredutíveis (FLORACK, 1993) entre as derivadas gaussianas parciais de uma imagem para construir uma consulta em um banco de imagens. Em nosso caso, realizamos uma série de experiências até ao ponto de considerar a informação extraída pelo cálculo de médias (μ) e de variâncias (σ^2) aplicadas a uma vizinhança local, em 4 posições das multi-retinas. As Equações usadas para o cálculo dos momentos são dadas por:

$$\mu_{i,j} = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \frac{G^{(i)}_{i+m,j+n}}{G^{(i)Max}} \tag{4.19}$$

$$\sigma^2_{i,j} = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \left(\frac{G^{(i)}_{i+m,j+n}}{G^{(i)Max}} - \mu ij \right)^2 \tag{4.20}$$

Sendo dx e dy as dimensões da imagem (no caso, $dx = 16$ e $dy = 15$), essas posições são dadas por ($dx/4$,$dy/4$), ($3dx/4$,$dy/4$), ($dx/4$,$3dy/4$) e ($3dx/4$,$3dy/4$). Para a parte gaussiana da retina (6 primeiras colunas), as médias e variâncias são calculadas para cada imagem do nível escolhido pelo processo de pré-atenção (as duas direções são consideradas). Para a parte de movimento, apenas a média é calculada e esta é aplicada não diretamente às imagens de movimento componentes da retina, mas sim à matriz de intensidade (magnitude) destas já calculada na fase de pré-atenção. Acrescentando-se 4 médias das medidas de disparidade estéreo, também obtidas da mesma forma, sobre a matriz de disparidade calculada pelo processo de pré-atenção, obtém-se um total se 112 características de entrada para o processo de identificação, sendo 4 médias de disparidade estéreo, 4 médias de intensidade de movimento, 24 médias das gaussianas e 24 variâncias das gaussianas, para cada olho. Esta quantidade de dados mostrou-se razoável para os nossos propósitos computacionais.

Ao usar as medidas estatísticas (μ e σ^2) dadas pelas Equações 4.19 e 4.20 acima, como características de entradas para a rede BP, uma normalização de contraste é incluída. Também, como a vizinhança é levada em consideração, as respostas consideram características locais de uma forma mais espalhadas, de acordo como a quantidade de energia presente na vizinhança, representando mais do que características locais. Para completar, as características resultantes levam consigo alguma invariância com respeito a rotação, translação e escala. Rotações até 30 graus foram bem suportadas nos experimentos realizados (ver Capítulo 5), Assim, consideramos que esses momentos são semi-invariantes com respeito à rotação.

4.2.12 Mapeamento Topológico e Atualização da Memória

Uma vez que uma representação é classificada como nova ou que ela é identificada, os mapas atencionais são atualizados, informando que a região correspondente já foi visitada. Para isto, o valor da variável de ativação "status de mapeamento" é ajustado para zero, o sendo também o valor da variável *interesse*. Isto também permite uma mudança do foco de atenção para outra região. As características usadas para atenção, já armazenadas anteriormente nos mapas pelo processo de atenção, são suficientes para detectar alguma mudança futura naquela região. Se a representação é nova, a memória associativa é retreinada. Isto envolve acionar o módulo de aprendizado supervisionado, o qual insere o novo conjunto de características na MLT, incrementa a memória associativa criando os nós necessários tanto na última camada quanto na intermediária, e retreina a memória em função dessas novas características inseridas. Note que os mesmos mapas (mapas atencionais) são usados aqui e no processo de atenção. Convém frisar que isto agiliza o processo de atenção e também diminui o custo computacional inerente a construção e manutenção de um mapa topográfico do ambiente contendo modelos geométricos dos objetos. Arguímos que, na prática, para executar uma dada tarefa, um robô não necessita usar modelos geométricos de objetos da cena percebida, mas sim apenas os padões perceptuais dos mesmos. Uma resposta (ou ação) a estes padrões perceptuais será gerada pelo sistema de decisão do robô, em forma de feedback.

Capítulo 5

Experimentos e Resultados

Foram realizados vários experimentos envolvendo atenção e categorização tanto no ambiente de simulação quanto utilizando a cabeça estéreo. Nos testes finais em ambas as plataformas, os dois comportamentos foram integrados numa tarefa única. Várias instâncias de vários tipos de objetos foram colocados em um ambiente restrito e então realizada uma tarefa de inspeção ou monitoração. Em ambas as plataformas, esta tarefa de inspeção foi desempenhada com sucesso. Nas seções seguintes, descrevemos estes experimentos e os resultados alcançados para ambas as plataformas.

5.1 Experimentos e Resultados em Simulação

Na plataforma de simulação, tanto a estratégia de controle simples como a estratégia que usa Q-learning foram aplicadas à tarefa de inspeção com bons resultados. A rede neural atua identificando um objeto positivamente se a ativação calculada a partir das características extraídas desse objeto estiverem acima do limiar dado pela Equação 3.1. Se um objeto não for identificado numa primeira instância usando-se apenas informação visual, os braços se movem para tentar adquirir mais informação. Após todas as tentativas, em caso de identificação negativa (ativação ainda abaixo do limiar), um procedimento de aprendizado supervisionado atualiza a rede neural automaticamente, incluindo o novo padrão encontrado. Após todas as regiões de interesse serem visitadas, *Roger* permanece no estado de monitoração (inspeção) descrito na subseção 3.2.3.

Torna-se difícil estabelecer um critério para medir o desempenho do sistema nas tarefas realizadas experimentalmente em simulação. Isto porque não temos meios de definir uma situação ideal para comparar com os algoritmos. Então, para efeitos de comparação, apresentamos na Figura 5.1 alguns dados obtidos aplicando-se ambas as estratégias (simples e Q-learnng) a um mesmo ambiente, submetido às mesmas condições de luz e com os mesmos objetos. Como pode ser visto nessa Figura, o método usando Q-learning realizou mais mudanças de atenção, menos tentativas de

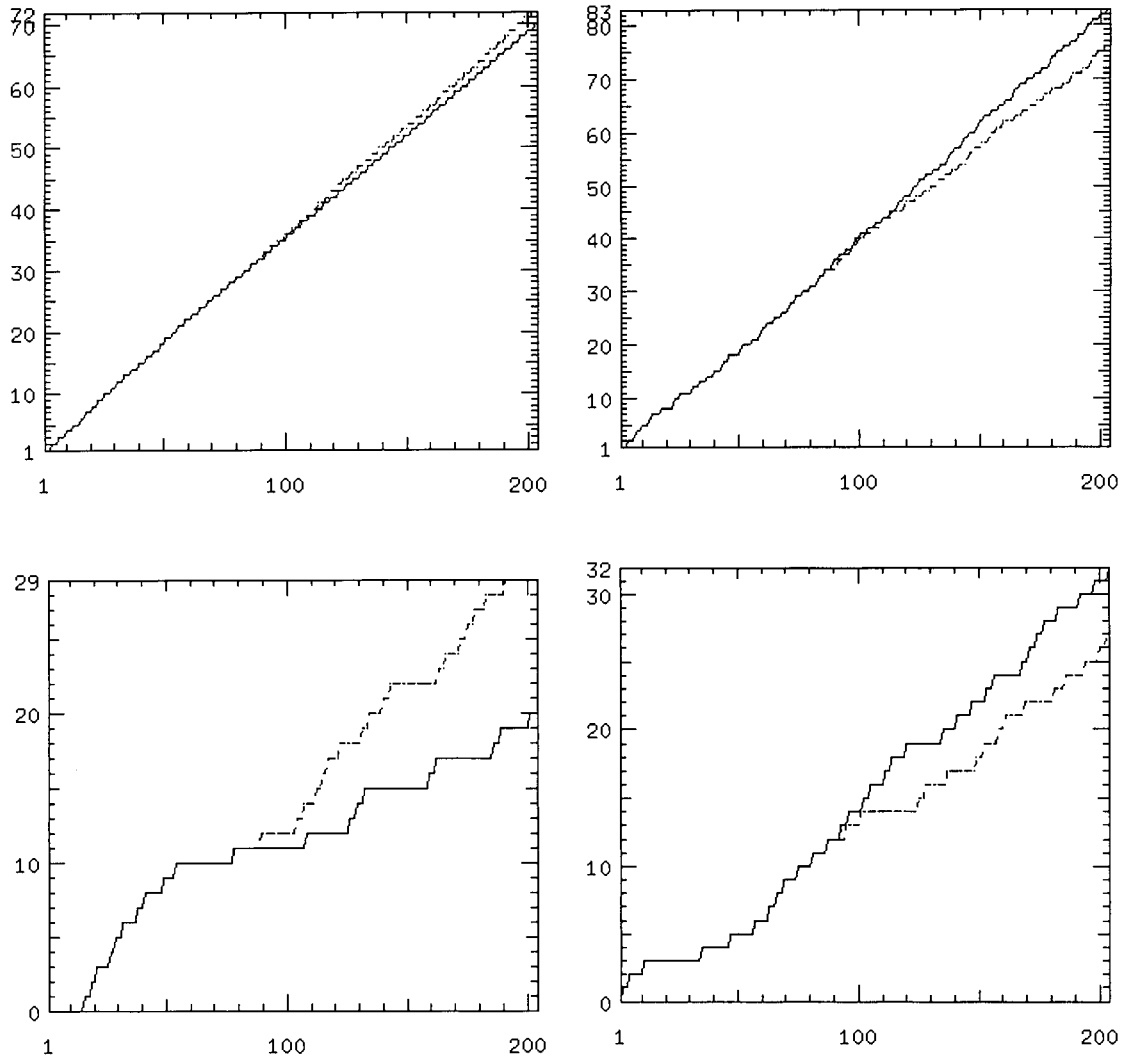


Figura 5.1: Avaliação parcial. O eixo horizontal mostra o número de ciclos de controle operados. O eixo vertical mostra as ações ou fases realizadas como descrito a seguir. Esquerda e acima: número de mudanças no foco de atenção. Direita e acima: número de tentativas de melhora da informação visual/háptica. Esquerda e abaixo: número de identificações positivas. Direita e abaixo: número de novas instâncias de objetos encontradas. Em todos os gráficos, a linha pontilhada refere-se ao método que usa Q-learning e a linha sólida à estratégia simples, descritos anteriormente nas subseções 4.1.7 e 4.1.6, respectivamente.

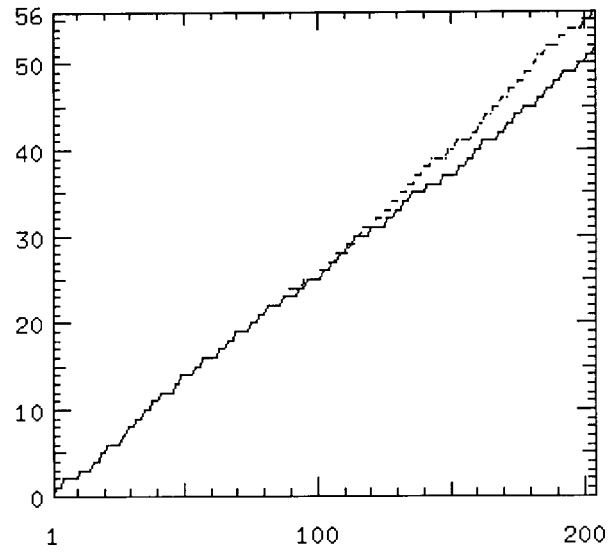


Figura 5.2: Avaliação global. O eixo horizontal mostra o número de ciclos de controle realizados. O eixo vertical mostra o número de objetos (novos ou já conhecidos) mapeados. A linha sólida é para o método usando a estratégia simples e a pontilhada é para o método usando Q-learning.

melhoria de informação visual/háptica, mais identificações positivas, e menos detecção de objetos novos (desconhecidos). A Figura 5.2 mostra o número de objetos mapeados, por ciclos de controle, para cada um dos dois métodos. Como são dadas recompensas para as ações que efetivamente inserem objetos nos mapas atencionais (ver Seção 4.1.7), já era esperado que neste gráfico o método usando Q-learning tivesse um desempenho superior ao outro método. Diversos tipos de análise dos algoritmos podem ser feitas a partir da Tabela 5.1, obtida a partir de um outro experimento em que o sistema operou sobre um mesmo ambiente para ambos os métodos. O experimento em questão foi interrompido após decorridos 204 ciclos de controle. Cada ciclo de controle envolve a transferênca de um estado a outro na máquina de estados finitos descrita na seção 4.1. Resultados similares foram obtidos em outros experimentos nos quais se permitiu que o sistema operasse até que não houvesse mais novas representações detectadas no ambiente. Nestes últimos experimentos, todos os objetos no ambiente foram visitados, categorizados e mapeados. Pelos resultados obtidos, o que podemos dizer é que ambos os métodos tiveram um bom desempenho na tarefa de inspeção, envolvendo atenção e categorização. Não podemos entretanto afirmar que o método usando Q-learning teve um desempenho significativamente melhor. Em outras tarefas que tenham um espaço de estados e ações mais complexos a diferença deve ser mais acentuada.

A Figura 5.3 mostra a convergência do processo de aprendizado Q-learning do qual a política de controle é derivada. É claro que não há como medir o desempenho

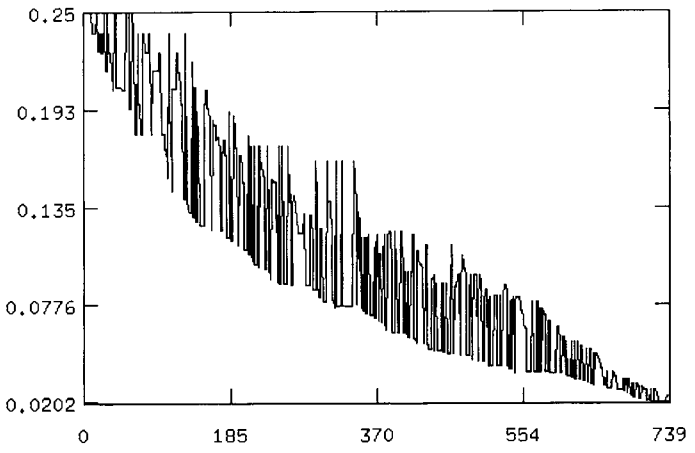


Figura 5.3: Convergência do processo de aprendizado Q-learning. O eixo horizontal mostra o número de ciclos de controle operados e o eixo vertical mostra o erro de diferença temporal (ver Equação 2.5 no algoritmo Q-learning apresentado na seção 2.3).

Método	Mudanças de antenção	Melhorias visuais/ hápticas	Identific. positivas	Novos objetos	Objetos mapeados
Q-learning	72	76	29	27	56
Simples	70	82	20	32	52

Tabela 5.1: Dados obtidos após 204 ciclos de controle, contendo o número de mudanças de atenção, o número de tentativas de melhoria na informação visual/háptica, o número de objetos positivamente identificados, o número de objetos descobertos no ambiente, e o número de objetos efetivamente mapeados.

do processo de treinamento precisamente em unidades de tempo, porque o sistema usa um relógio simulado, mas podemos ver que ele aprende qual a melhor política de controle a ser usada relativamente rápido, em cerca de 700 ciclos de controle.

