



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

# **Identificação e Rastreamento de Alvos Móveis Utilizando a Rede Neural Kohonen com o Auxílio do Filtro de Kalman e com a Aplicação da Lógica fuzzy**

**RELATÓRIO FINAL DE PROJETO DE INICIAÇÃO CIENTÍFICA  
(PIBIC/CNPq/INPE)**

**Bolsista: Thiago José da Silva (FEG/UNESP)  
thiagogj\_silva@yahoo.com.br**

**Orientador: José Ernesto de Araújo Filho (LIT/INPE)  
ernesto@lit.inpe.br**

**Julho de 2007**

# ÍNDICE

<b>RELATÓRIO FINAL DE PROJETO DE INICIAÇÃO CIENTÍFICA.....</b>	<b>1</b>
<b>ÍNDICE DE FIGURAS .....</b>	<b>3</b>
<b>1 – INTRODUÇÃO .....</b>	<b>4</b>
1.1 – OBJETIVO DO TRABALHO.....	5
1.2 – DESENVOLVIMENTO REALIZADO .....	6
<b>2 - HISTÓRICO DA IDENTIFICAÇÃO DE ALVOS MÓVEIS .....</b>	<b>7</b>
<b>3 - REDES NEURAIS KOHONEN COM O FILTRO DE KALMAN E COM A LÓGICA FUZZY NA IDENTIFICAÇÃO E RASTREAMENTO DE ALVOS MÓVEIS.....</b>	<b>10</b>
3.1 – INTRODUÇÃO TEÓRICA .....	10
3.1.1 – Rede Neural Kohonen.....	10
3.1.2 – Filtro de Kalman.....	13
3.1.3 – Descrição Semântica .....	14
3.2 – ACOPLAGEM DOS CONCEITOS .....	17
3.2.1 – União da Rede Neural Kohonen com o Filtro de Kalman.....	17
3.2.2 – União da Rede Neural Kohonen com a Lógica Fuzzy.....	18
3.2.3 – União da Rede Neural Kohonen com o Filtro de Kalman e com a Lógica Fuzzy .....	20
<b>4 – ANÁLISE DOS RESULTADOS .....</b>	<b>24</b>
4.1 – VALIDAÇÃO E RESULTADO DO ALGORITMO .....	24
4.1.1 – Rastreamento do Objeto Sobre a Reta.....	24
4.1.2 – Rastreamento do Objeto Sobre o Logaritmo .....	26
4.1.3 – Rastreamento do Objeto Sobre o Seno .....	27
4.1.4 – Rastreamento do Objeto com Mais de Uma Trajetória .....	29
<b>5 – CONCLUSÃO E SUGESTÃO PARA TRABALHOS FUTUROS.....</b>	<b>33</b>
5.1 – CONCLUSÃO.....	33
5.2 – SUGESTÕES PARA TRABALHOS FUTUROS .....	33
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>34</b>

## ÍNDICE DE FIGURAS

Figura 1 - Etapas de rastreamento.....	10
Figura 2 - Agrupamento em classes.....	11
Figura 3 - Treinamento competitivo. ....	11
Figura 4 - Curva padrão de treinamento da Rede Neural Kohonen.....	12
Figura 5 - Curva de treinamento da Rede Neural Kohonen utilizada neste projeto. ....	12
Figura 6 - Predição e atualização dos estados no Filtro de Kalman. ....	14
Figura 7 - Função de pertinência.....	15
Figura 8 - Comparação entre conjuntos clássicos e conjuntos difusos. ....	16
Figura 9 - Localização das áreas e suas funções de pertinência. ....	18
Figura 10 - Fluxograma do algoritmo tridimensional com Kohonen, Filtro de Kalman e Lógica Fuzzy juntos. ....	23
Figura 11 - Resultado do algoritmo com trajetória retilínea. ....	25
Figura 12 - Resultado do algoritmo com trajetória logarítmica.....	26
Figura 13 - Resultado do algoritmo com trajetória senoidal.....	28
Figura 14 - Rastreamento mal sucedido de dois objetos.....	29
Figura 15 - Rastreamento bem sucedido de dois objetos.....	31

# 1 – Introdução

Este trabalho estuda a identificação e o rastreamento de alvos móveis utilizando a Rede Neural Kohonen com o auxílio do Filtro de Kalman associado a sistemas baseados na Lógica Fuzzy.

A Rede Neural Kohonen, no projeto, é responsável pela identificação e rastreamento das trajetórias e é uma rede estruturalmente classificada como “feedforward”, seus neurônios são agrupados em duas camadas, uma de entrada e outra de saída e o sinal flui da camada de entrada para a de saída através de um caminho unidirecional. Durante o treinamento, a rede neural automaticamente adapta os pesos de seus neurônios de forma a unir os padrões de entrada em grupos com características semelhantes. Quando a entrada, representada pela posição de um objeto alvo no trabalho, é apresentada à rede, somente um neurônio é atualizado e, para tanto, existe um mecanismo que verifica qual neurônio possui o peso que mais se aproxima do valor da entrada, e com isto, existe uma competição entre eles. O neurônio vencedor é treinado para que seus pesos se aproximem dos valores da entrada, enquanto que os restantes são treinados para se afastarem do neurônio vencedor, realizando dessa forma o agrupamento da entrada. [13,65].

