

MAPEAMENTO DE AMBIENTES INTERIORES PARA ROBÔ MÓVEL BASEADO EM MARCOS VISUAIS E REDE NEURAL HIERÁRQUICA

Luciana Leal da Silva, lleals@gmail.com¹
 Mário Luiz Tronco, mariot@ibilce.unesp.br¹
 Henrique Antonio Vian, havian@gmail.com¹
 Giovana Pellinson, giovanaps@gmail.com¹
 Rogéria Cristiane Gratão de Souza rogeria@ibilce.unesp.br¹
 Arthur José Vieira Porto, ajvporto@sc.usp.br²

¹Universidade Estadual Paulista, Instituto de Biociências, Letras e Ciências Exatas – IBILCE. Av. Cristovão Colombo, 2265, CEP: 15054-000 São José do Rio Preto – SP – Brasil

²Escola de Engenharia de São Carlos – Universidade de São Paulo – USP Av. Trabalhador São-Carlense, 400, CEP: 13560-970 São Carlos – SP – Brasil

Resumo: Robôs autônomos precisam ser capazes de aprender e manter modelos de seus ambiente. Neste contexto o presente trabalho contempla técnicas de classificação e extração de características a partir de imagens em conjunto com redes neurais artificiais com o objetivo de usá-las na implementação do sistema de mapamento e localização do robô móvel do Laboratório de Automação e Controle Evolutivo (LACE). Para perceber o ambiente explorado o robô usa seu sistema sensorial formado por sensores de ultra-som e um sistema catadióptrico, composto por uma câmera e um espelho côncico. O sistema de mapeamento é dividido em três módulos. Dois deles serão apresentados neste artigo: módulo classificador de nós e módulo caracterizador de nós. O primeiro utiliza uma rede neural hierárquica para tarefa de classificação. Já o segundo usa técnicas de extração de atributos de imagens e de reconhecimento de padrões invariantes aplicadas em imagens coletadas do ambiente. A rede neural utilizada pelo módulo classificador é estruturada em duas camadas: razão e intuição; a qual é treinada para classificar cada local explorado pelo robô dentre quatro classes pré-definidas. O resultado final da exploração é a construção de um mapa topológico do ambiente. Resultados obtidos através da simulação de ambos os módulos do sistema serão apresentados neste trabalho.

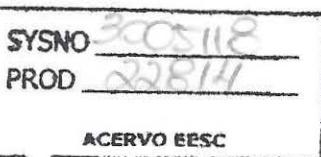
Palavras-chave: mapeamento topológico, sistema de visão omnidirecional, rede neural hierárquica (RNAH), vetor de atributos, padrões invariantes afim.

1. INTRODUÇÃO

A tarefa de navegar no contexto da robótica móvel necessita de métodos que utilizem de maneira mais eficiente as informações perceptuais sobre o ambiente externo. Tais informações afetam diretamente o grau de autonomia do robô, a qual é extremamente necessária no contexto de modelagem de ambientes.

O processo de modelar um ambiente consiste na criação de um modelo que represente sua estrutura física, ou seja, na construção de um mapa. Os primeiros esforços em navegação baseada em mapas foram principalmente inspirados nos processos cognitivos dos seres humanos, assumindo-se que os erros de sensores e atuadores pudessem ser detectados e corrigidos por um processo de nível mais alto, ou modificando-se o ambiente de tal maneira que tornasse a navegação mais fácil. Navegação baseada em mapas requer um processo de reconhecimento e análise de alto nível a fim de interpretar o mapa e estabelecer sua correspondência com o mundo real. Além disso, o problema de adquirir modelos é complexo e difícil de ser resolvido devido a limitações práticas sobre a habilidade dos robôs de aprender e usar modelos precisos. Estas limitações surgem a partir de problemas inerentes aos próprios sensores e ao ambiente.

Classicamente, a construção de mapas segue duas abordagens: geométrica e topológica. A abordagem adotada neste trabalho é a topológica, a qual é definida por Kuipers (Kuipers, 1991). De acordo com ela, o ambiente é representado por um número de lugares distintivos, e de modo que o robô possa se locomover de um lugar a outro. Em outras palavras, o ambiente é representado por um grafo, onde os lugares identificados formam o conjunto de vértices ou nós, e todas as passagens entre cada nó formam o conjunto de arestas.