Outro resultado importante obtido dos experimentos realizados em simulação é que todas as instâncias de objetos foram visitadas (olhadas pelo Roger), identificadas como novas ou já existentes na memória e mapeadas, por ambas as estratégias de controle. A Figura 5.4 mostra a interface do Roger em duas situações diferentes, obtidas enquanto ele percorria o ambiente.

5.2 Experimentos e Resultados na Cabeça Estéreo

Para a realização de experimentos usando a plataforma de hardware, os objetos encontram-se sobre uma mesa e o robô, cujo movimento é restrito pelo espaco que

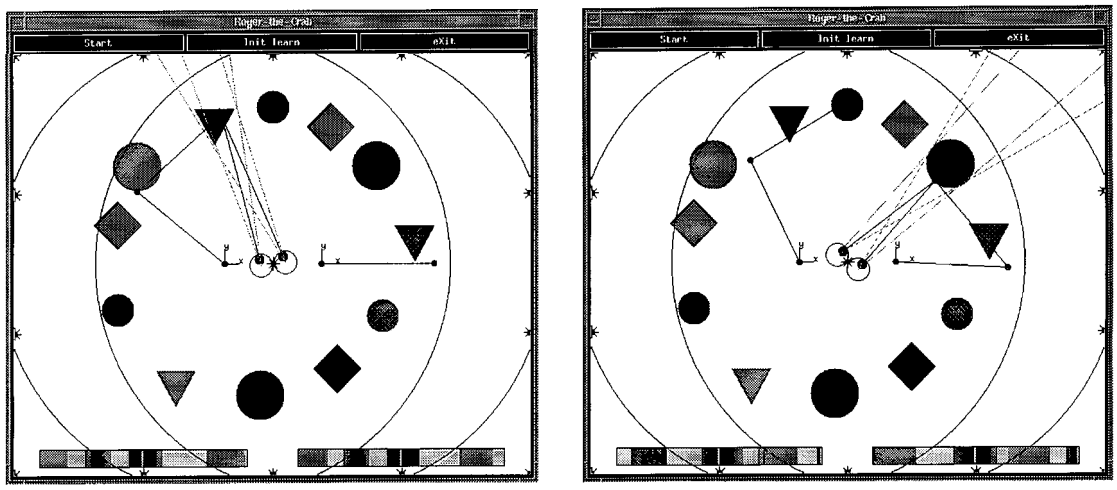


Figura 5.4: Estes dois quadros, extraídos de uma sequência obtida enquanto Roger percorria o ambiente, mostram o simulador em duas situações diferentes. Em ambas situações, informação a respeito dos objetos está sendo extraída e a correspondência sendo realizada na memória associativa. Note que o braço está sendo requisitado nas duas situações.

envolve a mesa, realiza vários experimentos envolvendo atenção, identificação ou reconhecimento e tarefas combinadas. Estes experimentos e os resultados alcançados serão detalhados a seguir.

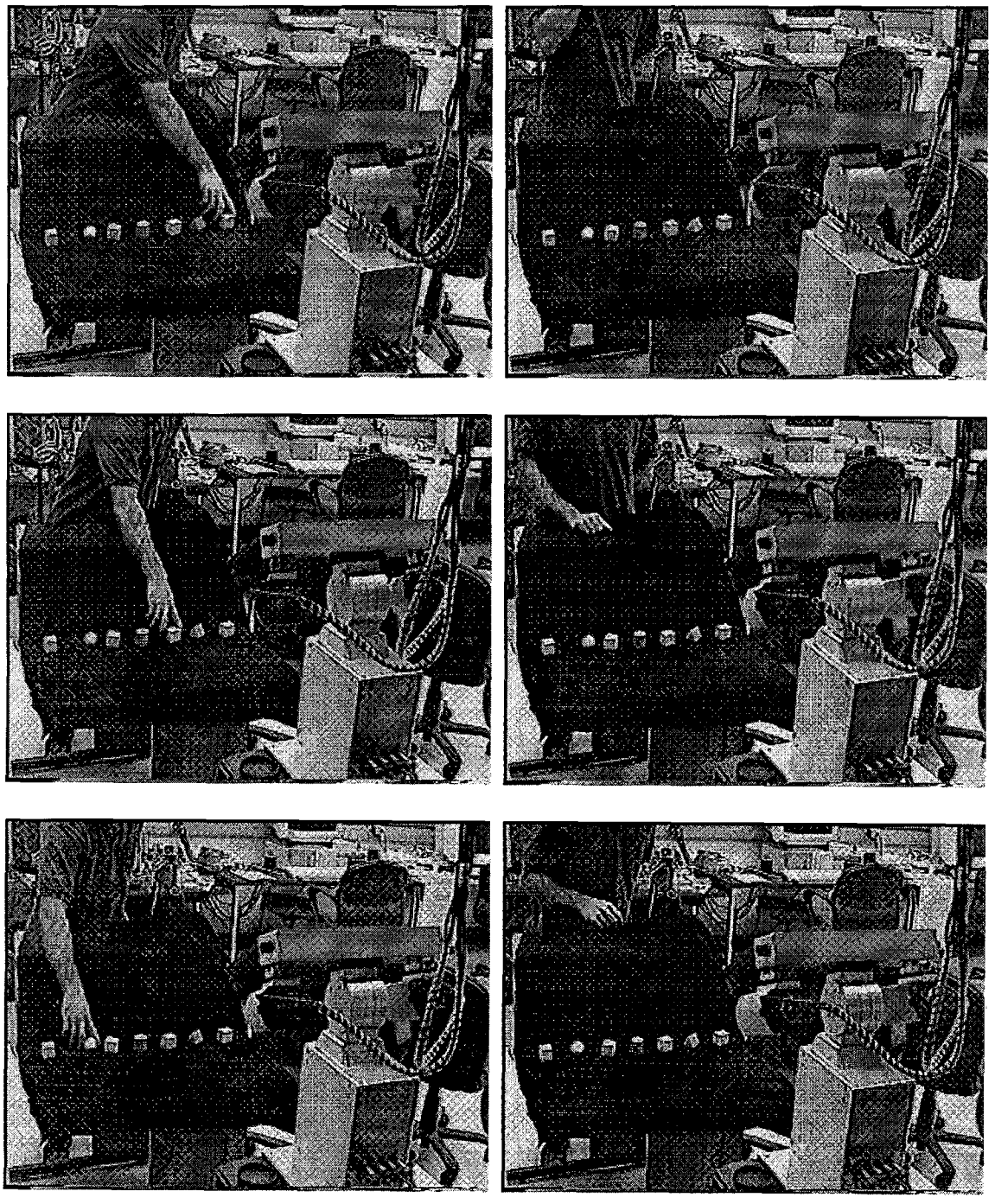
5.2.1 Experimentos e Resultados Envolvendo Atenção

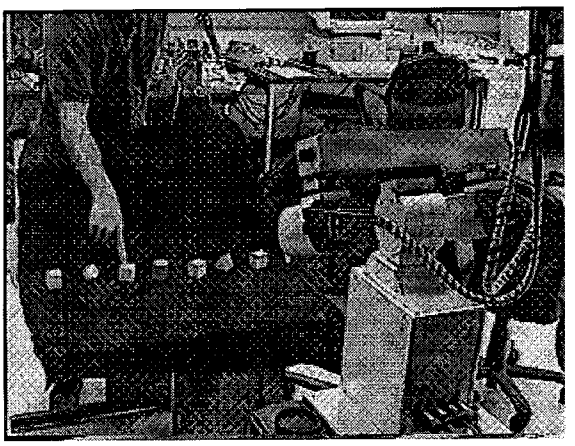
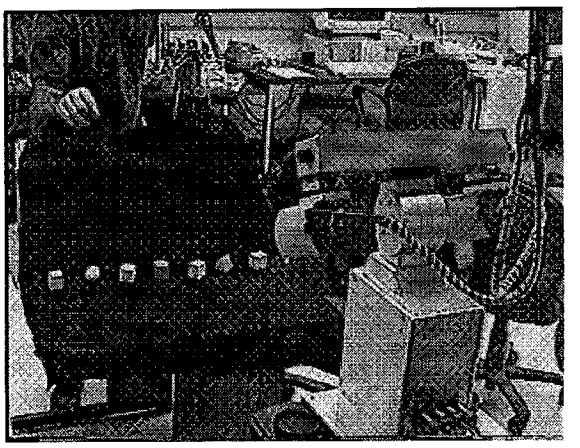
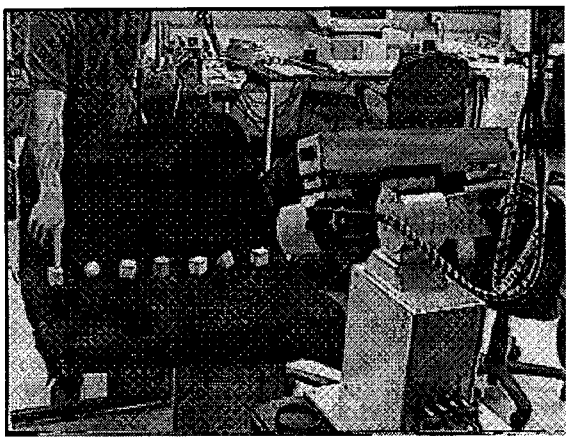
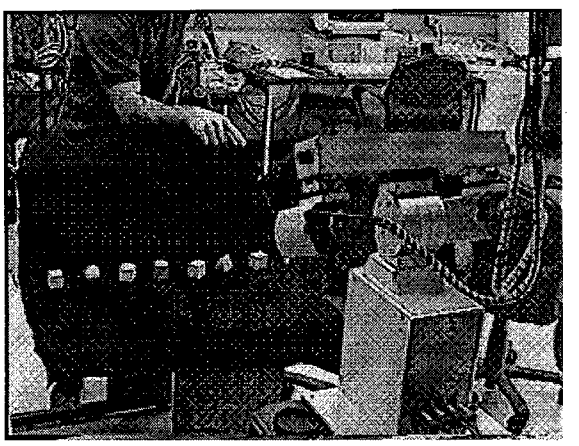
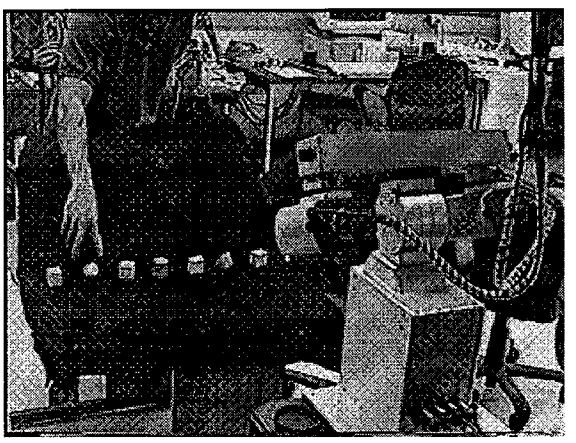
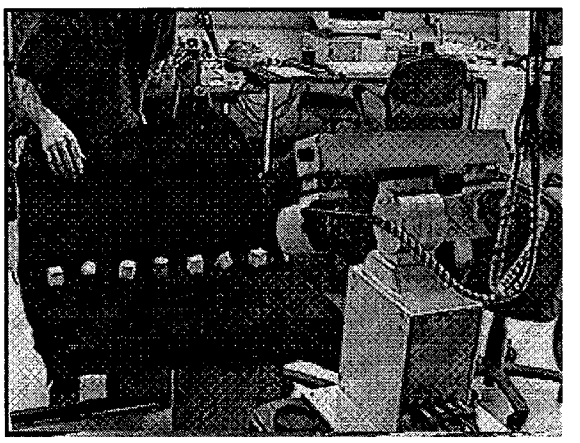
No que se refere à atenção, o comportamento esperado para a cabeça estéreo é que ela se mova de um objeto a outro, de forma que todos os objetos na mesa recebam a atenção. Três modalidades de tarefas atencionais foram testadas aqui, vistas a seguir. Como resultado prático, em todos os testes realizados para cada um dos três tipos de experimento, o robô visitou todas as regiões, descobrindo novas representações ou identificando outras já existentes na memória e mapeando todos os objetos nos seus mapas atencionais.

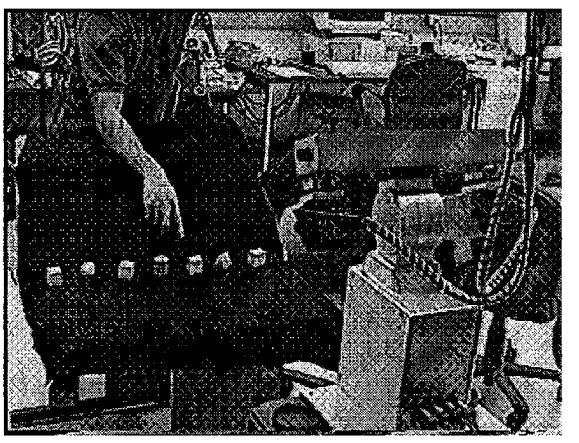
No primeiro tipo de experimento, indicamos os objetos para o robô sequencialmente, tocando ou apontando para os mesmos. Neste experimento, o robô usa primariamente o padrão de movimento (este predomina na determinação da região de interesse pelo processo de atenção) para colocar o objeto próximo da região da fóvea. Então, com a ausência do movimento (a mão é retirada bruscamente da proximidade do objeto), o robô usa as características baseadas em intensidade para colocar o objeto no centro da fóvea. Objetivamente, o robô verge nos objetos apontados, atendendo ao apelo de um sinal (dado pelo movimento realizado). A Figura 5.5 mostra uma sequência de movimentos ilustrando um dos experimentos deste tipo realizado.

No segundo tipo de experimento, não há nenhuma sinalização através de movi-

Figura 5.5: Na sequência ilustrada a seguir, os quadros do lado direito mostram a situação em que o robô atende ao apelo dado pelo movimento do braço e da mão que apontam para os objetos postados sobre a mesa (quadros do lado esquerdo). A mudança no direcionamento das câmeras vergindo na direção dos objetos apontados pode ser notada de um quadro a outro da sequência. As câmeras estão localizadas no lado direito das Figuras, sob uma barra de sustentação que contém os motores de vergência. (A continuação da sequência é mostrada na próxima pagina.)