A finalidade do Filtro de Kalman é prever os futuros estados dinâmicos dos neurônios para evitar que estes mudem de trajetória. Tal filtro é basicamente estruturado em duas etapas, a predição, na qual ele avalia o futuro valor da variável em tratamento e a correção, na qual ele corrige tal valor previsto e o torna mais confiável ou próximo ao valor real. Para que essas duas atividades operem corretamente é necessário que se criem matrizes que tornem a covariância do erro da variável em questão gradativamente menor, tornando o valor mais confiável [35,37]. Assim esse filtro é tratado como uma ferramenta preventiva no rastreamento, sendo usado como uma das componentes dos neurônios, o que os força a competirem entre si levando em consideração a estimativa feita pelo Filtro de Kalman.

Para classificar e descrever lingüisticamente o posicionamento dos neurônios é usada a Lógica Fuzzy. Fundamentada na teoria dos Conjuntos Difusos (*Fuzzy*), uma generalização dos Conjuntos Tradicionais, acostumada a classificar as premissas unicamente como verdadeiras ou falsas, é usada para resolver problemas cuja informação é imprecisa ou vaga. A Lógica Fuzzy se baseia em codificações que atribuem um grau de verdade entre zero (0) e um (1) às premissas, respectivamente, entre totalmente falso e totalmente verdadeiro [49,59]. Ela é utilizada para transformar todos os pesos gerados ao longo da trajetória dos neurônios em termos lingüísticos e fáceis de identificar. Como as trajetórias utilizadas para validação nesse trabalho serão tridimensionais, a forma do campo visual é um cubo e este é dividido em dezoito (27) cubos iguais. O processo para gerar as funções de pertinência é análogo ao caso bidimensional, estudado pelo ex-

bolsista Denys de Souza Scheiner [63]. Para limitar o campo de visão são usados os pontos mais extremos entre todas as trajetórias, os quais seriam, na prática, captados e armazenados pela máquina que realizaria o rastreamento do(s) alvo(s). Finalmente, para saber onde o neurônio se encontra são usados os graus de pertinência referentes aos espaços que limitam o campo visual.

A união dessas três abordagens tem como objetivo prover um rastreamento com uma maior capacidade de generalização e que torne a identificação dos posicionamentos dos neurônios, e conseqüentemente das trajetórias, mais próxima da realizada pelo ser humano.

## 1.1 – Objetivo do Trabalho

Para realizar o rastreamento de alvos móveis com o objetivo de analisar o comportamento da mesma foram utilizadas técnicas de Redes Neurais Artificiais neste trabalho. Para realizar essa tarefa foram utilizadas redes neurais do tipo *Kohonen* que, com aprendizado não supervisionado, é treinada quando os padrões de entrada são fornecidos à entrada da rede. Como ela não precisa de um treinamento prévio, o treinamento acontece em tempo real.

Foram pesquisados em livros e artigos publicados por pesquisadores da área, para formar uma base de conhecimento sobre redes neurais. Após o estudo sobre redes neurais e a implementação de alguns algoritmos, foi realizado uma pesquisa sobre o Filtro de Kalman e sobre Lógica Fuzzy. Estes também foram pesquisados em livros e artigos. Para tal pesquisa, foram utilizadas as seguintes bibliotecas: (1) Instituto Nacional de Pesquisas Espaciais; (2) Departamento da Universidade Estadual Paulista – Campus Guaratinguetá. Os artigos e periódicos foram pesquisados na internet, como as bases de dados: (1) Web of Science – [www.webofscience.com](http://www.webofscience.com), e (2) Capes – [www.periodicos.capes.gov.br](http://www.periodicos.capes.gov.br). Em sites de busca: (1) yahoo (Brasil) – [www.yahoo.com.br](http://www.yahoo.com.br), e robôs de busca: (1) radix – [www.radix.com.br](http://www.radix.com.br), (2) uol – [www.radaruol.com.br](http://www.radaruol.com.br), e (3) google – [www.google.com](http://www.google.com); o robô (Metamáquina) de busca [www.metaminer.com](http://www.metaminer.com); (4) scirus – [www.scirus.com](http://www.scirus.com); e (5) Web Of Science – [www.webofscience.com](http://www.webofscience.com).

## 1.2 – Desenvolvimento Realizado

Este trabalho é uma continuação de três trabalhos de iniciação científica [60-63] onde o primeiro utiliza Redes Neurais Kohonen, o segundo e o terceiro empregam a Redes Neurais Kohonen associada ao Filtro de Kalman para realizar o rastreamento de alvos tanto bidimensionais quanto tridimensionais e o último usa Redes Neurais Kohonen com Lógica Fuzzy para realizar o rastreamento de alvos móveis e descrever lingüisticamente o posicionamento dos neurônios. Neste projeto, foram integradas as três abordagens em um mecanismo apenas, porém antes, como não havia ainda um sistema difuso para decodificar trajetórias tridimensionais, foi preciso criá-lo e em seguida a integração foi estudada para validação.

Para realizar tal integração e para enfatizar a contribuição de cada um dos três conceitos, foram simuladas trajetórias tridimensionais no software Matlab© e usando o mesmo software foram aplicadas as Redes Neurais Kohonen, o Filtro e Kalman e em seguida a descrição semântica. A fim de simular uma situação real no qual seriam utilizados sensores elétricos de posição para obter medidas discretas no tempo da posição do alvo, foram utilizados conjuntos de pontos em torno de um baricentro para representar o objeto-alvo. A validação se deu com pontos estáticos em diferentes posições do espaço para representar as posições do objeto-alvo em diferentes tempos e a Rede Neural foi treinada para rastrear esses pontos, separando as entradas em diferentes classes.