Neste trabalho apresentamos um método de mapeamento usado pelo robô móvel do Laboratório de Automação e Controle Evolutivo (LACE). A Figura 1 mostra uma visão frontal do robô. O método de modelagem proposto usa o seu sistema sensorial embarcado para coletar informações sobre o ambiente, o qual é formado por sensores de ultra-som e um sistema de visão omnidirecional. O método utiliza uma rede neural artificial hierárquica (RNAH) treinada para reconhecer e classificar os lugares usando para isso informações sobre a vizinhança do robô. Dessa maneira, o objetivo é mapear os espaços livres pelos quais o robô possa se locomover, bem como as relações de adjacência entre eles, classificando-os dentre quatro classes pré-definidas, e finalmente criar um mapa topológico do ambiente explorado.



Figura 1. Foto do robô móvel do LACE.

Além da classificação e definição dos nós do mapa, faz-se necessário caracterizá-los, ou seja, identificar características que os tornem únicos e diferentes dos demais nós da mesma classe. Tais características são extraídas de um conjunto de imagens coletadas em cada nó, criando assim marcos naturais que os identifica. Navegação baseada em marcos está amplamente difundida na literatura da área e pesquisas envolvendo tais técnicas têm obtido bons resultados. Como exemplo, podemos citar os trabalhos definidos em (Arsênio, 1998; Betke, 1997; Marsland, 2001 e Se, 2002).

2. ESTRUTURA DO ARTIGO

Vários aspectos do uso de redes neurais artificiais e reconhecimento de padrões invariantes são apresentados na Seção 3. A Seção 4 aborda o procedimento geral adotado para construção do mapa topológico. A Seção 5 apresenta o sistema de visão omnidirecional embarcado no robô móvel do LACE, o qual é utilizado para coletar cenas do ambiente a ser mapeado. A Seção 6 descreve a rede neural hierárquica usada como classificador dos lugares explorados pelo robô. Na Seção 7 define-se o método proposto para selecionar e reconhecer marcos naturais no ambiente de navegação. Os resultados experimentais e conclusões obtidas a partir dos testes serão apresentados nas Seções 8 e 9, respectivamente.

3. TRABALHOS RELACIONADOS

Máquinas com a capacidade de emular funções do cérebro humano têm uma alta capacidade de processamento paralelo, além de serem capazes de aprender a resolver problemas e generalizar soluções não aprendidas. Tais atributos são muito atraentes em aplicações de robótica móvel e tem sido objeto de estudos de muitas pesquisas nesta área.

Matsumoto (Matsumoto et. al, 1999) propôs uma técnica que usa uma seqüência de imagens como um mapa do ambiente. As imagens são adquiridas numa etapa de aprendizagem de uma rede neural, seguindo uma rota pré-determinada e coletadas seguindo espaçamentos constantes. Após a fase de treinamento da rede, o mapa construído pode ser usado durante a navegação do robô para tarefas de localização e planejamento de trajetórias. A localização do robô é realizada através da classificação das imagens coletadas durante a navegação.

Ster (Ster, 2004) apresenta uma abordagem para aprender mapas topológicos, a qual usa redes neurais recorrentes. Esta técnica segue o paradigma de navegação reativa, ou seja, o robô navega baseado em comportamentos, os quais consistem em atitudes do robô frente à dinâmica do ambiente. Além disso, alguns critérios comportamentais são pré-definidos nos chamados pontos de decisão, de modo que à cada ponto, ou local definido no mapa, associa-se um conjunto de comportamentos. Estes são aprendidos pelo robô usando-se uma técnica de aprendizagem chamada de *reinforcement learning*, em conjunto com um novo método com base psicológica proposto pelo autor.

Em sistemas de navegação baseados em marcos, as percepções sensoriais são usadas para mapear o ambiente. Marsland (Marsland, 2001) descreveu um método que automatiza o processo de seleção de bons marcos sob o ponto-de-vista do robô, usando para isto uma rede neural capaz de aprender um modelo de relacionamento entre uma seqüência de percepções sensoriais obtidas durante a fase de treinamento da RNA. Zitová (Zitová, 1999) apresenta em seu trabalho um sistema de reconhecimento de marcos artificiais usado para navegação de robôs móveis, o qual aplica técnicas de reconhecimento de padrões invariantes afim.



4. SISTEMA DE VISÃO OMNIDIRECIONAL DO ROBÔ MÓVEL DO LACE

Nesta seção apresentamos o sistema de visão omnidirecional embarcado no robô móvel do LACE utilizado para captura de cenas do ambiente explorado. A Figura 2 ilustra o procedimento de captura e pré-processamento das imagens.