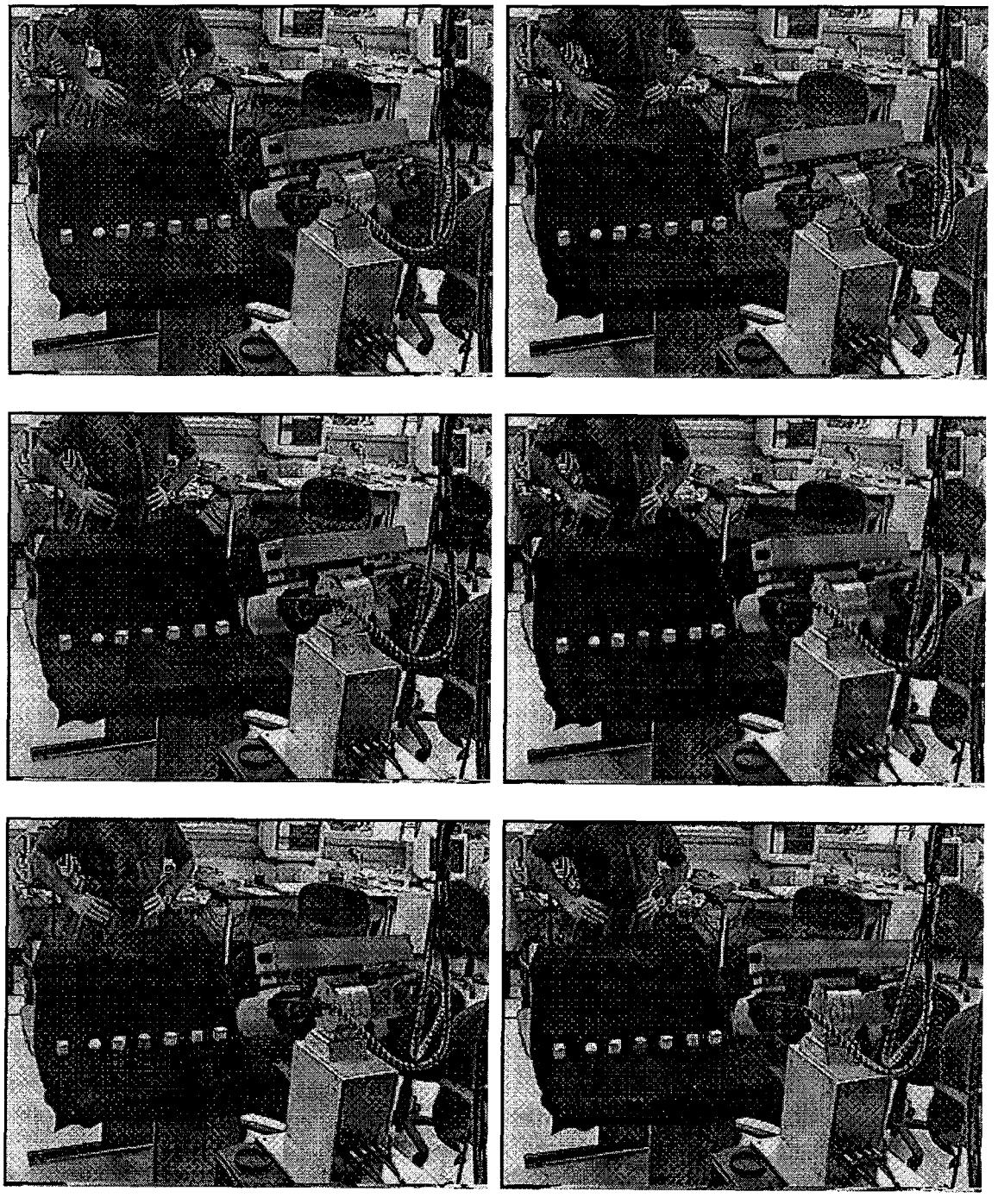
mento e o robô tem que definir por si só as regiões que deve visitar. Assim, apenas as características baseadas em intensidade são usadas. A Figura 5.6 mostra uma sequência gravada durante um dos experimentos deste tipo. A Figura 5.7 mostra os diferentes tipos de objetos detectados noutro experimento deste tipo. Como podemos ver, o mecanismo de atenção funciona de forma a colocar cada um dos objetos na fóvea e a percorrer todos os objetos postados sobre a mesa. Essa Figura valida também o processo de vergência, uma vez que podemos ver que o mesmo objeto é focado em cada par de imagens.

No terceiro tipo de experimento, após todos os objetos na mesa serem detectados e mapeados, um objeto é movido para outra posição ou ainda um objeto é retirado ou adicionado ao conjunto corrente. Verificou-se que o robô foi capaz de retificar os mapas internos para as posições alteradas. Para movimentos que ocorrem dentro do campo de vista das câmeras, o robô usa primariamente os padrões de movimento referentes a eles, seguido de características baseadas em intensidade. Quando não existir mais o movimento, ou seja, após o objeto parar na posição final desejada, essas últimas características prevalecem. Se um movimento ocorre fora do campo de vista das câmeras, o comportamento de inspeção (ver subseção 3.2.3) que o sistema segue após todas as regiões do ambiente serem visitadas, faz com que aquelas posições em que houver mudanças (no caso, a posição em que o objeto se encontrava e a sua nova posição) sejam detectadas e retificadas. É claro que se trocarmos de posição dois objetos idênticos, fora dos campos de vista das câmeras da cabeça estéreo, fazendo com que eles mantenham o mesmo posicionamento relativo às câmeras, esta mudança certamente não será notada pelo robô (os padrões perceptuais das regiões alteradas permanecerão os mesmos). A Figura 5.8 mostra uma sequência de movimentos, onde o robô segue um objeto cuja movimentação é realizada dentro do campo de vista. Neste caso, o robô realiza um acompanhamento (ou "tracking") do objeto até que este pare na posição final. Os mapas topológicos vão sendo retificados em cada instante, mantendo-se uma representação da imagem corrente.

5.2.2 Experimentos e Resultados Envolvendo Identificação

Nos experimentos envolvendo identificação e reconhecimento, espera-se basicamente que o robô aprenda de forma automática as características de todos os objetos, inserindo uma representação para cada um na memória associativa. Além disso, ele deve reconhecer as instâncias de objetos que já possuem uma representação na memória, e atualizar os mapas topológicos. Ao final, o comportamento de inspeção descrito na seção 3.2.3 deve ser adotado, com a cabeça estéreo movendo-se de uma região para outra, verificando alguma mudança que possa eventualmente ocorrer. A Figura 5.7, vista anteriormente, mostra várias imagens selecionadas do nível de mais alta resolução das multi-retinas (para as câmeras esquerda e direita), gravadas

Figura 5.6: A sequência a seguir ilustra um experimento em que não há uma sinalização ao robô indicando um objeto (o ambiente é estático). O robô muda o seu foco de atenção de um objeto a outro usando primariamente características baseadas em intensidade e textura. A mudança no direcionamento das câmeras, vergindo na direção dos objetos, pode ser notada de um quadro a outro da sequência. As câmeras estão localizadas no lado direito das Figuras, sob a barra de sustentação que contém os motores de vergência. (A continuação desta sequência é mostrada na próxima pagina.)



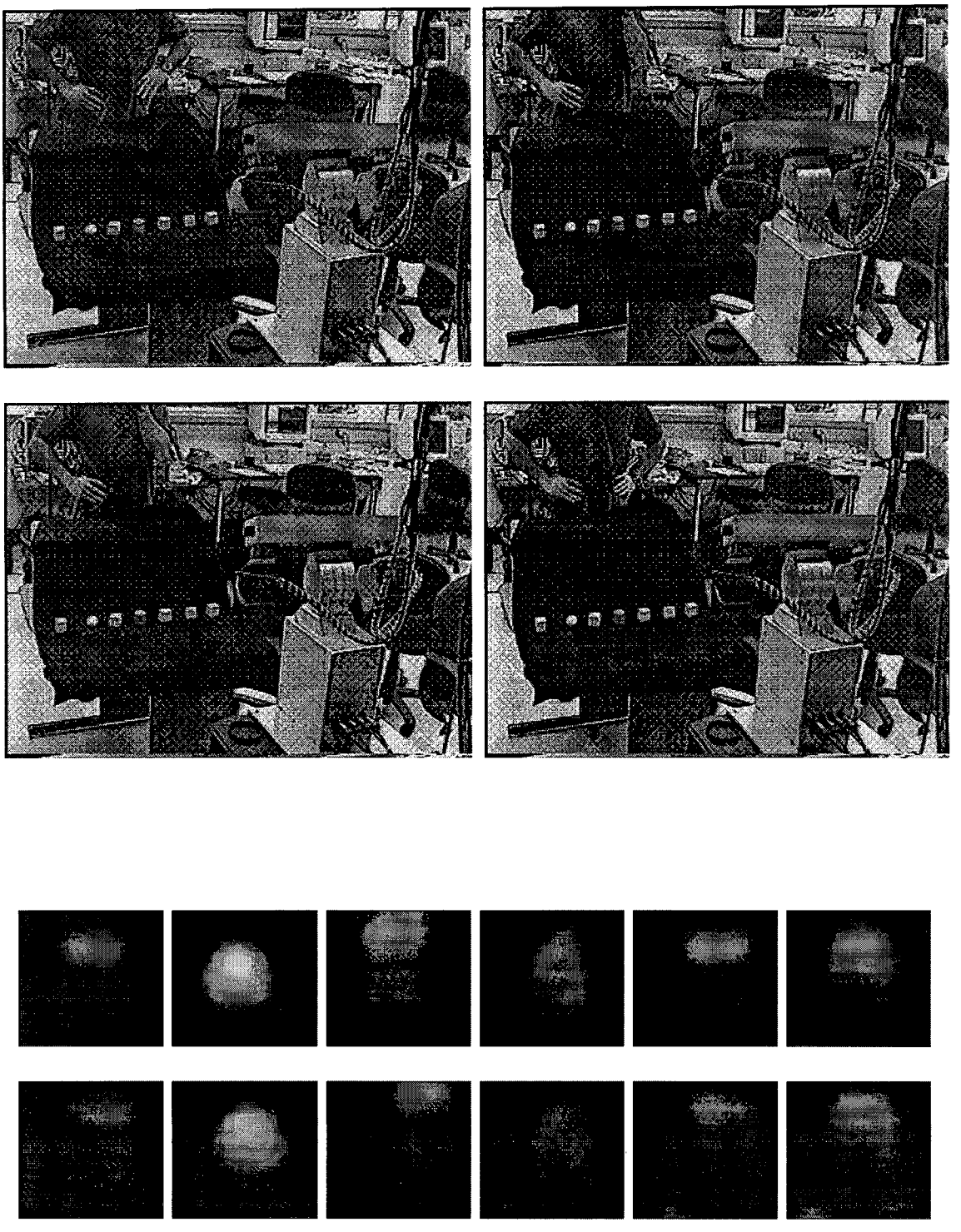


Figura 5.7: Pares de imagens obtidos do último nível (de mais alta resolução) das retinas mostrando apenas os tipos diferentes de objeto (novos) detectados no ambiente num dos experimentos. Da esquerda para a direita: um cilindro azul, uma bola de golfe branca, um cubo de madeira em cor natural, um prisma de faces triangulares verde, um cubo vermelho, e uma bola de tênis verde clara.

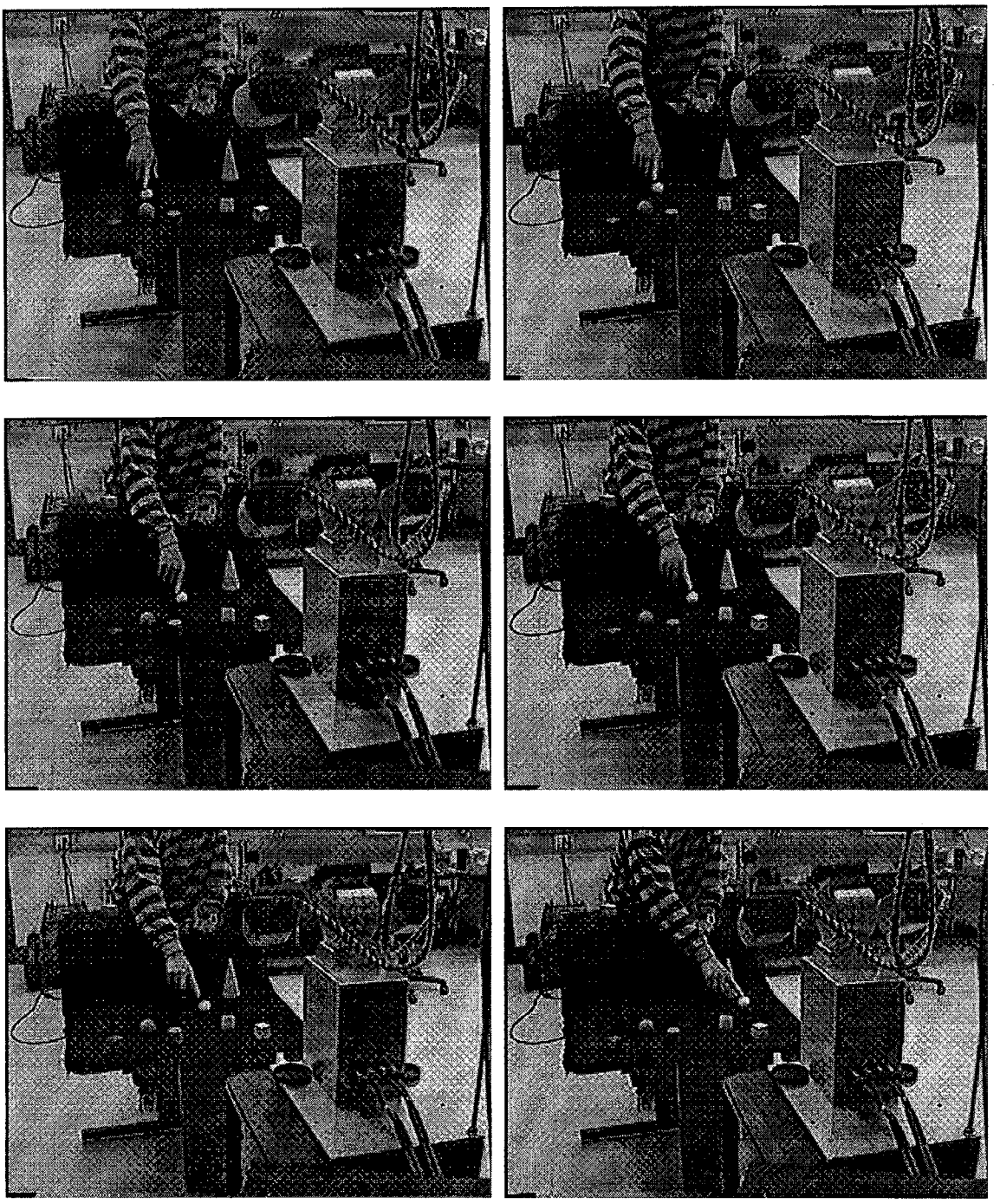


Figura 5.8: A sequência acima ilustra um experimento em que as câmeras seguem um objeto (no caso, uma bola de golfe). Os mapas atencionais são ajustados (atualizados) durante todo o processo, para refletir a percepção corrente do ambiente.

durante um dos experimentos. Embora neste experimento a cabeça estéreo tenha detectado mais de uma instância de cada tipo de objeto, na Figura são mostrados apenas os objetos novos detectados (um de cada tipo). Durante o experimento, o robô visitou todos os objetos. O desempenho da rede neural será discutido a seguir.