Para validar o algoritmo, primeiramente, foram simuladas trajetórias para esses conjuntos de pontos descritos por funções matemáticas na seguinte seqüência: (1) Reta; (2) Função Logarítmica e (3) Função Senoidal. Foi observado que a rede neural conseguiu rastrear os pontos com sucesso, o Filtro de Kalman conseguiu convergir para a rede com êxito e a descrição lingüística foi realizada também com êxito. Em seguida foram simulados múltiplos alvos, fornecendo um grau de dificuldade maior para a validação e o que se observou foi que a Rede Neural Kohonen cometia falhas ao tentar realizar o rastreamento, e quando o Filtro de Kalman foi inserido como uma realimentação para indicar à rede as estimações das entradas o rastreamento foi realizado com sucesso, não havendo mais troca de trajetórias pelos neurônios e em seguida foi realizado também com sucesso a descrição lingüística para os neurônios. Enquanto que a união do Filtro de Kalman com a rede se formava através de uma realimentação a união da Lógica Fuzzy com a rede era realizada diretamente, assim a Lógica Fuzzy era aplicada logo após os treinamentos.

## 2 - Histórico da Identificação de Alvos Móveis

O uso de técnicas de inteligência artificial e métodos de processamento digital de imagens está sendo muito freqüente hoje em dia e a identificação, ou rastreamento de alvos móveis, tem sido de grande interesse principalmente para sistemas de vigilância e lugares com radar.

Wu e Lin [2] utilizaram redes neurais artificiais em conjunto com sistemas nebulosos como um método de rastreo num campo aproximado de um alvo em movimento. A rede neural do tipo *feedforward* foi empregada por Schram, VanderLinden, Krose e Groen [3] na identificação de alvos estacionários e móveis para o controle de robôs através do uso de derivadas no tempo da posição do alvo e do manipulador.

Gaudiano, Zalama e Coronado [4] utilizam uma rede neural como controlador de um robô móvel. Esse controlador aprende de forma autônoma a indicar as distâncias percorridas de forma direta e inversa pelo robô através de um ciclo de treinamento não-supervisionado. Como o aprendizado é constante e não-supervisionado, perturbações, ou ruídos, significativas (como a mudança da planta do robô) levam a uma recalibração interna gradual de forma automática do controle do robô. Nas simulações realizadas, o robô foi capaz de rastrear alvos estacionários e móveis de forma correta dentro do erro mínimo estabelecido, graças à tolerância a ruídos da rede neural.

Um sistema de composição de dados para uma câmera situada no braço do robô SCORBOT-ER VII foi apresentado por Chen e Hsueh [5]. O sistema é capaz de realizar o rastreamento visual e a interceptação de alvos móveis utilizando uma rede neural Elman. A rede neural Elman é usualmente uma rede neural de duas camadas (a camada de entrada e a camada de saída, não possuindo a camada escondida) com um retorno de realimentação (feedback) da camada de saída para a camada de entrada. Nesta abordagem, o valor da saída é armazenado para ser processado no passo seguinte juntamente com a próxima entrada. Assim, se duas redes Elman idênticas (com os mesmos valores para os pesos) receberem uma mesma entrada em um dado passo, suas saídas poderão ser diferentes devido a estados de retorno diferentes.

Ma e Teng [6] exploram o problema do rastreamento de um alvo móvel em um campo aproximado usando redes neurais nebulosas. O alvo móvel irradia ondas de banda curta que incidem sobre uma série de sensores passivos. A posição do alvo é encontrada através de vários estimadores de ângulo e distância.

Yu e Azimi-Sadjadi [7] desenvolveram uma rede neural para a classificação de alvos explorando o comportamento dinâmico do alvo.

Yang e Meng [8] propõem uma rede neural artificial para a geração de trajetórias em um ambiente não-estacionário. A rede é usada para gerar trajetórias livres de colisões, evitar obstáculos em forma de U e rastrear alvos móveis em ambientes com obstáculos.

Cameron, Grossberg e Guenther [9] utilizam uma RNA que recebe como entrada uma imagem e a posição do ponto de onde se está visualizando a imagem. Como saída à rede fornece a representação da direção, da profundidade do cenário e da localização de objetos móveis. Essas representações são usadas para a navegação em simulações envolvendo o desvio de obstáculos e o rastreamento de um alvo móvel. Existem abordagens para a detecção e rastreamento de alvos inspirados em técnicas de rastreamento encontrados nos sistemas visuais de organismos vivos.

Missler e Kamangar [10] apresentam uma rede neural artificial que detecta e rastreia um objeto se movendo dentro de seu campo de visão. O trabalho é inspirado em funções de processamento observadas no sistema visual das moscas. A rede é capaz de detectar mudanças de intensidade de luz, determinar a movimentação de um alvo e fornecer as informações necessárias para realizar o rastreamento.

Dror, Florer, Rios e Zagaeski [11] realizaram estudos sobre a utilização de uma rede neural para reconhecimento de padrões que recebe dados em um formato similar ao do sonar de um morcego. A rede foi utilizada com sucesso no reconhecimento de rostos e para determinar a velocidade de um alvo que se movia em direção à câmera.

Navabi e Agarwal [12] apresentam um modelo artificial de organização de resposta neural de retina que pode servir como um quadro para a análise da resposta dinâmica em redes neurais artificiais, especialmente nos sistemas visuais.

Wann e Thomopoulos [27] utilizam uma rede neural artificial de aprendizado não-supervisionado no desenvolvimento de um sistema de composição de dados. Os dados são aplicados em problemas de detecção de alvos em um sistema multicanal de indicação de alvos móveis. As características dos dados recebidos de três canais de radares diferentes são extraídas através de técnicas de processamento de sinais digitais.