As imagens omnidirecionais são capturadas por uma câmera com seu foco alinhado a um espelho côncico. Estas imagens são retificadas no módulo de visão omnidirecional, e transformadas em imagens panorâmicas da cena. Elas são então pré-processadas e usadas pelos procedimentos de classificação e caracterização de nós. As equações e a metodologia adotada na captura e processamento das imagens estão definidas em (Spacek, 2003). As Figuras 3 e 4 ilustram exemplos de imagens omnidirecional e panorâmica, respectivamente.

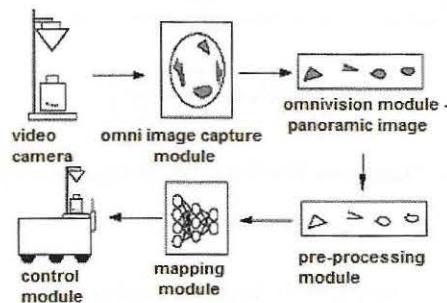


Figura 2. Fases do processamento das imagens coletadas pelo sistema de visão omnidirecional do robô móvel do LACE.



Figura 3. Imagem omnidirecional.



Figura 3. Imagem panorâmica obtida a partir de uma imagem omnidirecional.

5. CONSTRUÇÃO DO MAPA TOPOLOGICO

O método de mapeamento de ambientes proposto neste trabalho tem como objetivo construir um modelo topológico de ambientes interiores explorados pelo robô móvel do LACE. Para perceber o ambiente o robô utiliza seu sistema sensorial formado por sensores de ultra-som e um sistema de visão omnidirecional, sendo tais informações fornecidas como dados de entrada ao sistema de mapeamento. Este sistema é composto por três módulos: classificador, identificador/criador e caracterizador de nós. A principal tarefa do módulo classificador é identificar cada lugar visitado pelo robô, classificando-os dentre quatro classes pré-definidas, e assim criar os nós do mapa com suas respectivas relações de adjacência.

Uma rede neural hierárquica estruturada em duas camadas, *razão* e *intuição*, é usada como principal ferramenta do módulo classificador. Sua estrutura e procedimento de classificação serão descritos na Seção 6. A tarefa de identificação e criação de um novo nó é realizada por um procedimento que recebe a informação sobre a classe do lugar e cria um novo nó da mesma classe. A Figura 2 ilustra a relação entre os módulos do sistema de mapeamento.

Além de identificar e classificar nós, é necessário caracterizá-los, ou seja, cada nó definido precisa ser identificado de maneira única e diferente dos nós pertencentes à mesma classe. O módulo caracterizador de nós é responsável por esta tarefa. Sua função é selecionar bons marcos do ponto-de-vista do robô, usando para isto as imagens dos nós.

Marcos podem ser definidos como objetos da cena que podem ser encontrados de uma maneira distintiva pelo robô. Neste trabalho, marcos naturais serão definidos usando-se técnicas de extração de atributos e reconhecimento de padrões invariantes extraídos a partir de um conjunto de imagens dos nós. O procedimento adotado para implementação deste módulo será descrito na Seção 7.

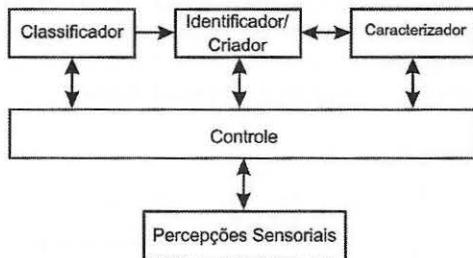


Figura 4. Diagrama funcional do sistema de mapeamento.

6. MÓDULO CLASSIFICADOR

A função do módulo *classificador* é classificar os lugares explorados pelo robô dentre quatro classes pré-definidas: *corredor*, *porta*, *intersecção* e *sala*; e assim passar esta informação ao módulo *criador*, o qual efetivamente cria e insere um novo nó no mapa topológico. O objetivo da classificação é atribuir parâmetros a cada classe para que, com base neles, o robô saiba quais decisões tomar durante a navegação. Dessa maneira, as decisões de navegação do robô podem ser definidas como uma máquina de estados, onde cada estado (classe) está associado a um conjunto de possíveis ações. A Figura 5 ilustra este procedimento.