Para testar a memória associativa (rede neural BP), vários experimentos foram realizados visando testar o seu desempenho no procedimento de identificação. Basicamente, dados de performance foram colhidos para a fase de treinamento, quando representações novas eram inseridas na rede BP. A Figura 5.9 mostra um gráfico com a média do número de passos completos ou "epochs" versus o número de representações correntemente na rede. A Figura 5.10 mostra um outro gráfico indicando o número de representações versus média do tempo em segundos gasto pelo processo de treinamento. Podemos ver que ambos os gráficos possuem a forma de funções exponenciais, apesar de não serem muito acentuados. Este tipo de gráfico é uma característica do modelo de Back-propagation, como pode ser visto em (RUMELHART *et al.* , 1986; WERBOS, 1988; BRAUN & RIEDMILLER, 1993). Na prática, isto não compromete o desempenho da rede neural, pressupondo-se que o robô não deve lidar com um conjunto infinito de objetos. Como podemos ver no segundo gráfico, para um conjunto de objetos relativamente grande (mais de 20 objetos), o tempo gasto pelo processo de treinamento fica na casa dos 20 segundos. Este tempo pode ainda ser melhorado, como será dicutido mas adiante, no próximo Capítulo. Além do mais, como em nosso caso queremos justamente testar a capacidade de aprendizado automática do robô, a rede BP é inicializada a cada vez que o sistema opera. Ou seja, o robô não conhece nada a respeito do ambiente a cada vez que inicia sua operação. Em um caso prático, o robô salvaria as conFigurações dos pesos da rede BP para um reaproveitamento futuro. Assim, com o passar do tempo, a probabilidade do robô encontrar um objeto novo seria muito pequena, não influindo no desempenho.

Para testar o grau de invariância introduzido com o uso de momentos semi-invariantes no processo de identificação, várias instâncias de vários tipos de objetos foram colocados sobre a mesa em diferentes posicionamentos, que entretanto, não variavam mais do que 30 graus de rotação relativos à posição nas quais foram observados durante a fase de aprendizado. Foram medidas então as ativações da última camada da rede neural no processo de identificação. Neste tipo de experimento, todas as instâncias foram identificadas positivamente, algumas com menor valor de ativação na memória devido às poses degradadas em relação as do treinamento. O gráfico na Figura 5.11 mostra as ativações máxima (linha superior) com o objeto no posicionamento da fase de aprendizado e a ativação mínima (próxima linha) que ainda permite uma identificação positiva. A rede neural foi experimentada com várias instâncias de quatro tipos de objetos, os quais estão especificados na Figura.

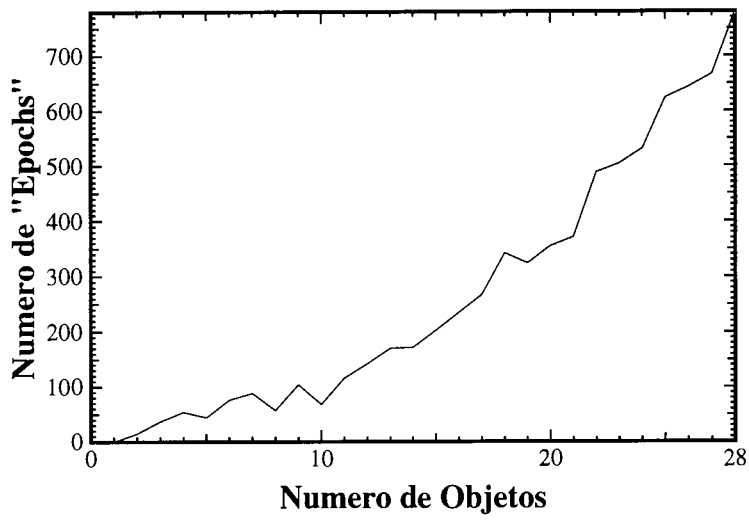


Figura 5.9: Desempenho da rede neural no processo de treinamento. O eixo horizontal mostra número de representações correntemente na rede neural; o eixo vertical mostra o número de passos (ou epochs) gastos no treinamento.

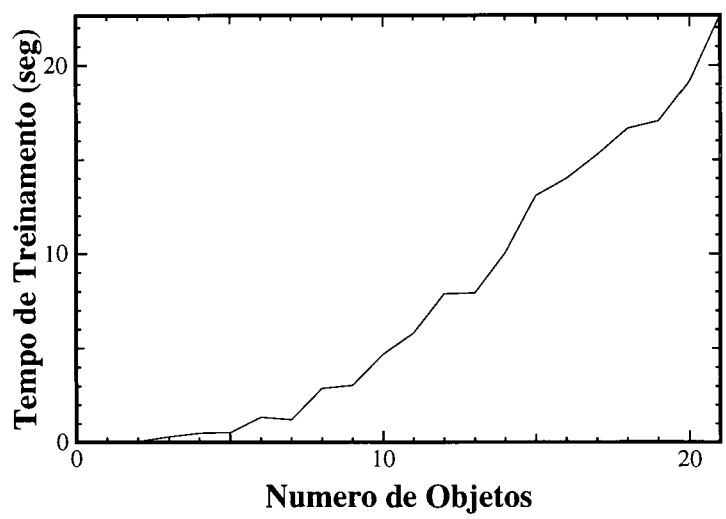


Figura 5.10: Desempenho da rede neural no processo de treinamento. O eixo horizontal mostra número de representações correntemente na rede neural; o eixo vertical mostra o tempo em segundos gasto no processo de treinamento.

As linhas referentes às outras instâncias (com valores de ativação intermediários) não são mostradas, visando evitar confusão. Podemos verificar que rotações de até 30 graus são bem suportadas pelo sistema, resultando ainda numa ativação com um valor mínimo acima do limiar, o que permite uma identificação positiva, segundo o critério adotado nestes experimentos.

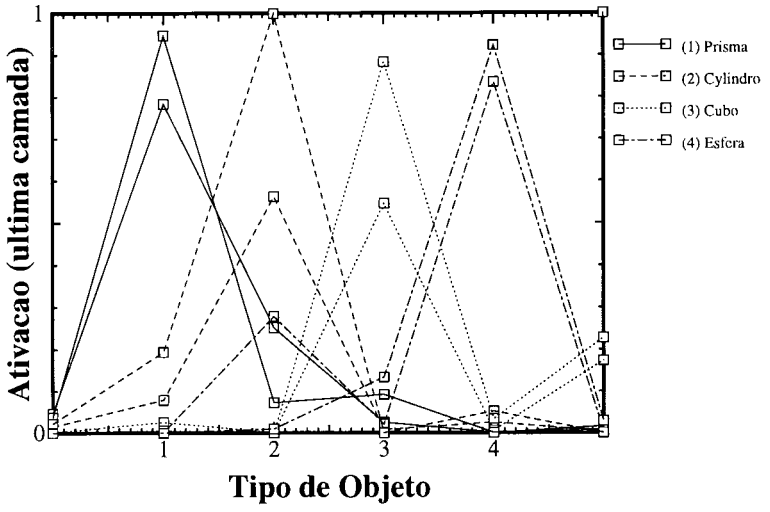


Figura 5.11: Desempenho da rede neural após o processo de treinamento. São mostradas as ativações simultâneas na última camada da rede BP para 4 tipos diferentes de objetos. Para cada objeto (eixo horizontal), a linha superior mostra a ativação máxima conseguida ou seja, quando os objetos estão num posicionamento ideal, ou seja, numa posição próxima àquela na qual foram detectados pela primeira vez. As linhas imediatamente abaixo dessas mostram a ativação mínima, que ainda permite uma identificação (objetos degradados).

Não foram feitos experimentos específicos para medir o grau de invariância no tocante aos deslocamentos, mas foi observado que o sistema suporta bem que objeto esteja situado em partes diferentes da retina, sendo apenas necessário que ele esteja totalmente dentro da imagem no nível considerado pelo processo de atenção. As caraterísticas estéreo só podem ser calculadas caso as duas imagens de um objeto estejam próximas da fóvea. Quanto à invariância com respeito à escala, verificou-se que alterações de até 30 % são bem suportadas. Isto pressupõe que um certo nível seja escolhido pelo processo de atenção que deixe o objeto dentro desta tolerância. Na prática, esta restrição não foi implementada, uma vez que se torna caro computacionalmente realizar um processo de segmentação, ou mesmo testar condições de diferenciabilidade com respeito à intensidade, disparidade, ou outro método qualquer que permita implementar isto.

5.2.3 Desempenho da Cabeça Estéreo

Alguns experimentos no sentido de obter resultados relacionados com o desempenho da cabeça estéreo foram realizados. Nestes experimentos, gravamos os tempos de processamento de todas as fases (ou processos) mais importantes, enquanto o sistema operava. A Figura 5.12 mostra os tempos obtidos ao longo de vários ciclos de controle para três dos processos: mudança de atenção (*shift-attention*), geração de movimentos sacádicos (envolvendo software e hardware) e identificação (*matching*). A Tabela 5.2 mostra separadamente os tempos requeridos para cada um dos processos (ou fases) envolvidos nas tarefas de atenção e identificação. Da esquerda para a direita, na primeira coluna encontra-se uma descrição da fase, na segunda coluna o tempo mínimo para executá-la, na terceira coluna o tempo máximo para isso e na última coluna o tempo médio. Este último (média μ) foi tomado após várias centenas de ciclos de controle para cada um dos processos. Os tempos das tarefas que envolvem cálculos, realizadas no computador após a transferência de imagens do Datacube, podem ser melhorados consideravelmente. O computador usado nos experimentos foi uma Sun Sparc 10 (com um processador de aproximadamente 40 MHz). Correntemente, uma placa dedicada, com um processador Sparc Ultra 10 encontra-se instalada no mesmo gabinete que suporta as placas do Datacube. Este processador opera em uma taxa de aproximadamente 300 MHz. Ainda, como esta última placa compartilha o mesmo barramento que as do Datacube, o tempo para transferência de dados também deve melhorar, dado que um adaptador do tipo *barramento-barramento* era necessário para transferir dados para a Sparc 10. A geração de sacádicos é outro componente que pode ser melhorado também. Basta aumentar o ganho dos controladores de movimento (derivativos posicionais) da interface (PMAC) que em última instância controla os movimentos da cabeça estéreo. Pode-se otimizar este processo ao ponto de se obter sacádicos tão rápidos quanto o dos seres humanos (80 a 200 milisegundos). Um problema que pode ocorrer nesta tentativa de otimização é que a cabeça pode apresentar alguma instabilidade, tremendo e prejudicando consequentemente a aquisição de imagens.

5.3 Análise dos Resultados e Dificuldades Encontradas

Um fator proponderante no sucesso da arquitetura, tanto em simulação quanto na plataforma de hardware foi ter conseguido evitar que um processo de reconstrução mais completo tivesse que ser executado a partir das medidas de disparidade estéreo. Isso foi possível usando-se um espaço de escalas para representação e extração das características e um processo em forma de cascata para os cálculos da disparidade estéreo. Isto facilitou sobremaneira o processamento a ser realizado. Além do mais,

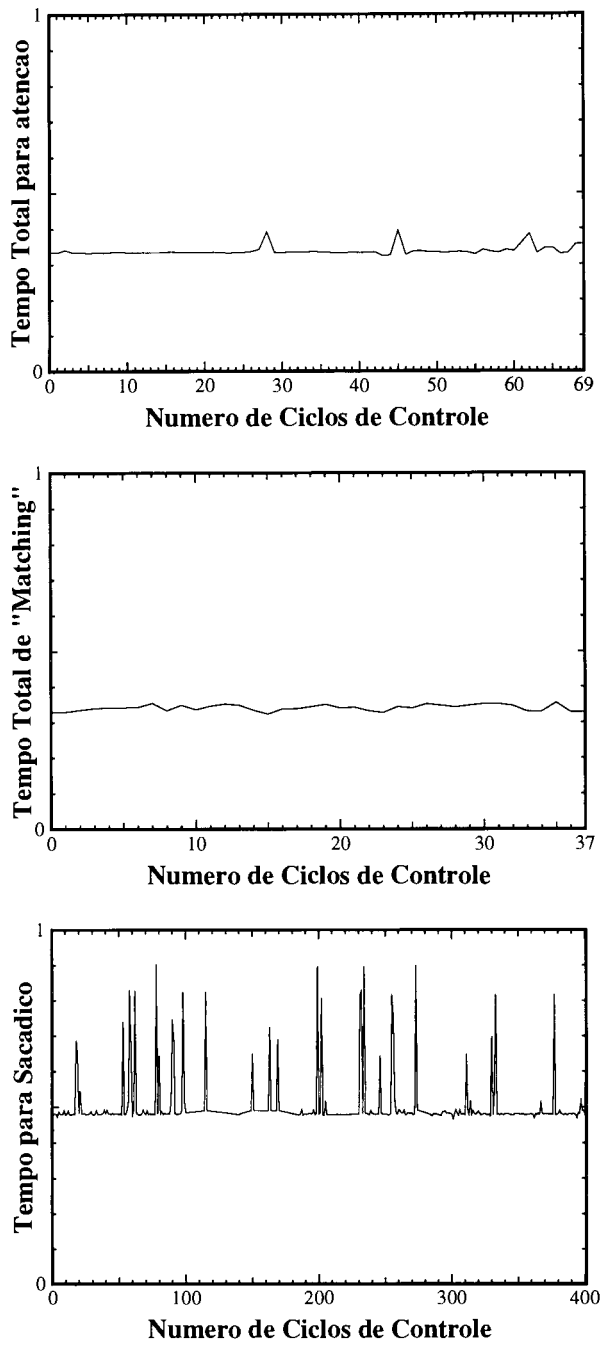


Figura 5.12: Tempo total requerido para mudança de atenção, correspondência na memória associativa (identificação), e geração de movimentos sacádicos. Os tempos em questão incluem também a aquisição de dados, geração das retinas, bem como o processo de cálculo de disparidade estéreo em cascata.

Fase ou processo	Min(sec)	Max(sec)	μ(sec)
Cálculo da retina	0.145	0.189	0.166
Transferência para host	0.017	0.059	0.020
Total aquisição	0.162	0.255	0.186
Pré-atenção	0.139	0.205	0.149
Mapa de saliência	0.067	0.134	0.075
Total atenção	0.324	0.395	0.334
Total sacádico	0.466	0.903	0.485
Extração de Características	0.135	0.158	0.150
Correspondência na memória	0.012	0.028	0.019
Total correspondência	0.323	0.353	0.333

Tabela 5.2: Tempo requerido para cada subprocesso ou fase. O cálculo das retinas em multi-resolução é realizado dentro da arquitetura Datacube. A fase pré-atencional é realizada por um programa operando no computador host, após a transferência dos dados providos pelo Datacube. O cálculo das características usadas para se obter uma correspondência na memória associativa são calculadas também no computador host. Ambos, pré-atenção e cálculo das características para correspondência, incluem a exibição das retinas no monitor do computador host. O movimento sacádico inclui também o envio de dados (parâmetros dos movimentos) para a interface PMAC e esperar até que os movimentos solicitados sejam completados.

uma vez que os controladores robóticos usam efetivamente a disparidade para calcular os movimentos, trabalhando em cima de um modelo de correção de erros (ver seção 2.3), não é necessária uma reconstrução estéreo completa.

O modelo de extração de características semi-invariantes e o uso da memória associativa através de uma rede BP demonstrou ser apropriado. Convém frisar que outros modelos (KOHONEN, 1990; VIOLA, 1996; PIATER & GRUPEN, 1999; COELHO *et al.* , 1999; MARENGONI *et al.* , 1999) podem ser testados, e (talvez), com alguma adaptação, melhores resultados possam ser alcançados. Como o modelo usando BP adaptou-se perfeitamente à funcionalidade da arquitetura de controle, tanto em simulação quanto na plataforma de hardware, não influindo no desempenho final nos testes realizados, nos atemos a ele. Além do mais, o modelo BP consegue lidar com características que estejam faltando de forma razoável. Isto pode ter uma boa aplicação para casos de oclusões, em que os objetos estejam parcialmente observáveis.