Shams [29] utiliza redes neurais no rastreamento de alvos utilizando sensores passivos. Cada sensor fornece o ângulo formado pela linha estimada entre o alvo e o sensor. A partir dos dados de todos os sensores, a rede neural é capaz de estimar a posição dos alvos. A rede neural proposta é uma rede auto-associativa chamada de ‘módulos elásticos múltiplos’.

D. Willett, C. Busch, e F. Seibert [41] utilizaram redes neurais do tipo *Kohonen* para realizar um algoritmo rápido para fazer análise de imagens. Este algoritmo poderia ser aplicado nas áreas médicas para segmentar objetos anatômicos, tais como tumores cerebrais de dados de ressonância magnética. Pode também ser aplicado a classificações de imagens de satélite.

Danil Prokhorov [42] desenvolveu um modo auxiliar para o treinamento de redes neurais recorrentes, pelo fato de elas estarem sendo utilizadas para uma ampla variedade de aplicações práticas e abstratas. Prokhorov realizou o treinamento da rede neural baseando-se no Filtro de *Kalman* Estendido.

H.D. Cheng, X. H. Jiang, Y. Sun, e Jingli Wang [43] fizeram um resumo das técnicas de segmentação de imagens coloridas que são utilizadas para realizar sua análise através de redes neurais.

Sung-Bae Cho [44] utilizou um método híbrido que combinava algoritmos genéticos com redes neurais para realizar reconhecimento de padrões. O algoritmo produziu melhores resultados que alguns métodos convencionais.

Neste trabalho a solução escolhida para efetuar o rastreamento de alvos móveis é a utilização das RNAs Kohonen devido as suas interessantes propriedades, as quais serão comentadas a seguir, e principalmente por este trabalho, conforme já mencionado, ser a continuação de outros trabalhos realizados anteriormente [60-63].

### 3 - Redes Neurais Kohonen com o Filtro de Kalman e com a Lógica Fuzzy Na Identificação e Rastreamento de Alvos Móveis

Esse trabalho destina-se a criar um instrumento de identificação e rastreamento de alvos móveis utilizando a Rede Neural Kohonen com o auxílio do Filtro de Kalman e com a aplicação da Lógica Fuzzy. A união do filtro à rede e da Lógica Fuzzy a mesma já foram desenvolvidas em trabalhos anteriores [60-63], cabendo a essa etapa do projeto acoplar os dois mecanismos criados nos trabalhos citados.

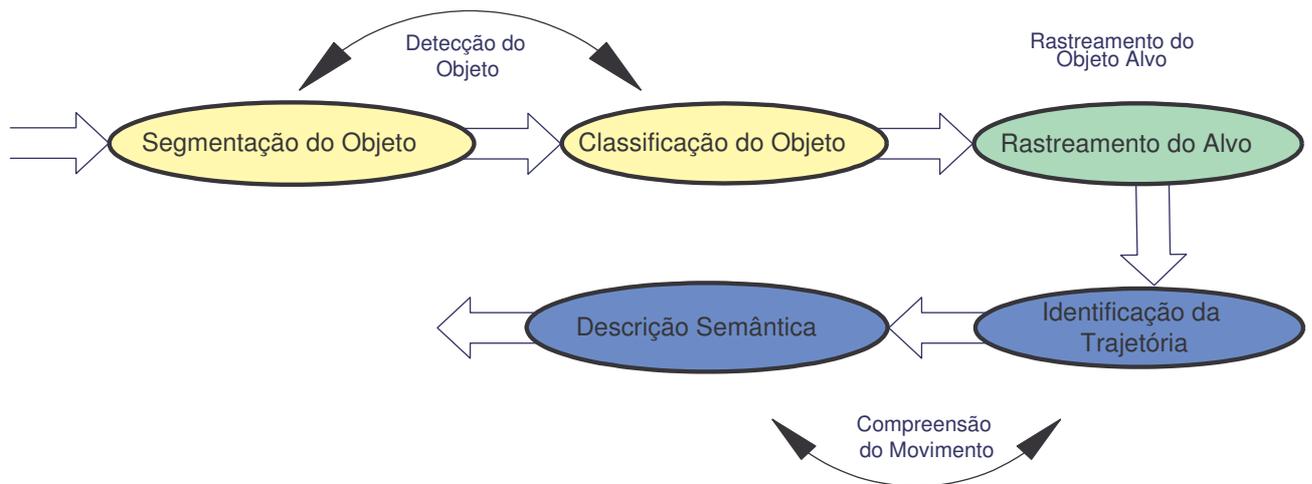


Figura 1 - Etapas de rastreamento.

#### 3.1 – Introdução Teórica

A ênfase dada nesta etapa do projeto é na acoplagem dos três conceitos, assim, ao longo deste capítulo será apresentada uma breve introdução teórica destas três etapas: (1) Rede Neural Kohonen, (2) Filtro de Kalman e (3) Descrição semântica através da Lógica Fuzzy.

##### 3.1.1 – Rede Neural Kohonen

As redes neurais podem ser categorizadas utilizando dois critérios: a sua estrutura e o seu algoritmo de aprendizado.

Quanto à estrutura, elas podem ser alimentação à frente (*feedforward*) ou recorrentes (*recurrent*). As redes *feedforward* são agrupadas em camadas e o sinal flui da camada de entrada para a de saída via conexões unidirecionais, enquanto que nas redes *recurrent* as saídas de alguns neurônios são realimentados para os mesmos neurônios ou para neurônios de camadas anteriores, assim os sinais podem flui em um sentido positivo ou negativo em relação às camadas.