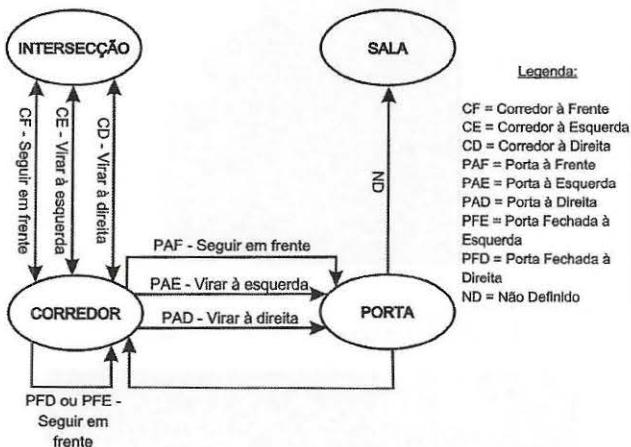


Figura 5. Diagrama em máquina de estados das decisões de navegação do robô.

Analizando a Fig. 5 pode-se constatar que uma vez que o mapa proposto esteja disponível, basta ao robô identificar a classe do local atual para então decidir quais decisões tomar. Nesse caso, tendo um plano de percurso definido, o robô pode então escolher o próximo nó do mapa a ser ocupado a fim de alcançar seu objetivo. Dessa maneira, o sistema de classificação de lugares pode ser usado tanto para propósitos de mapeamento quanto para localização do robô durante sua navegação.

Para implementar a tarefa de classificação usamos uma Rede Neural Artificial Hierárquica (RNAH) treinada para reconhecer as classes de lugares definidas. A estrutura da RNAH bem como seu procedimento de classificação será descrito na próxima seção.

6.1. Rede Neural Hierárquica - RNAH

Nesta Seção descrevemos a estrutura da rede neural hierárquica utilizada para tarefas de classificação dos lugares explorados pelo robô. A RNAH é formada por duas camadas, *razão* e *intuição*, como ilustrado na Fig. 6.

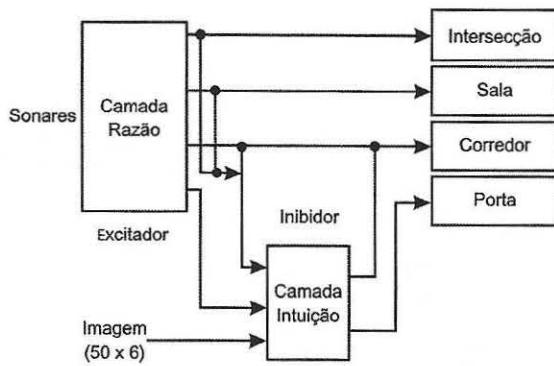


Figura 6. Estrutura da Rede Neural Hierárquica – RNAH.

A RNAH recebe como entrada os dados lidos pelo sistema sensorial do robô: medidas de distância e imagens do ambiente fornecidas pelos sonares e sistema de visão omnidirecional, respectivamente. A rede razão recebe os dados lidos pelos sonares e com base neles é capaz de identificar a classe *corredor*, quando o robô navega através deste, e a classe *intersecção*, quando detecta o encontro de dois ou mais corredores de modo que o robô possa eventualmente mudar a direção do seu percurso. As informações de distância são suficientes para identificar as referidas classes nas condições descritas. Os parâmetros necessários para definição da classe sala estão em estudo, de modo que a mesma não foi incluída nos testes apresentados neste trabalho. Quando o robô se locomover através de um corredor e seus sonares detectarem a existência de uma abertura lateral, esta pode consistir de uma porta ou um corredor. Neste caso as informações de distância não são suficientes para identificar a classe correta. É neste momento que a rede intuição é ativada.

Na situação descrita anteriormente um dos neurônios da saída da rede razão é ativado, o qual consiste no neurônio excitador da rede *intuição*. Esta executa então seu procedimento de classificação utilizando imagens do local atual com o objetivo de identificar a classe correta: *porta* ou *corredor*. Para isso, a segunda camada da RNAH é treinada com imagens destas duas classes de lugares para adquirir a capacidade de distingui-las.

O fato descrito no parágrafo anterior é o que justifica a criação de uma rede neural hierárquica para propósitos de classificação. Os testes envolvendo a RNAH foram realizados e serão descritos na Seção 8.

6.2. Treinamento da RNAH

Para identificar as classes definidas, ambas as camadas da RNAH precisam ser treinadas para adquirir tal capacidade. Definimos então parâmetros para cada classe, os quais as caracterizam tornando-as diferentes das demais. Tais parâmetros são levados em consideração durante a construção do conjunto de padrões de treinamento.