Neste trabalho usamos o mesmo vetor de características para ambos os propósitos (atenção e categorização). O uso da magnitude do vetor gradiente da diferença de quadros consecutivos como um dos componentes a serem levados em consideração no processo de atenção permite que regiões estáticas com alto valor de intensidade, sujeitas a iluminação artificial, tenham eventualmente diferentes valores para o padrão movimento. A iluminação artificial é uma função do tempo, obedecendo a um ciclo (algumas vezes igual a 60 Hz). Isto somado a outros efeitos do ambiente

explicam a variação da intensidade em uma mesma região. Mesmo variações suaves ocorrem produzindo diferentes intensidades nos sensores das câmeras. Esta característica não é necessariamente ruim, uma vez que o sistema muda a atenção para regiões de altos valores de intensidade, que são intuitivamente atrativas.

Ainda, levando-se também em consideração características locais ponderadas para o processo de atenção, esta é focada em posições que localmente representam regiões, mais do que simples pontos. Na hipótese de que esta região seja parte de apenas um objeto, isto pode facilitar uma segregação do ambiente em regiões de interesse, tomando-se uma média da intensidade e usando um intervalo ao redor desta média como um filtro passa banda. Então, pode-se usar uma função de espalhamento, baseada nas respostas deste filtro que aumente a região de escopo, até encontrar uma possível fronteira para um objeto.

Em simulação, o modelo Q-learning agindo ao nível de processo supervisor orquestrou eficientemente o sistema, com um desempenho ligeiramente melhor do que a estratégia simples, o que entretanto não temos ainda condições de generalizar. O modelo simples usado, baseado na segmentação do ambiente percebido em regiões de interesse e extração seletiva de características, mostrou-se adequado devido ao alto custo computacional do processo de identificação. Já na implementação realizada no hardware, o mesmo modelo não foi usado por dificuldades de se encontrar um método de segmentação que seja factível em tempo real.

Uma dificuldade que apareceu durante a implementação em ambas as plataformas foi como determinar exatamente a convergência do processo de atenção. A solução encontrada na plataforma de hardware foi fazer com que o processo de *shift-attention* opere em todo ciclo de controle e determinar um critério de convergência para ele. Assim, mudar a atenção pode implicar em mais de uma execução do processo *shift-attention*. Esta dificuldade surge também devido a que um mesmo objeto pode dar respostas diferentes aos sensores, em momentos diferentes. Uma convergência da função *shift-attention* significa ordenar à memória associativa que tente estabelecer a correspondência.

Outra dificuldade encontrada na plataforma de hardware, que nos levou a restringir os movimentos da cabeça estéreo, relaciona-se à emissão de luz artificial provocada pelos monitores dos computadores existentes na sala em que a mesma se encontra (que geralmente trabalham em ciclos de 60 hertz) e também a sua reflexão em superfícies espelhadas. Devido a isto, algumas regiões do ambiente estavam em constante mudança e a cabeça estéreo tinha predileção por estas regiões. Ao restringir o ambiente, solucionamos em parte este problema.

Capítulo 6

Discussões, Conclusões e Trabalhos Futuros

Usamos inicialmente a plataforma de simulação para validar a arquitetura de controle para um sistema robótico sensorial multi-modo que descrevemos neste trabalho. A implementação posterior na plataforma de hardware, permitindo à mesma operar em tempo real, consolidou de forma definitiva nossa proposta. Estas ferramentas podem ser empregadas em uma série de tarefas de mais alto nível que exijam atenção e categorização de padrões. Atenção e categorização são subtarefas básicas e é difícil imaginar a realização de alguma tarefa de caráter prático que não as use. Identificação é primordial em qualquer tarefa. Por outro lado, sem a habilidade de mudar o foco de atenção não há cognição. Assim, estas duas tarefas estão integradas uma à outra de tal forma que um sistema de comportamento ativo necessita de ambos subsistemas trabalhando bem para poder realizar outras tarefas. Em outras palavras, concluímos que é praticamente impossível lidar com atenção sem lidar com identificação e reconhecimento. Podemos ser ainda mais categóricos arguindo que não apenas todos os sistemas sensoriais devem agir como uma unidade, mas também todas as funções de um sistema cognitivo devem ser facilmente integráveis para que se possa pensar em iniciar algum desenvolvimento. Neste trabalho, nós desenvolvemos sob esta arquitetura básica, um sistema integrando um mecanismo de atenção que usa informação visual e háptica, apesar desta última não ter sido testada em hadrdware, mais um classificador que usa uma rede neural. Alguns exemplos de tarefas de mais alto nível poderiam ser tais como inspeção ou vigilância (realizado neste trabalho com sucesso), planejamento de movimento, orientação, desvio de obstáculos e navegação. Como exemplo de uma tarefa mais avançada, um robô poderia usar como base os procedimentos desenvolvidos neste trabalho para aprender tarefas de exploração, por exemplo, navegar pelas diferentes salas de um edifício reconhecendo e identificando as pessoas.

Apesar de se usar apenas informação visual e háptica neste trabalho, a arquitetura proposta é mais geral. Outros sensores tais como microfones estéreo, detectores

de batimento cardíaco, detectores de temperatura, sonares e sensores infra-vermelho, entre outros, podem ser adicionados facilmente à arquitetura. Isto melhoraria ainda mais ambos procedimentos (atenção e categorização), produzindo um mapa de saliências melhor e um conjunto de características discriminativas também melhor. O modelo *Controller Oriented* usado na implementação realizada na cabeça estéreo, em que dentro de um ciclo de supervisão todos os recursos são controlados, permite que outros processos concorrentes possam ser desenvolvidos independentemente e incorporados facilmente à arquitetura.

A construção e manutenção de um mapa topológico específico para codificar a informação espacial, contendo a posição e orientação dos padrões detectados em regiões que já tenham sido foco do processo de atenção, torna-se computacionalmente cara. Assim, a solução adotada neste trabalho, que usa os mesmos mapas atencionais para guardar informação sobre o ambiente, foi fundamental no desenvolvimento do sistema na plataforma de hardware. Ao invés de se postar nesses mapas apenas os padrões das regiões que já receberam atenção, os mesmos contêm também informação espacial sobre regiões que ainda não receberam atenção, mas que já foram detectadas pelo processo de pré-atenção. Este modelo, além de tornar mais rápido um mapeamento topológico efetivo, facilita a codificação de nova informação. Parece ser muito natural que se saiba da existência de um padrão ainda sem uma identificação positiva, mas cuja informação espacial, posição e orientação já estejam armazenadas nos mapas sensoriais. Em algumas situações, não há nem mesmo necessidade de se categorizar todos os objetos em uma cena, mas apenas aqueles de interesse imediato para a tarefa sendo executada, como é o caso em navegação, onde certas marcas ou objetos geralmente situados à frente do robô são priorizados.

6.1 Trabalhos Futuros

Tanto na plataforma de simulação quanto em hardware, o processo de vergência dos olhos foi implementado basicamente por maximização de medidas de correlação. O foco também pode ser usado para este propósito, além de fazer as câmeras acomodarem numa determinada região. Uma diferença de profundidade aproximada pode ser extraída da diferença de foco entre duas regiões (HORII, 1994). Considerando-se, para o olho dominante, uma dessas regiões como sendo a região correntemente na fóvea e a outra região a do objetivo, selecionada pelo processo de atenção, e supondo-se que os dois olhos devam ter o mesmo valor para o foco, o ângulo de vergência para o olho não dominante pode ser também determinado aproximadamente a partir da diferença de foco entre as regiões consideradas. Desse modo o cálculo de valores de correlação entre imagens pode ser usado apenas numa segunda fase, para uma vergência mais fina, que operaria simultaneamente ao processo de

acomodação do mecanismo de foco. Em caso de oclusões, a comparação dos valores de correlação com um limiar poderia descartar o processo de vergência mais fina usando a correlação e somente o foco seria usado para acomodação. Calcular o foco envolve realizar medidas estatísticas sobre as imagens. Mudanças no histograma, ou maximização de somatórios locais do gradiente determinam o melhor foco, o que pode ser realizado na arquitetura de processamento de imagens Datacube.

Várias sugestões de trabalhos futuros podem ser tentadas a título de melhorar o desempenho da rede neural. Um treinamento local ao invés de um global, como é atualmente usado, pode ser aplicado quando um novo padrão de representação for detectado no ambiente. Isso possibilitaria aumentar o número de características de entrada para a rede BP, e neste caso, uma possibilidade que consideramos interessante é incluir padrões de movimento que podem ser calculados usando filtros que estimam o fluxo ótico, como por exemplo os descritos em (SOATTO *et al.* , 1997; LEE & KAY, 1991). Mais ainda, um agrupamento (ou "clustering") poderia melhorar a performance da BPNN, bem como um modelo hierárquico pode ajudar a lidar com dados parcialmente observáveis. Finalmente, outros modelos que não uma rede do tipo "back-propagation" podem é claro ser usados para tentar melhorar o desempenho do classificador.

Além do modo direto usado para direcionar atenção, com uso de uma função com pesos fixos para transferir ativação dos mapas de características para os de saliências, uma função, com pesos variáveis, de acordo com a tarefa sendo executada, pode ser derivada via aprendizado Q-learning. Neste caso, o conjunto de tarefas seria parte do espaço de estados, e o sistema receberia recompensas para as ações realizadas em tarefas que conduzissem a boas situações em busca de sua realização, dependendo do contexto. Uma outra melhora para esta função de transferência é fazê-la variar em função do tempo. Numa primeira fase, o sistema usaria essencialmente sinais de movimento para determinar objetivos atencionais e, numa segunda fase, características baseadas em intensidade e textura (características gaussianas) seriam predominantemente empregadas para focar a atenção no ponto exato. A atenção encoberta (mudar a janela de atenção sem realizar movimentos físicos) também pode ser incorporada ao sistema, o que seria mesmo relevante para uma melhoria de desempenho. Usando estes modelos, uma função de transferência mais próxima de um modelo biológico possa talvez ser derivada.

Outro experimento que poderia ser realizado no tocante à atenção é considerar as características relativas ao movimento apenas no nível de menor resolução (mais grosseiro). Isto também tem inspiração biológica. O caminho *magno-celular* é conhecido por detectar movimento e tem uma maior influência predominante na atenção, como pode ser visto no apêndice B. Outros níveis de mais alta resolução teriam uma maior influência no caso de atenção top-down. Isto também tem alguma

relação com o modelo biológico, mais especificamente com o caminho '*parvo-celular*. Se um objeto necessita atenção em uma parte específica, um mecanismo top-down colocaria esta região no primeiro nível cuja resolução seja suficiente para descrever a região considerada no objeto com um detalhamento suficiente.

Ainda, em relação à atenção *top-down*, acreditamos que a mesma arquitetura desenvolvida aqui possa ser empregada em outras tarefas que exigem exclusivamente esse tipo de atenção, como é o caso quando procuramos por um objeto específico em meio a outros que possam distrair a nossa atenção. Nesse caso, a procura pelo objeto poderia ocorrer em duas fases distintas, uma paralela e outra sequencial. Na primeira fase, um filtro definido a partir da informação do modelo seria aplicado às imagens realçando regiões com características similares a do modelo. Numa segunda fase, uma busca sequencial nas regiões mais realçadas determinaria ou descartaria a presença do objeto em questão. Políticas de controle também poderiam ser derivadas usando Q-learning. Numa fase de aprendizado, o sistema receberia recompensas por detectar os objetos considerados no ambiente. Com isto, o sistema aprenderia qual o melhor conjunto de características que é relevante na procura pelo objeto em questão. Esta informação ficaria armazenada na MLT.

Apêndice A

Redes Neurais do Tipo "Back-Propagation"

O algoritmo de aprendizado baseado em redes neurais do tipo "back-propagation" tem uma história relativamente recente. O algoritmo foi introduzido por Werbos (WERBOS, 1974) e também, de forma independente, por Parker (PARKER, 1985). Sua importância em relação à solução de problemas envolvendo redes neurais compostas por várias camadas ficou ressaltada a partir do trabalhos de Rumelhart et al (RUMELHART *et al.* , 1986). Mais recentemente, Le Cun (LE CUN, 1985) e Simard (SIMARD, 1991) mostraram o seu relacionamento com a teoria de controle clássica. A seguir descreveremos suscintamente as particularidades do algoritmo "back-propagation" (doravante referido simplesmente como BP).

Uma rede neural do tipo BP, como mostrada graficamente na Figura A.1, é totalmente conectada e seus nós são organizados em camadas. O fluxo de ativação nos nós da rede BP se processa "para a frente", ou seja, num único sentido: os valores de ativação impostos nos nós da camada de entrada são propagados para os nós das camadas intermediárias e finalmente promovem alguma ativação nos nós da camada de saída. Cada nó de uma camada é conectado a todos os outros nós da camada subsequente. Uma rede BP pode conter mais de uma camada intermediária, mas em geral, boas aproximações podem ser conseguidas com apenas três camadas. O conhecimento adquirido pela rede BP é codificado em pesos (sinapses) ajustados por um processo de treinamento, em cada ligação ou conexão entre as unidades (nós) da rede. A ativação obtida na última camada determina uma resposta, representando a ativação obtida para toda a rede.

Uma rede BP é tipicamente inicializada com um conjunto aleatório de valores para as sinapses. Para o treinamento da rede, é necessária uma fase de treinamento, usando pares de vetores de entrada e de saida associados, que sejam "a priori" conhecidos. A rede ajusta os valores iniciais das conexões baseando-se em algumas regras de aprendizado (Delta, Hebbian). Cada vez que um par de vetores entrada-saída é apresentado, dois estágios para a ativação ocorrem: um passo de propagação

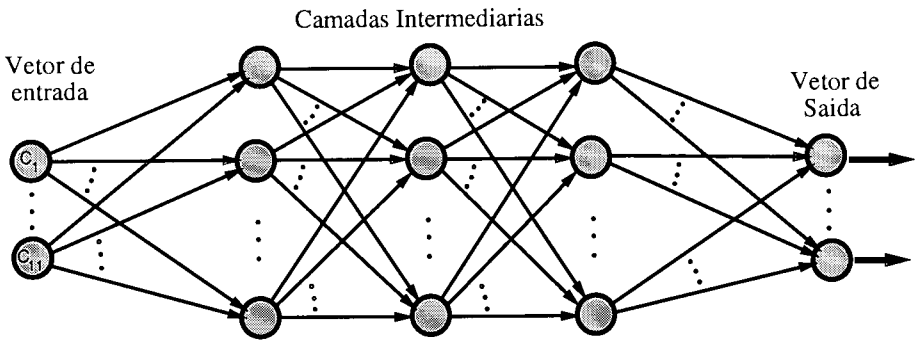


Figura A.1: Rede neural do tipo "back-propagation". No caso do exemplo mostrado, a rede possui três camadas intermediárias

"para a frente" e um de propagação "para trás". No passo para a frente, a ativação imposta à camada de entrada é propagada para as camadas subsequentes, resultando em ativação na camada de saída. Então, durante o passo de propagação para trás, a saída atual da rede, experimentada após o passo para a frente, é comparada com a saída ideal, dada pelo vetor de saída associado àquela entrada. Erros são calculados para os nós da camada de saída e os pesos ou sinapses das conexões que chegam aos nós desta última camada podem então ser ajustados para se tentar uma redução dos erros. Estas estimativas de erros para os nós da última camada são então usadas para se derivar estimativas para os nós das camadas intermediárias. Finalmente, os erros são retro-propagados para as conexões que saem dos nós da camada de entrada. Após cada passo completo (propagação para a frente e para trás), o sistema "aprende" incrementalmente dos pares de vetores de entrada-saida associados e reduz o erro diferencial entre as saídas dada pela rede (estimada) e a saída real dada pelo vetor de saída associado (desejada). Após um treinamento intensivo, a rede irá certamente estabelecer as relações existentes entre os vetores de entrada e de saída através dos pesos ou sinapses.