Quanto ao algoritmo de aprendizado, elas podem ter um aprendizado supervisionado ou não supervisionado. Em um algoritmo de aprendizado supervisionado é preciso fornecer um sinal de saída, o qual se deseja atingir, assim os pesos dos neurônios são ajustados de acordo com a diferença entre a saída desejada e a atual da rede dada uma entrada. E em um algoritmo de aprendizado não supervisionado, as entradas automaticamente adaptam os pesos dos neurônios e se agrupam a determinados padrões de entradas com características semelhantes.

A Rede Neural Kohonen é um tipo de rede neural de aprendizado não-supervisionado e com uma estrutura *feedforward*, portanto ela oferece duas vantagens: ela não necessita de um professor ou supervisor para indicar a saída desejada, o que pode ser de grande vantagem, pois isso aumenta a velocidade do rastreador considerando que o mecanismo deve rodar em tempo real e como ela tem uma estrutura *feedforward*, a sua construção é mais simples, o que implica também implica em uma redução de tempo. Nesta rede, os padrões de entrada são identificados e agrupados utilizando a grau de proximidade geométrica entre as entradas e os neurônios como se pode observar na figura 2. Esse grau geralmente é a distancia euclidiana, como está ilustrado na figura 3.

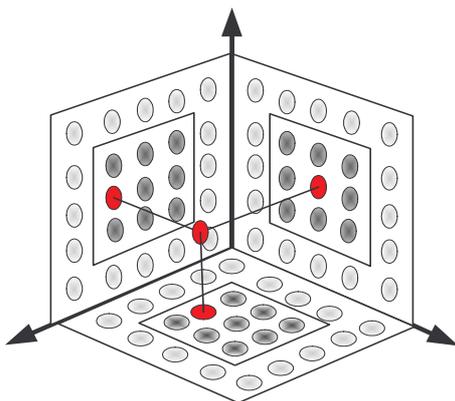


Figura 2 - Agrupamento em classes.

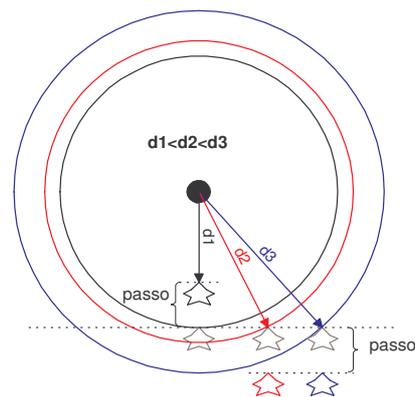
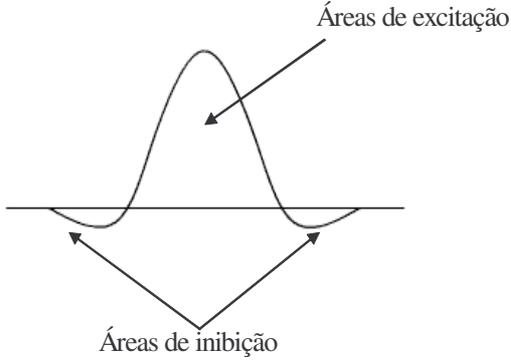


Figura 3 - Treinamento competitivo.

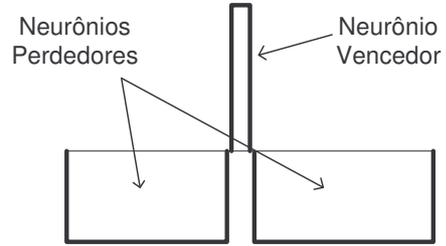
Com essas características, à Rede Neural Kohonen é necessário apenas informar os padrões de entrada, e a mesma detecta qual neurônio possui um padrão mais parecido com o da entrada, sendo, portanto o mais adequado a convergir e o seleciona para alterar os seus pesos.

Em termos mais matemáticos, quando um padrão de entrada é apresentado à rede, esta entrada  $x^k$  terá uma saída  $y^k$  da rede neural, sendo que  $y_i^k \in \{-1, +1\}$ , e para essa saída, somente um neurônio está ativo, ou seja, está recebendo treinamento, então:  $(y_j^k = +1)$ . Somente esse neurônio será treinado, agrupando o padrão selecionado e ele passa a ser chamado de vencedor, sendo esse processo chamado de “winner-takes-all”. Neste projeto, os outros neurônios são afastados do padrão de entrada atual, por conseguinte a curva de treinamento tem seu formato ilustrado na figura 5, em comparação com a curva de treinamento padrão da rede na figura 4.

O  $j$ -ésimo neurônio ativo indica que a entrada  $x^k$  pertence à classe  $j$ , portanto, a rede neural classifica ou vincula os padrões de entrada em classes. Neste projeto, as entradas são tridimensionais, então:  $x^k = [x_1^k, x_2^k, x_3^k]$ .



**Figura 4 - Curva padrão de treinamento da Rede Neural Kohonen.**



**Figura 5 - Curva de treinamento da Rede Neural Kohonen utilizada neste projeto.**

Para cada entrada, um neurônio vencedor  $w^j$  é selecionado, sendo:

$$\|x^k - w^j\|^2 < \|x^k - w^i\|^2, \forall i \neq j \quad (1)$$

Determinado o neurônio vencedor, a rede neural o treina usando a seguinte fórmula:

$$W_{\text{novo}} = W_{\text{anterior}} + \alpha (X - W_{\text{anterior}}) \quad (2)$$

sendo  $W$  associado a um neurônio, e  $\alpha$ , à taxa de aprendizado da rede (*learning rate*).