O conjunto de treinamento fornecido à camada *razão* da RNAH permite que a mesma classifique com certo grau de certeza as classes cujos parâmetros que as caracterizam são bem definidos e modelados com base nas distâncias lidas pelos sonares. As classes onde isso não acontece são caracterizadas por suas imagens, e é responsabilidade da segunda rede identificá-las, a qual é ativada sempre que houver qualquer dúvida no processo de reconhecimento da primeira camada.

Por exemplo, um corredor é definido como um lugar limitado continuamente por obstáculos em ambos os lados do robô. Uma intersecção é um local onde dois ou mais corredores se interceptam, de modo que o robô possa eventualmente alterar a direção do seu percurso. Essas duas classes são então modeladas levando-se em consideração tais características e as informações disponíveis para a construção dos modelos: leitura das distâncias dos sonares. No caso da rede intuição, o conjunto de treinamento é formado por diferentes imagens de portas e corredores, com o objetivo de capacitar a rede a distingui-las quando estes forem encontrados durante a etapa de mapeamento.

A arquitetura da rede neural adotada é a Perceptron Multicamadas (MPL), que são redes adequadas para a tarefa de classificação. Determinar o número de camadas ocultas e a quantidade de neurônios em cada uma é uma importante decisão de projeto determinada empiricamente durante a fase de testes, levando-se em consideração os resultados obtidos.

Os sonares localizados nas laterais do robô se movem na horizontal enquanto realizam a leitura das distâncias. O resultado é a criação de um vetor de números reais, onde cada número representa a distância do robô em relação a um objeto, calculada de acordo com o ângulo formado entre a respectiva direção de leitura e a posição central de leitura, seguindo o procedimento ilustrado na Fig. 7. Já o sensor dianteiro se move tanto na horizontal quanto na vertical, resultando em uma matriz de números reais. Isto é necessário para garantir que o robô detecte a altura do espaço livre à sua frente, evitando possíveis colisões. Dessa maneira, cada neurônio da camada de entrada da rede *razão* recebe o valor armazenado em uma das posições do vetor ou matriz de leituras. Assim esta camada tem tantos neurônios quanto o número de leituras realizadas pelos três sonares. Neste caso, este número pode ser modificado, adequando-o às necessidades do sistema de modo a torná-lo tanto confiável quanto eficiente. Sua camada de saída é composta por quatro neurônios: os três primeiros correspondem às classes *sala*, *intersecção* e *corredor*, enquanto o último consiste no neurônio excitador da rede *intuição*.

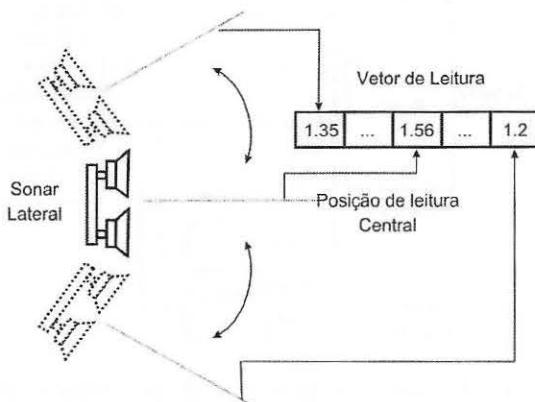


Figura 7. Procedimento de leitura dos sonares laterais do robô.

Como já citado, a rede *intuição* recebe como entrada imagens dos locais a serem classificados. Dessa maneira, o número de neurônios de sua camada de entrada é igual à resolução das imagens, ou seja, o número total de pixels utilizados para representar as imagens. Este número deve ser "grande" o suficiente para garantir a qualidade no processo de classificação, e "pequeno" o suficiente para garantir eficiência durante as fases de treinamento e validação da rede, etapas estas que demandam o maior tempo de execução.

7. MÓDULO CARACTERIZADOR

Nesta seção descrevemos a técnica proposta para implementar o módulo *caracterizador* de nós do sistema de mapeamento. A função deste módulo é caracterizar os nós identificados pelo módulo classificador, a fim de torná-los únicos e distintos dos demais nós pertencentes à mesma classe. O objetivo é criar marcos naturais para os nós a partir de cenas de um conjunto de imagens, as quais serão usadas como marcos visuais, o que difere da abordagem onde marcos são definidos como objetos individuais da cena.