Devido aos cálculos na rede BP, a um dado instante, envolverem apenas duas camadas adjacentes, nas Equações seguintes adotamos A como sendo o número de nós numa dada camada r e B como sendo o número de nós na camada seguinte $r+1$. O valor do nó 0 (x_0) é usado como um limiar para cada camada. O algoritmo BP específico adotado nesta pesquisa é estruturado como apresentado a seguir:

1. *Inicializar as sinapses (ou pesos) das conexões: Inicialmente, são atribuídos valores aleatórios entre* -0.1 *e* $+0.1$ *para todos os pesos* w_{ij} *das conexões da rede, isto é:*

$$w_{ij} = random(-0.1, +0.1) \forall i \in \{A\} \, and j \in \{B\}$$

2. *Inicialização do limiares: Para cada camada, é atribuído o valor* 1 *ao nó que é usado como limiar.*

$$x_0 = 1.0$$

3. *Inicialização das sinapses na primeira camada (de entrada): A rede recebe cada par de vetor de entrada e de saída associado e usa os valores do primeiro destes como valores para os nós da camada de entrada, isto é,* $x_i, \forall i = 1, ..., A$.

4. *Propagação para a frente: Baseando-se na função de ativação sigmoidal, contínua e diferenciável, dada pela Equação A.1 mostrada abaixo, são calculados os valores para os nós de todas as outras camadas, a partir dos valores da primeira (de entrada), e até que os valores dos nós da última camada (de saída) sejam calculados. Os valores calculados para cada camada são usados para calcular os valores da próxima camada.*

$$o_i = (1 + e^{-\sum_{i=0}^{A} \omega_{ij} x_i})^{-1} \tag{A.1}$$

onde w_{0j} *é o valor que serve como limiar para o nó* j *na camada subsequente.*

5. *Propagação para trás: Partindo da última camada e propagando-se para trás até a primeira camada, são calculados os erros para os nós de cada camada. Uma fórmula de correção de erros (regra Delta), a qual é baseada na derivada da função de ativação e que tenta minimizar o erro nos nós da última camada (de saída) é usada para calcular os erros. Esta função usada para calcular os erros define uma superfície sobre o espaço das sinapses (pesos) e estas são modificadas na direção do gradiente daquela superfície (estratégia do gradiente descendente). Para a última camada, o cálculo dos erros é baseado na diferença entre os valores ideais* y_j *(valores dados pelo vetor de saída associado à entrada correspondente) e os valores atuais* o_j *(calculados pela rede). Os erros para a última e para as outras camadas são calculados respectivamente pelas Equações A.2 e A.3.*

$$\delta_j = o_j(1 - o_j)(y_j - o_j), \forall j = 1, ..., B \tag{A.2}$$

$$\delta_j = o_j(1 - o_j) \sum_{k=1}^{B} \delta_k \omega_{jk}, \forall j = 1, ..., A \tag{A.3}$$

6. *Ajuste dos pesos das conexões (sinapses): Os erros calculados acima são usados para ajustar as sinapses em todas as camadas da rede. Uma taxa de aprendizado ϵ e um fator momentum α são usados para agilizar o processo de aprendizado. Os novos valores para as sinapses são ajustados pela Equação A.4.*

$$\Delta\omega_{ij}(t+1) = \epsilon\delta_j o_i + \alpha\Delta\omega_{ij}(t), \tag{A.4}$$

Na Equação A.4 os valores para as variáveis o_i e δ_j são determinados no instante de tempo $t+1$ e o valor de $\Delta\omega_{ij}(t)$ é a mudança experimentada no peso durante os passos para a frente e para trás prévios (no instante t). O fator *momentum* inclui a influência da mudança de peso experimentada no passo anterior no ajuste corrente da sinapse.

Uma iteração completa incluindo passos para a frente e para trás e ajuste das sinapses usando todos os pares de vetores de entrada e de saída associados conhecidos, é conhecido como um "epoch". Para garantir a convergência do processo de aprendizado, a cada epoch, a ordem em que os pares de vetores devem ser escolhidos para o treinamento deve ser aleatória. Uma rede BP necessita "aprender", usando os mesmos dados, durante algumas centenas e às vezes milhares de "epochs" para refinar os pesos das conexões. Duas condições de paradas podem ser usadas para interromper o treinamento: um número de epochs máximo tenha sido atingido ou os erros calculados para todos os nós da rede estabilizem, o que estabelece uma convergência.

Após o treinamento, a rede BP tem a habilidade de "lembrar" as características dos dados previamente treinados. Uma rede BP *generalizada* pode ainda exibir excelentes capacidades de predizer pares de vetores de entrada/saída eventualmente desconhecidos. Estes três estágios, aprendizado, lembrança e generalização são essenciais ao modelo de rede neural do tipo "back-propagation".

Apêndice B

Sistema Visual Biológico

A Figura B.1 mostra um esquema do globo ocular humano. Baseando-se nessa Figura, um resumo da funcionalidade do olho é apresentado a seguir. A luz que entra pela pupila é focalizada e invertida pela córnea e lentes, sendo projetada na parte posterior do globo ocular. Nesta parte posterior do olho localiza-se a retina, que converte o sinal luminoso em sinal neurológico. Na retina, as estruturas que recebem primariamente a luz (os fotoreceptores) são os bastões e cones ("rods" e "cones") e as células que transmitem o sinal traduzido em impulsos neurológicos para o cérebro são os gânglios. Os axônios destes últimos formam o nervo ótico, uma rota única pela qual a informação deixa os olhos para os processos de mais alto nível.

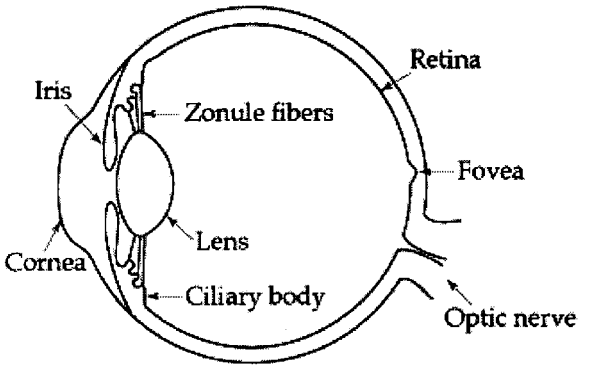


Figura B.1: Anatomia do globo ocular.

A Figura B.2 mostra os caminhos seguidos pela informação após deixar os olhos. Em resumo, a informação que sai do olho pelo nervo ótico passa pelo "chiasm" óptico, onde ocorre um cruzamento parcial dos axônios. Após o "chiasm", os axônios são denominados de trato ótico. O trato ótico passa ao redor da parte central do cérebro para então chegar ao núcleo lateral "geniculate" (LGN), onde todos os axônios devem se juntar. Daí, os axônios do LGN saem para a camada branca mais profunda do cérebro, em forma de radiação ótica que irá se propagar finalmente em direção ao

córtex visual, localizado na parte posterior do cérebro.

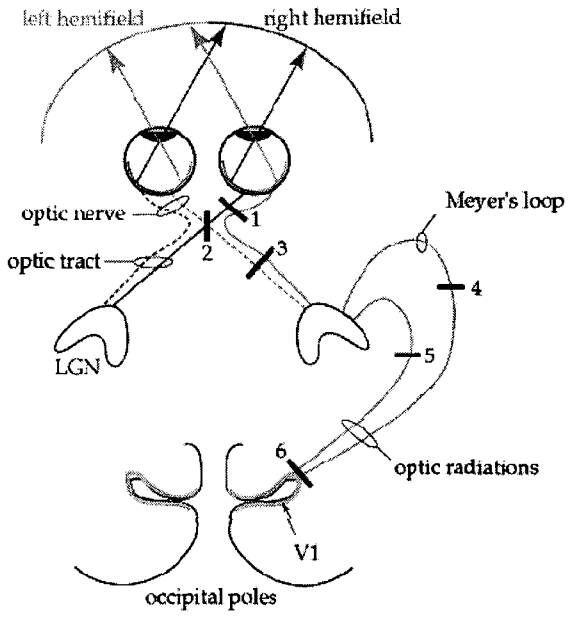


Figura B.2: Caminho do fluxo de informação visual a partir dos olhos até o córtex visual.

No restante deste Apêndice, as estruturas e funções apresentadas nas Figuras B.1 e B.2 serão melhor detalhadas.

B.1 Anatomia e Funções do Olho

O cristalino é na realidade um conjunto de lentes formando uma estrutura naturalmente elástica. Se esta estrutura fosse solta poderia se acomodar tomando uma forma mais esférica. Porém, sob condições normais, um conjunto de fibras radiais ("zonule fibers") mantém as lentes numa forma mais parecida com um disco. Esta forma facilita focalizar objetos distantes. Para focalizar objetos próximos, o corpo ciliar, uma estrutura que suporta as fibras radiais é acionado. Quando o músculo ciliar contrai, as fibras se estendem e as lentes são liberadas de sua tensão normal, podendo buscar uma posição mais esférica (isto é necessário para focalizar um objeto próximo). Este processo de ajuste do foco é denominado de acomodação.

A iris, parte colorida dos olhos, age como um diafragma, podendo ser fechado ou aberto para regular a quantidade de luz que deve entrar na cavidade ocular. No centro da iris localiza-se a pupila dos olhos que age como um orifício, reagindo à intensidade da luz, abrindo-se conforme seja menor a ordem de grandeza da iluminação. Com luz brilhante ou intensa, ela possui approximadamente 2 mm de diâmetro e age como um filtro passa-baixa (para a luz verde), com uma banda

passante de aproximadamente 60 ciclos por grau.

Os sensores fotoreceptores localizam-se na parte da retina voltada para o interior da cavidade ocular. Existem basicamente dois tipos de sensores: os bastões e os cones. Os bastões, em número de aproximadamente 100 milhões, são relativamente longos e finos e provêem a denominada visão escotóptica: a resposta visual às 5 ou 6 ordens de magnitude mais baixas do intervalo de iluminação. (O intervalo de iluminação sobre o qual o sistema visual humano pode operar é aproximadamente de ordem dez de magnitude, ou seja, de 1 a 10^{10}.) Os cones, numa quantidade bem menor de aproximadamente 6 milhoes, são menores e mais grossos e são menos sensitivos do que os bastões. Eles provêem a denominada visão fotóptica: a resposta visual às 5 ou 6 ordens de magnitude mais altas do intervalo de iluminação. Na região intermediária, ambos bastões e cones são ativos e provêem a denominada visão mesóptica. Geralmente, estudos envolvendo sistemas visuais computacionais concentram um maior interesse na visão fotóptica, uma vez que os dispositivos eletrônicos para captura e visualização de imagens apresentam boas condições de iluminação. Os cones são também responsáveis pela visão colorida. A sua distribuição espacial é densa na parte central da retina (fóvea), a uma densidade de aproximadamente 120 cones por arco de grau subtendido no campo de visão. Este espacamento corresponde a um arco de aproximadamente 30 segundos ou a 2 μm. Nesta região da fóvea, a razão entre as células gânglios e os fotoreceptores é de 2 : 1.

A densidade dos cones cai rapidamente quando se afasta para as regiões periféricas à fóvea, a partir de um círculo de 1 grau de raio. Em adição, na fóvea, todos os outros tipos de célula deixam de existir para permitir a passagem máxima de luz aos cones. Isto torna a fóvea visível ao microscópio. Os vasos sanguíneos também se mantêm numa área longe das margens da fóvea. A região interior e ao redor da fóvea possui uma pigmentação amarelo claro, que é visivel através de um oftalmoscópio, chamada de mácula. Os cones são lateralmente conectados por células horizontais e sua conexão frontal ocorre através de células bipolares. Estas são conectadas às células gânglios, que se juntam para formar o nervo ótico que, por sua vez, provê a comunicação com o sistema nervoso central, levando as informações ao cérebro, através dos estímulos provocados pela luz filtrada que incide na retina.

A retina é composta por sete camadas alternadas envolvendo células e outras estruturas para transduzir e transferir a informação. Em geral, as camadas mais escuras (nucléolos) contém corpos celulares enquanto que as camadas mais claras (plexiformes) contém axônios e dendritos. A luz que entra pela pupila incide primariamente nos cones e bastões numa camada denominada de GCL. Os segmentos externos destes transduzem a luz e enviam o sinal transduzido através de uma outra camada celular da retina denominada de ONL e seus axônios. Na outra camada denominada de OPL, os axônios dos fotoreceptores entram em contato com os

dendritos de células bipolares e de células horizontais. As células horizontais são inter-neurônios que ajudam no processamento do sinal. As células bipolares de uma outra camada (INL) processam a saída provinda dos fotoreceptores e das células horizontais e transmitem o sinal aos seus axônios. Numa outra camada denominada de IPL, os axônios das células bipolares anteriores entram em contato com os dendritos das células gânglios e das células amacrinas (outra classe de inter-neurônio). Finalmente, as células gânglios da camada GCL enviam seus axônios através da camada OFL para o disco óptico dando origem ao nervo ótico. Estes axônios levarão a informação ao LGN. Todos os axônios das células gânglios saem do globo ocular numa única localização: o disco ótico. No disco ótico, não existem os foto-receptores e as células de acesso a estes, isto para que os axônios do nervo ótico possam penetrar no "coroid" e no "sclera". Isto cria um buraco em nossa visão, denominado de ponto cego. Normalmente, cada olho cobre o ponto cego do outro, e também o cérebeo completa a informação faltando com o mesmo padrão da sua vizinhança. Assim, o ponto cego nos é imperceptivel.

B.2 Caminhos Visuais Básicos

A informação sobre o ambiente entra em ambos os olhos com uma larga superposição. Considere um corte vertical no campo visual de cada olho passando pela fóvea e dividindo-o em dois. A metade da retina mais próxima da têmpora é denominada de retina temporal e a mais próxima do nariz é denominada de retina nasal. Como as imagens são invertidas ao passar pelas lentes, no olho direito, a retina nasal vê na realidade a metade direita do ambiente enquanto que a retina temporal vê a metade esquerda do ambiente. Note também que a retina nasal direita e a retina temporal esquerda vêem exatamente a mesma coisa numa região mais ou menos próxima da fóvea. A metade do ambiente vista por estas duas últimas é definida como o hemi-campo de vista direito, enquanto que a outra metade é definida como o hemi-campo esquerdo.