No entanto, antes que seja realizado o treinamento é imprescindível que todos os neurônios e entradas sejam normalizados:

$$\|x^k\|^2 = \|w^i\|^2 = 1, \forall k, i \quad (3)$$

Tal normalização é feita determinado o valor de  $M$  [11]:

$$M = \max\{\|x^k\|^2\}, \forall k \quad (4)$$

sendo  $M$  o máximo da norma de  $x^k$ .

E em seguida, aumentando a dimensão de  $x^k$  através da inserção de mais uma componente:

$$x_{n+1}^k = (M - \sum_{i=1}^n (x_i^k)^2)^{1/2} \quad (5)$$

E usando como entrada o vetor  $x^k = M^{-1/2}[x_1^k, x_2^k, \dots, x_n^k, x_{n+1}^k]$ .

Em termos de rastreamento em um espaço físico, as entradas apresentadas à rede seriam as coordenadas do objeto-alvo, captadas por sensores elétricos e assim, através da distância euclidiana, o neurônio mais próximo deste objeto seria treinado e convergido em direção à posição dele.

### 3.1.2 – Filtro de Kalman

A função do Filtro de Kalman é, a partir de estados anteriores, gerar uma estimativa dos estados futuros de um sistema dinâmico. Esse processo é realizado em duas fases: (1) Predição ou propagação no tempo e (2) Atualização ou correção da estimativa. Antes de começar a utilizar o filtro, é preciso inicializá-lo, fornecendo a covariância de erro inicial dos estados e os ruídos de processo e de medição. Esses valores são guardados em matrizes para que sejam usados no futuro e de forma simultânea. Esse filtro se baseia na Teoria da Probabilidade, possuindo equações que estimam o valor futuro da variável do sistema dinâmico analisado e diminuem a covariância do erro da estimação, tornando-a mais confiável.

Na fase de estimação, o filtro propaga no tempo uma estimativa do estado do sistema chamada de estimativa da matriz de covariância de erro, chamadas de estimativas *a priori*, que possuem um certo erro. As equações são:

$$X_{k+1} = \text{PHI} * X_k + \text{GAMMA} * U_k + W_k \quad (6)$$

$$P_{k+1} = \text{PHI} * P_k * \text{PHI}^T + G * Q * G^T \quad (7)$$

sendo,  $X_k$ , o vetor que contém os estados anteriores,  $X'_{k+1}$  é o vetor que contém os estados estimados *a priori*, e  $P'_{k+1}$  a matriz de covariância de erro estimado *a priori*

Na fase de atualização, a medida real do estado do sistema é obtida, assim como os ruídos de medição, permitindo o filtro fazer a correção das estimativas realizadas na primeira fase, assim elas passam a se chamar estimativas *a posteriori*. As equações dessas fases são:

$$K = P_{k+1} * C * 1 / (C * P_{k+1} * C^T + R) \quad (8)$$

$$X_{k+1} = X_{k+1} + K * (Z - C * X_{k+1}) \quad (9)$$

$$P_{k+1} = (1 - K * C) * P_{k+1} \quad (10)$$

sendo K, o ganho de Kalman, Z, o vetor que contém as medidas reais dos estados,  $X_{k+1}$ , o vetor de estados corrigidos ou estimados *a posteriori*, e  $P_{k+1}$  é a matriz covariância de erro corrigida

As matrizes C, W, G e Q são ruídos de processo e de medição, PHI é a matriz que reporta o estado anterior ao estado futuro da rede neural, e U é uma entrada de controle opcional.

Essas duas etapas são repetidas, tornando as estimativas cada vez mais confiáveis devido à

diminuição da covariância de erro conforme ilustra a figura 6, até que o filtro convirja para o sistema estudado fornecendo valores com erros desprezíveis dependendo do propósito a ser conquistado.

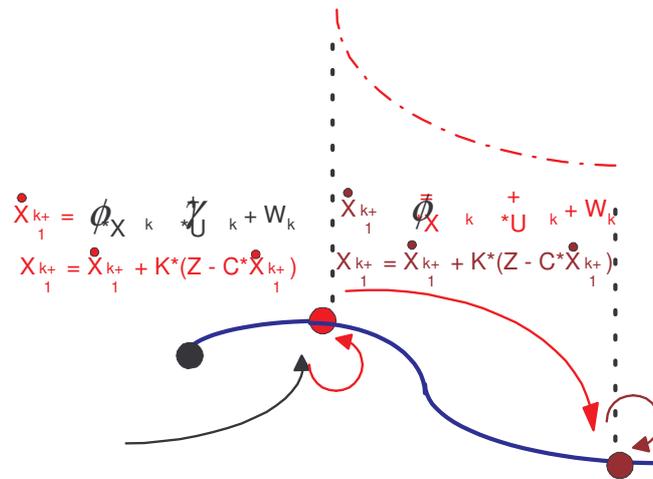


Figura 6 - Predição e atualização dos estados no Filtro de Kalman.

Uma característica importante do Filtro de Kalman é que ele suporta ruídos brancos Gaussianos – com média igual a zero. Esses ruídos costumam se manifestar em sensores elétricos de posição, por exemplo.

### 3.1.3 – Descrição Semântica

A necessidade e importância de prover uma descrição lingüística para as atitudes humanas em sistemas de visão têm feito com que esse assunto recebesse grande atenção. A idéia é gerar um conjunto de palavras ou expressões para descrever o comportamento dos objetos a serem vigiados.