Em nossa abordagem, os marcos são definidos por vetores de atributos e invariantes de momento afim extraídos a partir das imagens dos nós. A técnica usada para extrair os vetores de atributos e diminuir o tamanho dos mesmos é a PCA (Principal Component Analysis) (Duda, 1973), cuja aplicação resulta em uma representação de menor dimensão da imagem, levando em consideração a variância dos atributos. As equações usadas para calcular os momentos afins foram derivadas por Zitovà (Zitovà, 1999), as quais são invariantes sobre transformações afins geral.

O procedimento descrito no parágrafo anterior é responsável pela seleção de marcos naturais (vetores de atributos e momentos afins) em cada nó do mapa. Dessa maneira, durante a criação do mapa, os marcos em cada nó são usados para treinar uma rede neural, criando assim um classificador de marcos naturais usado para tarefas de mapeamento e localização.

7.1. Seleção de Marcos Visuais - Vetores de Atributos

A abordagem proposta para implementação da PCA é baseada no trabalho descrito por Martinez (Martinez, 2005), e é definida como segue.

Considere a imagem I fornecida pelo sistema de visão, representada por uma matriz de dimensão $m \times n$, de modo que cada elemento representa o nível de intensidade de cinza naquele ponto. A imagem pode ser representada por um vetor através da leitura coluna a coluna da matriz da imagem e armazenando cada pixel num vetor coluna. Assim,

$$x(l) = I(i, j), \quad (1)$$

para $i = 1, \dots, n$, $j = 1, \dots, m$ e $l = i + (j \times l) \times m$. O tamanho do vetor de atributos é dado por $d = m \times n$.

Considere t padrões de treinamento conhecidos, x_1, x_2, \dots, x_t . O conjunto de treinamento pode ser visto como uma matriz, onde cada coluna contém um padrão de treinamento,

$$X = \begin{bmatrix} x_1(1) & x_2(1) & \mathbf{L} & x_t(1) \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ x_1(d) & x_2(d) & \mathbf{L} & x_t(d) \end{bmatrix} \quad (2)$$

A matriz de covariância Σ da matriz de treinamento X , pode ser definida como

$$\sum x = (X - \mu)(X - \mu)^T, \quad (3)$$

onde μ é a matriz média de dimensão $d \times t$, e cada coluna da matriz contém o valor esperado do padrão X .

$$\mu(i, j) = \frac{1}{t} \sum_{j=1}^t (X(i, j)), \quad (4)$$

para $i = 1, \dots, d$ e $j = 1, \dots, t$.

Dada a matriz de covariância $\sum x$, estima-se seus auto-vetores e auto-valores de maneira que:
 λ_i and v_i , para $i = 1, \dots, d$, são os auto-valores e auto-vetores de $\sum x$, respectivamente.

Os autovalores indicam a relevância dos autovetores. No caso da PCA, se um dado autovetor tem um grande autovalor, isto significa que este autovetor está em uma direção com grande variância nos padrões. Assim, os autovetores são dispostos em ordem decrescente de autovalores.

O conjunto ordenado de autovetores compõe a matriz de transformação H abaixo,

$$H = [e_1, e_2, \mathbf{K}, e_d]. \quad (5)$$

o vetor de atributos é obtido através de uma mudança de base, onde a matriz H é a matriz de mudança de base

$$Y = H^T X, \quad (6)$$

e H^T é a matriz transposta de H .

A matriz de novos atributos Y obtida não apresenta redução na sua dimensão, sendo somente uma mudança de base no espaço de atributos. Entretanto, a matriz de covariância H pode ser construída somente com os autovetores que possuem os maiores autovalores. Assim, escolhendo-se k autovalores, $k < d$, a dimensão do vetor de atributos é reduzida para uma representação k -dimensional.

Nas técnicas de reconhecimento de padrões pesquisa-se por características distintivas, ou seja, por vetores de atributos que não tenham covariância entre si. Através da matriz de covariância $\sum x$ é possível checar se existe covariância entre os atributos definidos. Fazendo-se a transformação dos atributos X para os novos atributos Y usando PCA, observa-se que a matriz Y tem média zero e a matriz de covariância $\sum y$ é diagonal, em que os elementos da diagonal principal são os auto-valores de $\sum x$, e os elementos fora da diagonal principal são zero. Isto significa que os elementos da matriz Y não estão correlacionados.