Cada olho adquire informação de ambos hemi-campos. Para cada objeto que nossos olhos vêem, geralmente ambos os olhos o estão vendo (isto é essensial para a reconstrução estéreo), mas as duas imagens formadas estarão sendo projetadas uma na retina nasal e a outra na retina temporal. Como a parte esquerda do cérebro controla a parte direita do corpo, a informação que realmente interessa à primeira é a relativa ao hemi-campo direito. As fibras óticas nervosas que saem da retina são organizados de tal forma que os hemi-campos são separados. Mais especificamente, as fibras provenientes das retinas nasais se cruzam no "chiasm" ótico, enquanto que as retinas temporais, já posicionadas para ver o lado oposto do ambiente, não se cruzam. As consequências práticas deste cruzamento são tais que

cortes ou destruições totais das fibras nervosas ocorridas antes do "chiasm" afetará apenas um dos olhos em ambos hemi-campos (como se fechássemos um olho). Se a destruição ocorre após o "chiasm", esta afetará visão em ambos os olhos, mas apenas em um hemi-campo. Isto não é fácil de visualizar. O campo de vista seria de apenas 90 graus, a partir da fóvea para um dos lados.

Como já foi citado, após a separação das fibras nervosas no "chiasm", elas desviam-se dos pedúnculos que compõem a parte central do cérebro e atingem o LGN. Este corpúsculo realmente faz parte do tálamo (nada passa ao córtex cerebral sem que tenha havido antes uma sinapse no tálamo). Assim, quase todos os axonios do trato ótico fazem uma sinapse no LGN. Os axonios restantes fazem uma sinapse nos nucléolos do cerebro central: o culículo superior e a área "pretectal". Esta última área está relacionada com as funções de contração e expansão da pupila em reação à luz. A palavra "Geniculate" significa "em forma de joelho" e esta é exatamente a descrição da forma do LGN. As estrias internas ao LGN, notadas se este for cortado, são camadas e haverá 6 delas em quase todas as partes do órgão. Cada camada recebe entrada de um olho diferente: 3 para o olho esquerdo e 3 para o direito, ou seja, um hemi-campo completo é projetado aqui. Estas camadas se alternam, assim, os axônios de apenas um dos olhos aparecem em estrias alternadas, como pode ser visto na Figura B.3.

Figura B.3: Axônios do LGN vistos em camadas separadas para cada olho.

Um segundo aspecto a ser notado na organização do LGN é que as 4 camadas de fora são compostas de células pequenas, e correspondentemente, recebem informação das células gânglio pequenas da retina. Estas camadas são chamadas de parvo-celulares e podem ser vistas na Figura B.4. Por outro lado, as camadas magno-celulares são compostas de células grandes e recebem informação de células gânglio também grandes.

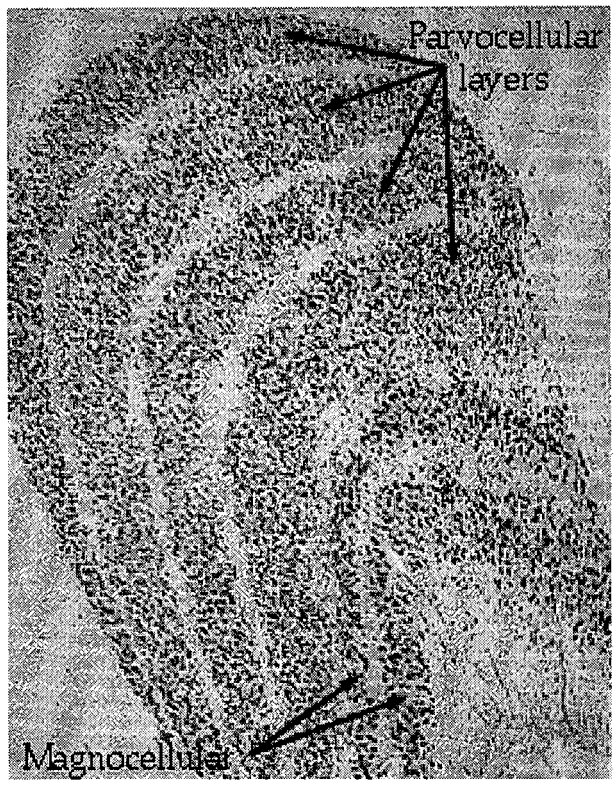


Figura B.4: Camadas parvo-celulares e magno-celulares do LGN.

Os neurônios do LGN enviam seus axônios, através das radiações óticas, diretamente à área V1, também conhecida como córtex visual primário, ou córtex "striate", ou ainda área 17. Este caminho passa através da matéria branca dos lobos temporal e parietal. Uma vez que os axônios encontram a área V1, eles terminam primariamente numa sub-camada simples do córtex visual. O córtex visual (tal qual as outras áreas corticais do cerebro) é composto de 6 camadas básicas. As camadas da área V1 foram organizadas esquematicamente de forma primária. A camada 4 foi expandida em 4 sub-camadas: 4A, 4B, 4Ca e 4Cb. A camada 4A é uma camada escura, enquanto que a mais profunda 4B é uma camada muito mais clara. A camada 4B é visível sem uso de microscópio: na Figura B.5 ela é a linha de Gennari, uma estria branca que dá à camada V1 o seu outro nome, o córtex "striate". A camada 4C é importante porque ela recebe muita da informação do LGN. Devido a essas especializações, as transições entre estas áreas (denominadas de áreas

de Brodmann) podem ser vistas. Basta, na Figura B.5, seguir ao longo da estria (4B) e esta desaparecerá subitamente numa camada mais compacta, a camada 4. Esta é a transição entre as áreas V1 e V2, denominadas de córtex visual secundário ou de área 18.

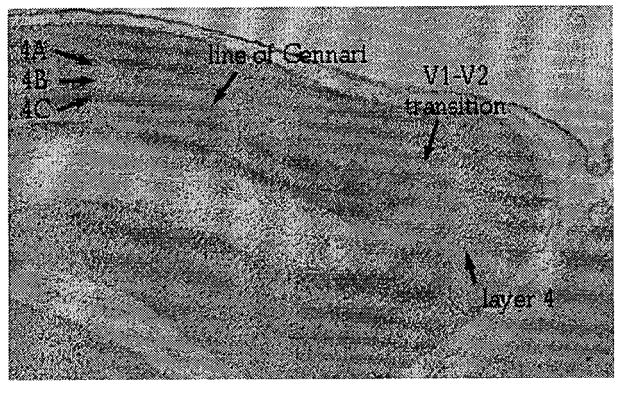


Figura B.5: Organização do córtex visual em camadas.

B.2.1 Dominância Ocular

A separação por olho ocorrida no LGN se mantém quando os axônios entram o córtex visual. Cada setor de axônios de cada camada do LGN se espalham em uma fina coluna dentro da área 4C. À medida que o sinal é transmitido às camadas mais altas do córtex, a informação dos dois olhos se junta, e a visão binocular é criada. Mas em 4C, os olhos estão ainda totalmente separados. Um corte tangencial (paralelo à superfície) através da camada 4C, mostra pequenos "pilares" próximos uns dos outros, como formando as listas de um tigre. Estas listas são denominadas listas de dominância ocular (ver Figura B.6).

B.2.2 Forma dos campos receptivos da área V1

Os campos receptivos dos neurôneos das células gânglios e também do LGN são determinados ao redor do centro de cada célula, respondendo otimamente à "pontos de luz". Já no córtex, os campos receptivos respondem otimamente à "barras de luz" com uma determinada orientação (Figura B.7).

As células mais simples do córtex visual possuem um campo receptivo muito próximo desse arranjo. Uma barra de luz próxima do seu centro, a uma orientação precisa irá excitá-la. Outras conexões e convergências em outras áreas do córtex criarão campos receptivos cada vez mais complexos, até que uma simples célula seja responsável pela definição da forma de uma face completa. Note que a informação sobre a face que entrou na retina não é nada mais do que alguns milhões de pontos de luz.

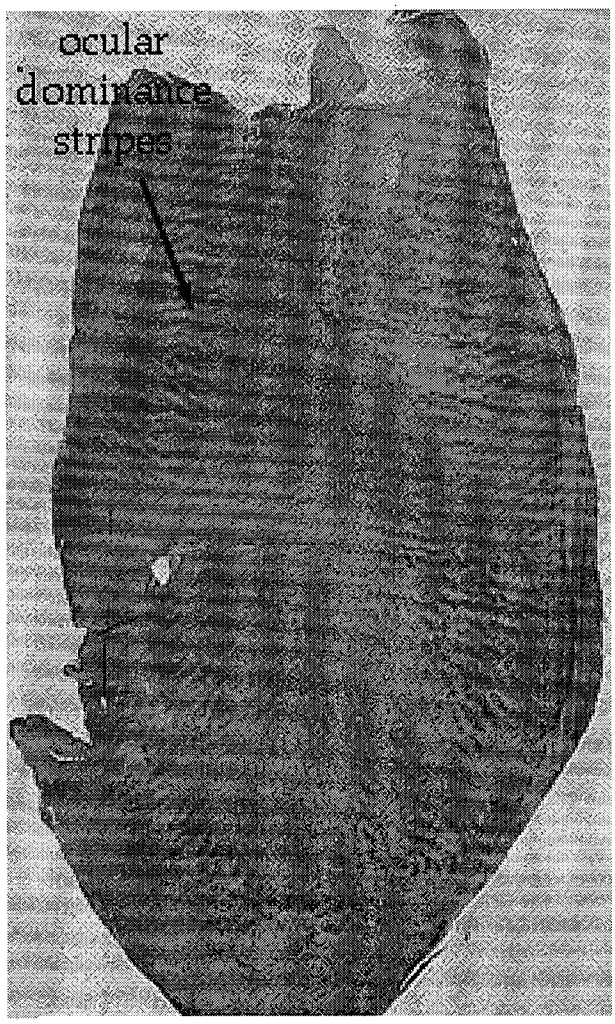


Figura B.6: Listas de dominância ocular notadas no córtex visual.

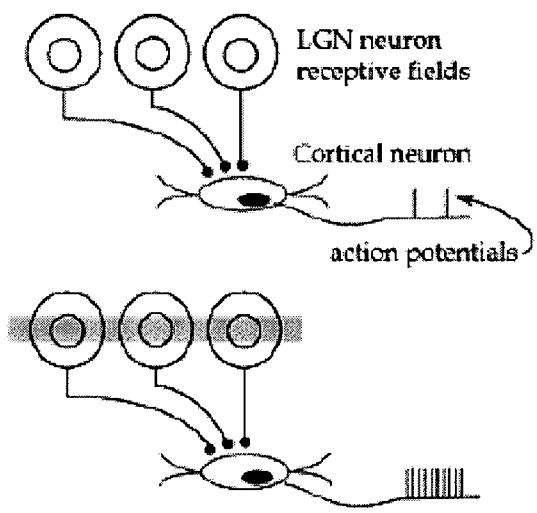


Figura B.7: Arranjo das células do LGN para formar os campos receptivos das células do córtex visual.

B.2.3 Divisão e especializações

Algumas divisões da informação, criando especializações são feitas logo na retina. Os bastões só podem ver em preto e barnco e podem funcionar com luz de fraca intensidade. Já os cones podem ver todas as cores do espectro de distribuição, mas requerem alta intensidade de luz.

Também temos pelo menos dois tipos de células gânglios. Há células gânglios pequenas que são dominantes na região da fóvea, sendo sensitivas a cor e possuindo um campo receptivo é suficientemente pequeno que pode pegar os detalhes em um nível de resoluyção bem alto. Estas são denominada de células P (p de "parvo" ou pequeno). O segundo tipo de células gânglios, possuem vetores de dendritos bem largos, e recebem informação de uma grande região de células bipolares. Este tipo de célula é mais encontrada na periferia da retina, não são sensitivas à cores e possuem um campo receptivo maior, sendo menos susceptivel aos detalhes. Porém, a principal característica é que elas são sensíveis ao movimento, podendo seguir um sinal piscando ao longo de várias células bipolares. Estas células são denominadas de células P (P de "magno" ou grande).

Estes dois tipos de informação (movimento versus cor) são mantidos em separado por todo o caminho visual até o córtex associativo, passando pelo LGN, V1 (nas sub-camadas 4Ca e 4Cb), e após, V2. No final, as áreas corticais parietal (tais como MT e PP) usam esta informação de movimento para lidar com o movimento de objetos, navegação, orientação e localização espacial. As áreas temporais como V4 e IT estão envolvidas com a percepção e o reconhecimento de padrões complexos.

Referências Bibliográficas

ARAUJO, E. & GRUPEN, R. A. 1996 (September). Learning Control Composition in a Complex Environment. *In: Proceedings of Int. Conf. on Simulation of Adaptive Behavior (SAB'96).*

BALLARD, D. H. 1991. Animate Vision. *Artificial Inteligence Journal*, **48**, 57–86.

BALLARD, D. H. 1997. *An Introduction to Natural Computation.* Cambridge, MA: The MIT Press.

BALLARD, D. H. & BROWN, C. M. 1982. *Computer Vision.* Englewood Cliffs, NJ: Prentice-Hall.

BALLARD, Dana H.; HAYHOE, Mary M.; POOK, Polly K. & RAO, Rajesh P. N. 1999. Deictic Codes for the Embodiment of Cognition. *Behavior and Brain Sciences.*

BERTHIER, N. E. 1996a. Infant Reaching Strategies: Theoretical Considerations. *Infant Behavior and Development*, **17**, 521.

BERTHIER, N. E. 1996b. Learning to reach: A mathematical model. *Developmental Psychology*, **32**, 811–823.

BEVERIDGE, J. R.; GRIFFITH, R. K.; HANSON, A. R. & RISEMAN, E. M. 1989. Segmenting Images Using Localized Histograms and Region Merging. *International Journal of Computer Vision*, **2**, 311–347.

BRAUN, H & RIEDMILLER, M. 1993. Rprop: A fast and robust backpropagation learning strategy. *In Proc. of the International Conference on Neural Networks*, 123–134.

CLIFTON, R. K.; ROCHAT, P.; ROBIN, D. J. & BERTHIER, N. E. 1994. Multimodal Perception in the Control of Infant Reaching. *Journal of Experimental Psychology: Human Perception and Performance*, **20**, 876–886.

COELHO, Jefferson A. & GRUPEN, Roderic A. 1997. A Control Basis for Learning Multifingered Grasps. *Journal of Robotic Systems*, **14**(7), 545–557.

COELHO, Paulo S. S.; ESPERANÇA, Claudio & OLIVEIRA, Antonio A. F. 1999. Enhancing the Bayesian Network Approach to Face Detection. *Proceedings of the XIX International Conference of the Chilean Computer Science Society (SCCC '99)*, November.

COHEN, L. D. 1991. On Active Contour Models and Balloons. *CVGIP:Image Understanding*, **53**(2), 211–218.

COLLET, T. S.; CARTWRIGHT, B. A. & SMITH, B. A. 1986. Landmark Learning and Visuo-spatial Memories in Gerbils. *Journal of Comparative Physiology A*, **158**, 835–851.

CONNOLY, C. I. & GRUPEN, R. A. 1993. On the Applications of Harmonic Functions to Robotics. *Journal of Robotic Systems*, **10**(7), 931–946.

DAVIS, L. S.; DEMENTHON, D.; BESTUL, T.; HARWOOD, D.; SRINAVASAN, H. V. & ZIAVRAS, S. 1992. *RAMBO, Vision and Planning on the Connection Machine.* Technical Report. University of Mariland.