Em 1995, R. K. Srihari [45] realizou um workshop sobre linguagem natural e visão computacional.

Kojima [50] propôs um novo método de descrição natural de atitudes humanas em seqüências de vídeos reais.

Remagnino [51] propôs um sistema de vigilância baseado em eventos envolvendo pedestres e veículos. Este seria capaz de descrever as interações entre os objetos em movimento em cenas 3D.

Intille e Bobick [52] elaboraram um sistema automático de anotações para cenas esportivas. Cada informação do jogador era representada por uma rede baseada em evidências visuais e constatações temporais.

Kollnig [57] utilizou funções Fuzzy com diferentes graus de incerteza para associar verbos com detalhes quantitativos, obtidos por análise de seqüência automática de imagens para gerar descrições em linguagem natural de cenas de trânsito.

Herzog [58] desenvolveu o projeto que usa linguagem natural para descrever cenas visuais.

A descrição semântica possui grande importância no movimento e trajetória de alvos móveis devido ao seu poder de aplicação. Além deste também se podem mencionar aplicações em diferentes áreas, por exemplo: (1) Segurança – Militar (monitoramento do espaço aéreo, marítimo e territorial) e Civil (vigilância envolvendo pedestres e veículos) [51], (2) Psicológica – comportamento humano [50], e (3) Transito – comportamento do trânsito em grandes cidades [57];

A descrição semântica escolhida para este trabalho será realizada através da modelagem difusa (Fuzzy), devido seu alto potencial para inferir decisões e lidar com dados imprecisos de maneira simples e natural, conforme mostrado a seguir.

### 3.1.3.1 – Teoria dos Conjuntos Fuzzy

Diferente dos Conjuntos Tradicionais, onde um elemento tem a possibilidade de pertencer ou não a um conjunto, na teoria dos Conjuntos Difuso (*Fuzzy*), um elemento associado a um conjunto base  $A(x)$  possui valor que indica o quanto ele pertence a esse conjunto, esse valor pode variar de 0 a 1, e é dado por uma função de pertinência conforme ilustra a figura 7. Assim esse conjunto difuso pode gerar respostas parcialmente falsas ou parcialmente verdadeiras, sendo de grande valia para sistemas onde o obtenção de valores é imprecisa, vaga e incerta.

$$A: X \rightarrow [0,1] \quad (11)$$

sendo que  $A(x)$  representa o grau com que  $x$  pertence a  $A$  e indica o grau com que  $x$  se enquadra na categoria representada por  $A$ .

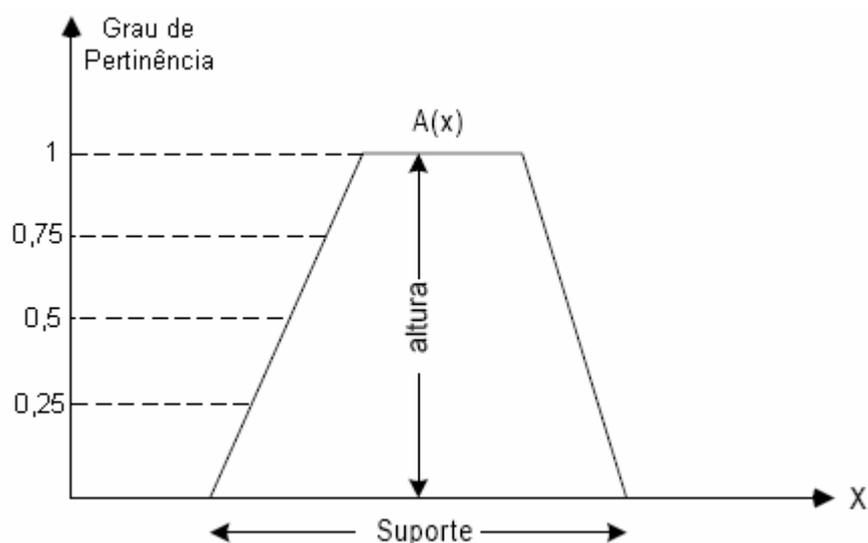


Figura 7 - Função de pertinência.

Na figura 8 é ilustrada uma comparação entre os Conjuntos Clássicos e os Conjuntos Difusos, mostrando a vantagem da última em relação à primeira e é feita de maneira compacta uma

explicação de como são inseridos os conectivos lógicos para gerar as inferências a partir da análise dos valores do conjunto.