8. RESULTADOS EXPERIMENTAIS

Nesta seção apresentamos os resultados obtidos durante a fases de testes dos procedimentos de seleção e reconhecimento de marcos do módulo caracterizador e de ambas as camadas da RNAH: *razão* e *intuição*.

Para criar as duas camadas da RNAH, bem como a RNA do classificador de marcos, usamos o simulador de redes neurais SNNS (Stuttgart Neural Network Simulator) (Zell, 2000), o qual fornece também suporte para etapas de treinamento, validação e teste.

A camada *razão* criada é uma rede perceptron de camada única com 35 neurônios em sua camada de entrada, cada um dos quais recebe uma das posições do vetor e matriz de leitura dos sonares. A saída desta camada tem quatro neurônios, os três primeiros relativos às classes *intersecção*, *sala* e *corredor*, e o último é o neurônio excitador da segunda rede.

A rede *intuição* foi testada usando perceptrons de camada única e multicamadas. Isso porque os resultados obtidos no primeiro caso foram insatisfatórios, apresentando uma baixa taxa de acertos na classificação. Este problema foi resolvido com o uso de uma perceptron multicamadas, cujos resultados serão apresentados ainda nesta seção.

A camada de entrada da rede *intuição* tem 300 neurônios, cada um dos quais recebe o valor armazenado em um dos pixels da imagem de 50 X 6 de resolução. Sua camada de saída tem dois neurônios relativos às classes *corredor* e *porta*.

O conjunto de treinamento fornecido a rede *razão* tem 95 padrões, o qual foi utilizado para treiná-la para o reconhecimento de corredores e intersecções, sendo estas classes modeladas levando-se em consideração as leituras dos sonares. Já o conjunto de treinamento da camada *intuição* tem 65 imagens de portas e corredores. Durante a fase de testes foram apresentados 24 e 50 padrões à camada *razão* e *intuição*, respectivamente.

Todos os 24 padrões fornecidos à primeira rede eram completamente desconhecidos desta, ou seja, eram situações não apresentadas em etapas anteriores. A rede *intuição* foram apresentados 30 padrões desconhecidos e 20 imagens de portas e corredores conhecidas da rede, vistas, contudo, sob diferentes pontos-de-vista ou sob diferentes condições de iluminação.

A Tabela 1 ilustra o resultado geral da classificação das duas camadas. A coluna 2 indica a taxa geral de acertos. A coluna 3 mostra a taxa de acertos no processo de classificação dos padrões desconhecidos, ou seja, padrões não utilizados durante as etapas de treinamento e validação. Já a última coluna, válida somente para a rede *intuição*, indica a porcentagem de acertos na classificação daqueles padrões conhecidos da rede e que sofreram algum tipo de modificação.

Tabela 1. Classificação geral das camadas razão e intuição.

Camada	Classificação Geral	Padrões Desconhecidos	Padrões Alterados
Razão	100%	100%	
Intuição	94%	90%	100%

A Tabela 2 resume os resultados obtidos pela rede *intuição* no processo de classificação das classes *porta* e *corredor*. Analisando-se a tabela, pode-se concluir que a rede alcançou uma taxa de acertos superior a 90%. Além disso, somente 7% dos padrões corretamente classificados obtiveram um baixo valor de ativação, o qual está dentro do intervalo [0.65, 0.75], o que pode ser considerado aceitável.

Tabela 2. Classificação da camada intuição - classes corredor e porta.

Classe	Classificação Correta	Classificação Incorreta	Classificação Correta com baixo Valor de Ativação
Porta	93%	7%	7%
Corredor	95%	5%	0%

Para implementar o módulo *caracterizador* nós usamos uma RNA treinada com informações sobre os marcos definidos em cada nó. A Figura 8 mostra a estrutura da RNA.

Nos testes utilizamos perceptrons com diferentes números de neurônios na camada oculta. A Tabela 3 resume os resultados obtidos, mostrando o número de neurônios na camada oculta, o Mean Square Error (Braga, 2000), o erro total do conjunto de treinamento e a porcentagem de classificação incorreta.

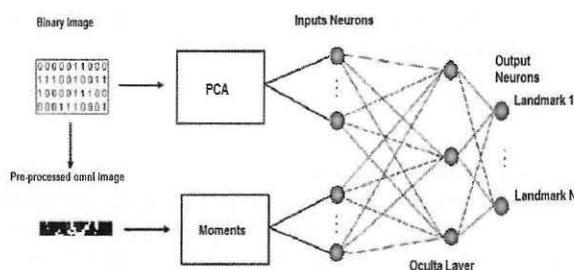


Figura 8. Estrutura da RNA caracterizador de marcos.