FARAH, M. J. 1990. *Visual Agnosia: Disorders of object recognition and what they tell us about normal vision.* Cambridge, Massachusetts: The MIT Press.

FERRELL, C. 1998. *Orientation Behavior Using Registered Topographic Maps.* TR. Massachusetts Institute of technology, Cambridge, MA.

FISCHER, B.; WEBER, H.; BISCALDI, M.; AIPLE, F.; OTTO, P. & STUHR, V. 1993. Separate Populations of Visually Guided Saccades in Humans: Reaction Times and Amplitudes. *Exp-Brain-Res.*, **92**, 528–541.

FLEET, D. J.; WAGNER, H. & HEEGER, D. J. 1997. *Neural Encoding of Binocular Disparity: Energy Models, Position Shifts and Phase Shifts.* Tech. rept. Personal Notes.

FLORACK, Ludvicus M. J. 1993. *The Syntatic Structure of Scalar Images.* Ph.D. thesis, University of Ultrecht, Ultrecht, ES.

FOLEY, James D.; VAN DAM, Andries; FEINER, Steven & HUGHES, John F. 1990. *Computer Graphics: Principles and Practice.* Addison-Wesley Publishing Company.

FREEMAN, R. D. & OHZAWA, I. 1990. On Neurophysiological Organization of Binocular Vision. *Vision Research*, **30**, 1661–1676.

FREEMAN, W. T. & ADELSON, E. H. 1991. The Design and use of Steerable Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(9), 891–906.

GIRALDI, Gilson A. & OLIVEIRA, Antonio A. F. 1999. *Dual and Topologically Adaptable Snakes.* In Preparation.

GONÇALVES, Luiz M. G. & OLIVEIRA, Antonio A. F. 1998. Pipeline Stereo Matching in Binary Images. *XI International Conference on Computer Graphics and Image Processing (SIBGRAPI'98)*, October, 426–433.

GONÇALVES, Luiz M. G. & OLIVEIRA, Antonio A. F. 1999. A reinforcement learnig approach for attentional control based on a multi-modal sensory feedback. *III Workshop on Cybernetic Vision*, February.

GONÇALVES, Luiz M. G.; OLIVEIRA, Antonio A. F. & GRUPEN, Roderic A. 1998a. A Control Architecture for Multi-modal Sensory Integration. *XI International Conference on Computer Graphics and Image Processing (SIBGRAPI'98)*, October, 418–425.

GONÇALVES, Luiz M. G.; OLIVEIRA, Antonio A. F. & GRUPEN, Roderic A. 1998b. A Framework for Attention and Object Categorization Using a Stereo Head Robot. *XII International Symposium on Computer Graphics, Image Processing and Vision (SIBGRAPI'99)*, October, 418–425.

GONÇALVES, Luiz M. G.; GIRALDI, Gilson A.; OLIVEIRA, Antonio A. F. & GRUPEN, Roderic A. 1999a. Learning Policies for Attentional Control. *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '99)*, November.

GONÇALVES, Luiz M. G.; OLIVEIRA, Antonio A. F. & GRUPEN, Roderic A. 1999b. Multi-modal Stereognosis. *III International Conference on Autonomous Agents (Agents '99)*, May.

GONÇALVES, Luiz M. G.; WHEELER, David; OLIVEIRA, Antonio A. F. & GRUPEN, Roderic A. 1999c. Towards a Framework for Robot Cognition. *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '99)*, November.

GONZALES, Rafael C.; WOODS, RICHARD E. (Contributor) & GONZALES, Ralph C. 1992. *Digital Image Processing.* Addison-Wesley Publication Company.

GRIMSON, W. E. L. 1981. *From Images to Surfaces: A Computational Study of the Human Early Visual System.* Cambridge, MA: The MIT Press.

GRUPEN, Roderic A. 1999. A Sensorymotor Approach Approach to Robotics. *UMass Computer Science Technical Notes.* Unpublished.

GUNN, S. R. & NIXON, M. S. 1997. A Robust Snake Implementation: A Dual Active Contour. *IEEE Trans. Pattern Anal. Mach. Intell*, **19**(1), 63–68.

HANSON, A. R.; WEISS, R.; KOHL, C.; KOHL, T. & BEVERIDGE, J. R. 1998. *An Introduction to Computer Vision.* Computer Science Technical Notes, UMass.

HARALICK, Robert M.; STERNBERG, Stanley R. & ZHUANG, Xiatma. 1987. Image Analysis Using Mathematical Morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-9**(4), 532–550.

HORII, A. 1994. *Depth from Defocusing.* Tech. rept. Royal Institute of Technology, Stockholm, Sweden.

HORN, Berthold K. P. 1986. *Robot Vision.* MIT Press.

HUBER, E. & KORTENKAMP, D. 1995. Using Stereo Vision to Pursue Moving Agents with a Mobile Robot. *In: IEEE Conference on Robotics and Automation.*

HUBER, Manfred & GRUPEN, Roderic A. 1997. A Feedback Control Structure for On-line Learning Tasks. *Robotics and Autonomous Systems*, **22**(3-4), 303–315.

HUBER, Manfred; W.S., MACDONALD & A., GRUPEN Roderic. 1996. A Control Basis for Multilegged Walking. *Proceedings of IEEE Conference on Robotics and Automation*, October, 2988–2993.

ITTI, L.; BRAUN, J.; LEE, D. K. & KOCH, C. 1997. A Model of Early Visual Processing. *In: NIPS Int. Conference.*

JAIN, Anil K. 1989. *Fundamentals of Digital Image Processing.* Prentice-Hall.

JULESZ, B. 1971. *Foundations of Cyclopean Perception.* Chicago, IL, USA: University of Chicago Press.

JULESZ, B. & SAARINEN, J. 1991. The Speed of Attentional Shifts in the Visual Field. *Pages 1812–1814 of: Proc. of the National Academy of Sciences of USA*, vol. 88.

KASS, M.; WITKIN, A. & TERZOPOULOS, D. 1988. Snakes: Active Contour Models. *International Journal of Computer Vision*, **1**(4), 321–331.

KOHONEN, Teuvo. 1990. The Self Organizing Map. *Journal of Electrical and Electronics Engineers*, **78**, 1464–1480.

KOLLER, D.; LUONG, Tuan & MALIK, J. 1994. *Binocular Stereopsis and Lane Marker Flow for Vehicle Navigation: Lateral and Longitudinal Contro.* Technical Report. University of California.

KOSSLYN, S.M. 1994. *Image and Brain. The Resolution of the Imagery Debate.* Cambridge, MA: The MIT Press.

LE CUN, Y. 1985. Une Procédure d'Aprentissage pour Réseau à Seuil Assymétrique. *Cognitiva 85: A la Frontiére de l'Intelligence Artificielle des Sciences de la Connaissance des Neurosciences*, 599–604.

LEE, S. & KAY, Y. 1991. A Kallmann Filter for Accurate 3D Motion Estimation from a Sequence of Stereo Images. *CVGIP Image Understanding*, **54**(2), 244–258.

MACDONALD, W. S. 1996. Legged Locomotion over Irregular Terrain using the Control Basis Approach. *Master Thesis.*

MARENGONI, M.; JAYNES, C.; HANSON, A. & RISEMAN, E. 1999 (January). ASCENDER II, A Visual Framework for 3D Reconstruction. *In: International Conference on Computer Vision Systems.*

MARR, D. 1982. *Vision – A Computational Investigation into the Human Representation and Processing of Visual Information.* Cambridge, MA: The MIT Press.

MARR, D. & POGGIO, T. 1979. A Computational Theory of Human Stereo Vision. *Pages 301–328 of: Proc. of the Royal Society of London*, vol. 204.

MATTHIES, Larry & BROWN, E. 1997. *Machine Vision for Obstacle Detection and Ordnance Recognition.* Technical Report. JPL-CIT/NASA and Wright Laboratory.

MATTHIES, Larry & SHAFFER, S. A. 1987. Error Modeling in Stereo Navigation. *IEEE Journal of Robotics and Automation*, **RA-3**(3).

MATTHIES, Larry; KELLY, A. & LITWIN, T. 1995. *Obstacle Detection for Unmanned Ground Vehicles: A Progress Report.* Technical Report. JPL. California Institute of Technology.

MCINERNEY, T. & TERZOPOULOS, D. 1995 (June). Topologically Adaptable Snakes. *Pages 840–845 of: Proc. Of the Fifth Int. Conf. On Computer Vision (ICCV'95), Cambridge, MA, USA.*

MCINERNEY, T. J. 1997. *Topologically Adaptable Deformable Models for Medical Image Analysis.* Ph.D. thesis, Department of Computer Science, University of Toronto.

NEWMAN, Willian M. & SPROULL, Robert F. 1979. *Principles of Intreractive Computer Graphics.* McGraw-Hill Book Company.

NISHIHARA, K. 1984. *Practical Real-Time Stereo Matcher.* AI Lab Technical Report, Optical Engeneering. Massachusetts Institute of Technology.

NISHIHARA, K. 1991. *Minimal Meaningful Measurements Tools.* Technical Report. Teleos Research.

NISHIHARA, K.; THOMAS, H. J. & HUBER, E. 1984. *Real-Time Tracking of People Using Stereo and Motion.* AI Lab Technical Report. Massachusetts Institute of Technology.

PAPOULIS, Athanasius. 1991. *Probability, Random Variables, and Stochastic Processes.* MacGraw-Hill.

PARKER, D. 1985. *Learning Logic.* Technical Report TR 47. Center for Computational Research in Economics and Management Science, Massachusetts Institute of Technology.

PIATER, J.; RAMAMRITHAM, K. & GRUPEN, Roderic A. 1999. Learning Real-Time Stereo Vergence Control. *Proceedings of the 1999 IEEE International Symposium on Intelligent Control (ISIC '99).*

PIATER, Justus H. & GRUPEN, Roderic A. 1999. A Framework for Learning Visual Discrimination. *Proceedings of the 12th International FLAIRS Conference (FLAIRS '99).*

QIAN, N. 1994. Computing Stereo Disparity and Motion with Known Binocular Cell Properties. *Technical Report, Dept of Brain and Cognitive Sciences, MIT, 1993. Also in Neural Computation*, **6**, 390–404.

QIAN, N. & ZHU, Y. 1997. Physiological Computation of Binocular Disparity. *Vision Research*, **37**, 1811–1827.

RAO, Rajesh P. N. & BALLARD, Dana. 1995. An Active Vision Architecture Based on Iconic Representations. *Artificial Intelligence Journal*, **78**, 461–505.

RAVELA, S. & MANMATHA, R. 1997. Retrieving Images by Similarity of Visual Appearance. *Workshop on Content Based Access of Image Databases (with CVPR)*, **2**, 311–347.

RAVELA, S. & MANMATHA, R. 1998. On Computing Global Similarity in Images. *UMass Technical Report.*

REDISH, A. David & TOURETZKY, David S. 1997. Navigating with Landmarks: Computing Goal Locations from Place Codes. *In:* Ikeuchi, K. & Veloso, M. (eds), *Symbolic Visual Learning.* Oxford University Press.

RODRIGUEZ, J. J. & AGGARWAS, J. K. 1990. Stochastic Analysis of Stereo Quantization Error. *IEEE Transactions on Patter Analysis and Machine Intelligence*, **12**(5).

ROGERS, David F. 1985. *Procedural Elements for Computer Graphics.* MacGraw-Hill Book Company.

ROGERS, David F. & ADAMS, J. Alan. 1990. *Mathematical Elements for Computer Graphics.* MacGraw-Hill Book Company.

RUMELHART, D. E.; HINTON, G. E. & WILLIAMS, R. J. 1986. *Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland, editors, Parallel Distributed Processing: Explorations in the microstructure of cognition.* Vol. 1: Foundations. Cambridge, Massachusetts: The MIT Press.

RYBAK, I. A.; GUSAKOVA, V. I.; GOLOVAN, A. V.; PODLADCHIKOVA L, N. & SHEVTSOVA, N. A. 1998. A Model of Attention-Guided Visual Perception and Recognition. *Vision Research.*

SANGER, T. D. 1988. Stereo Disparity Computation using Gabor Filters. *Biology and Cybernetics*, **59**, 405–418.

SCHNACKERTZ, T.J. & GRUPEN, Roderic A. 1995. A Control Basis for Visual Servoing Tasks. *Proceedings of the IEEE Conference on Robotics and Automation*, October.

SIMARD, Patrice Y. 1991. *Learning State Space Dynamics in Recurrent Networks.* Technical Report TR 383 and Ph.D. Thesis. Computer Science Department, University of Rochester.

SOATTO, S.; FREZZA, R. & PERONA, P. 1997. *Motion Estimation via Dynamic Vision.* TR. California Institute of Technology, Pasadena, CA.

SOUCCAR, Kamal; COELHO, Jefferson A. & GRUPEN, Roderic A. 1998. A Control Basis for Haptically-Guided Grasping and Manipulation. *UMass Computer Science Technical Report*, October.

STRERI, A. 1993. *Seeing, Reaching, and touching.* Cambridge, MA: The MIT Press.

SUTTON, Rich S. & BARTO, Andrew G. 1998. *Reinforcement Learning: an Introduction.* Cambridge, MA: The MIT Press.

ULLMAN, Shimon. 1996. *High-level Vision: Object Recognition and Visual Cognition.* Cambridge, Massachusetts: The MIT Press.

VAN DER LAAR, Pierre; HESKES, Tom & GIELEN, Stan. 1995. A Neural Model of Visual Attention. *Neural Networks: Artificial Intelligence and Industrial Applications, eds. Kappen,B. and Gielen,S.*, 111–114.

VAN DER LAAR, Pierre; HESKES, Tom & GIELEN, Stan. 1997. Task-dependent Learning of Attention. *Neural Networks*, **10**(6), 981–992.

VIOLA, Paul A. 1996 (November). *Complex Feature Recognition: A Bayesian Approach for Learning to Recognize Objects.* AI Memo 1591. Massachusetts Institute of Technology.

WATKINS, C. J. C. H. 1989. *Learning from Delayed Rewards.* Ph.D. thesis, King's College, Cambridge, UK.

WATKINS, C. J. C. H. & DAYAN, P. 1992. Technical Note: Q-Learning. *Machine Learning - Special Issue on Reinforcement Learning*, **8(3/4)**, 279–292.

WERBOS, P. 1974. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences.* Ph.D. thesis, Harvard University, Harvard, MA, USA. PhD Thesis Dissertation.

WERBOS, P. 1988. Backpropagation: Past and future. *IEEE International Conference on Neural Networks*, 343–353.

WESSLER, Mike. 1996. *A Modular Visual Tracking System.* AI Lab Technical Report. Massachusetts Institute of Technology.

WESTELIUS, C. J. 1995. *Focus of Attention and Gaze Control for Robot Vision.* Ph.D. thesis, Linköping University, Sweden, S–581 83 Linköping, Sweden. Dissertation No 379, ISBN 91–7871–530–X.

WESTIN, C. F. 1994. *A Tensor Framework for Multidimensional Signal Processing.* Ph.D. thesis, Linköping University, Sweden, S–581 83 Linköping, Sweden. Dissertation No 348, ISBN 91–7871–421–4.

ZHANG, Z. 1993. *Le Probleme de la Mise em Correspondence: L'état de l'art. Rapport de Recherche.* Technical Report. INRIA.