Tabela 3. Resultados dos testes da RNA – Caracterizador de marcos.

Neurônios	MSE	Erro	Porcentagem de
			Erro
0	0.1360	5.43981	30%
5	0.52133	20.85314	33,3%
10	0.53058	21.22337	33,3%
20	0.51565	20.62615	46,6%
30	0.51304	20.52151	53,3%

9. CONCLUSÕES

Este trabalho contempla uma técnica de modelagem de ambientes para navegação do robô móvel do Laboratório de Automação e Controle Evolutivo – LACE, o qual usa seu sistema de visão omnidirecional e sensores de ultra-som para adquirir informações sobre o ambiente externo e fornecê-las ao sistema de mapeamento. Este é composto por três módulos, sendo dois deles descritos neste artigo: módulo classificador de nós e módulo caracterizador de nós.

O módulo *classificador* utiliza como ferramenta de classificação uma rede neural hierárquica estruturada em duas camadas: razão e intuição. Ambas as camadas da rede foram construídas usando o simulador de redes neurais SNNS. O uso de uma perceptron de camada única nos testes da rede *razão* apresentou excelentes resultados já que a classificação da mesma alcançou 100% de acertos. Sendo assim, tal arquitetura de rede mostrou-se adequada para a implementação do sistema real do sistema de mapeamento do robô.

Os resultados obtidos no teste da camada *intuição* foram também satisfatórios, pois a mesma obteve em geral uma taxa de acertos de 94%. Assim, esta camada alcançou o objetivo para a qual foi criada: resolver possíveis conflitos no processo de classificação da primeira rede. Além disso, pode-se concluir que a baixa resolução das imagens de entrada da rede, 50 x 60 pixels, foi suficiente para o propósito de classificação.

Sobre o módulo *caracterizador*, pode-se dizer que a estrutura da RNA não está adequada para a tarefa de classificação de marcos, pois a porcentagem de erro neste processo foi maior do que a esperada. Serão realizados alterações na estrutura da RNA ou em alguns parâmetros utilizados na etapa de treinamento com o objetivo de se gerar melhores resultados.

10. REFERÊNCIAS

- Arsênio, A. And Ribeiro, M.I., 1998, "Absolute localization of mobile robots using natural landmarks", IEEE International Conference on Electronics, Circuits and Systems, vol 2, 483-486.
- Betke, M. and Gurvits, L., 1997, "Mobile Robot Localization Using Landmarks", IEEE Transactions on Robotics and Automation", Vol. 13, No. 2.
- Braga, A.; Ludemir, T. B. and Carvalho, A.; Redes neurais artificiais, LTC, 2000.
- Duda, R. O. and Hart, P. E., 1973. "Pattern Classification and Scene Analysis". John Wiley & Sons.
- Kuipers, B. J., Byun, Y. T., 1991. "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations", Robotics and Autonomous Systems, Vol. 8, 47–63.
- Marsland, S., Nehmzow, U. and Duckett, T., 2001, "Learning to select distinctive landmarks for mobile robot navigation", Robotics and Autonomous Systems, Vol. 37, pp. 241 – 260.
- Martinez, V. O. and Costa, A. H. R., 2005. "Reconhecimento de marcos visuais como cenas para localização de robôs móveis".
- Matsumoto, Y., Ikeda, K., Inaba, M. and Inoue, H., 1999. "Visual navigation using omnidirectional view sequence". Proceeding of the IEEE International Conference on Intelligent Robots and Systems, pp. 317-322.
- Se, S., Lowe, D. and Little, J., 2002, "Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks", The International Journal of Robotics Research, Vol. 21, pp. 735 – 758.
- Spacek, L., 2003. "Omnidirectional Catadioptric Vision Sensor with Conical Mirrors", Journal of Robotics e Auonomous System.
- Ster, B., 2004. "An integrated learning approach to environment modeling in mobile robot navigation", Neurocomputing, Vol. 57, pp. 215 – 238.
- Zell, A. et al.; SNNS Stuttgart neural network simulator user manual, Disponível em <http://www.ra.informatik.uniuebingen.de/downloads/SNNS/SNNSv4.2.Manual.pdf>. Acessado em: 7 de março de 2007.
- Zitová, B., Flusser, J., 1999, "Landmark recognition using invariant features", Pattern Recognition Letters, Vol 20, pp. 541-547.