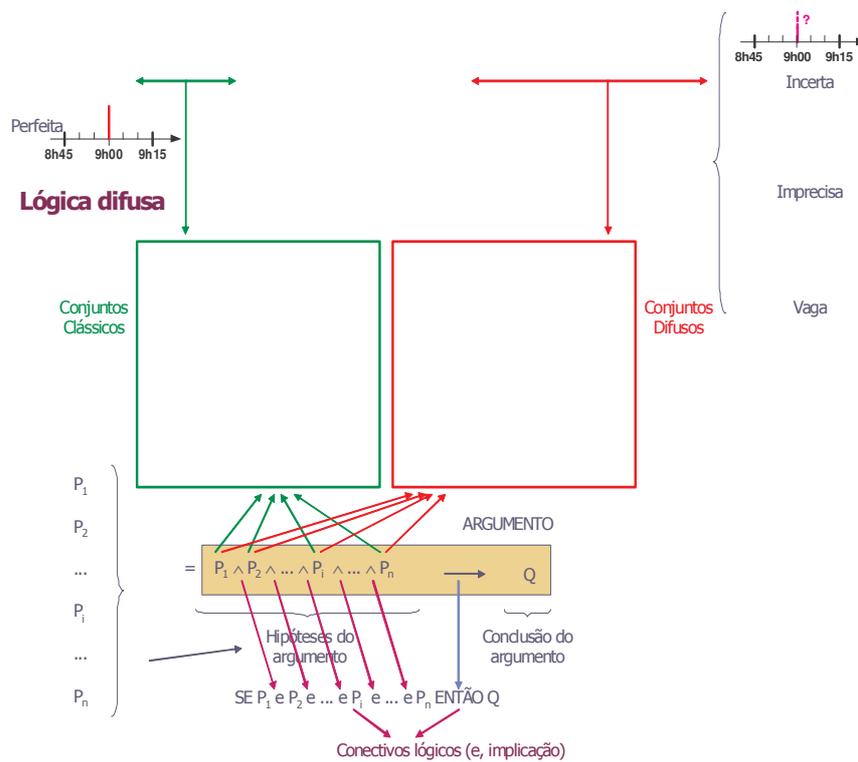


Figura 8 - Comparação entre conjuntos clássicos e conjuntos difusos.

### 3.1.3.2 – Lógica Fuzzy

Uma das características da Lógica Clássica, o axioma do Terceiro Excluído, diz que não há alternativa para um valor verdade além do par {Verdadeiro, Falso}. No mundo real, entretanto, os valores obtidos podem ser incertos, vagos, desconhecidos, imprecisos, parcialmente verdadeiros etc., o que torna essa Lógica obsoleta para esses universos, assim é preciso usar uma das extensões dela, chamada Lógica Fuzzy.

A Lógica Difusa (*Fuzzy*) é fundamentada na teoria dos Conjuntos Difusos (*Fuzzy*) para sua representação, e esta tem um grande poder de decisão, gerando respostas baseadas em informações vagas, ambíguas e qualitativamente incompletas e imprecisas, ela permite que sejam atribuídos aos valores do par {Verdadeiro, Falso} graus de pertinência, geradas por funções de pertinência.

A teoria de conjuntos e lógica difusa são usados em diversos setores de pesquisa e desenvolvimento. Na matemática, ela pode ser empregada na modelagem de valores parciais verdadeiros ou idéia de vago, enquanto que na inteligência artificial ela é usada com raciocínio aproximado e é usada para emular a maneira humana de pensar e na engenharia, ela é usada para gerar funções numéricas a partir de computação numérica.

Um algoritmo difuso é uma seqüência ordenada de instruções, as quais podem ser atribuições difusas, sentenças condicionais ou comandos de ação, com o propósito de gerar uma inferência a partir de informações incompletas, vagas, imprecisas etc. Alguns exemplos de instruções se encontram abaixo:

São atribuições difusas: "x é bom"; "Y é regular"; "z é satisfatório".

São sentenças condicionais: "se x é bom e y é regular, então z é satisfatório".

São comandos de ação: "aumente um pouco".

Nesse trabalho serão usadas as regras difusas SE – ENTÃO, essas regras são conjuntos de decisão e controle que são implementados segundo condições de comportamento e assumem a forma geral:

$$\text{SE } \langle(x \text{ é } A)\rangle \text{ ENTÃO } \langle(y \text{ é } B)\rangle \quad (12)$$

sendo  $(x \text{ é } A)$  e  $(x \text{ é } B)$  são conhecidos como afirmações, sendo A e B termos lingüísticos definidos em conjuntos difusos em universos de discursos X e Y, respectivamente. A proposição  $(x \text{ é } A)$  é conhecida como premissa da regra e a proposição  $(x \text{ é } B)$  é conhecida como conclusão da mesma. Na teoria de sistemas difusos as proposições podem ser classificadas como difusa elementar ou difusa geral. A primeira atribui um valor de verdade, gerado por uma função de pertinência, a uma única variável, por exemplo: “a temperatura está alta” ou “o sinal está fechado”. A segunda junta dois valores de verdade atribuídos a variáveis diferentes através de um conectivo “e”, por exemplo: “a temperatura está alta e a umidade está baixa” ou “o sinal está fechado e a rua está vazia” [59].

### 3.2 – Acoplagem dos conceitos

Nesta seção serão apresentados os procedimentos realizados para unir a Rede Neural Kohonen com o Filtro de Kalman, a Rede Neural Kohonen com a Lógica Fuzzy e as três abordagens juntas.

#### 3.2.1 – União da Rede Neural Kohonen com o Filtro de Kalman

Para um dado padrão de entrada, que são as coordenadas da posição do alvo, a rede neural *Kohonen* realiza a competição entre os neurônios e determina qual será o neurônio vencedor. Em seguida, esta treina o neurônio, alterando seus pesos de forma a aproximá-los dos padrões de entrada. Com este processo, as entradas são agrupadas segundo seus padrões. Se este padrão de entrada representar um objeto-alvo no espaço, os neurônios irão se aproximar deste objeto, realizando o rastreamento e identificando a trajetória do mesmo. Como o objeto a ser rastreado é móvel, os padrões de entrada se modificam representando que o alvo se locomove no espaço, assim serão necessários tantos treinamentos quantas as modificações das entradas. Contudo, quando existirem mais de um objeto a ser rastreado, os neurônios poderão trocar de trajetória, pois como o

critério utilizado é a distancia euclidiana, os diferentes padrões podem assumir valores parecidos fazendo a rede vincular equivocadamente diferentes padrões de entrada. Feito isso, a identificação e o rastreamento não serão mais plausíveis, pois os neurônios perderam os alvos.

Uma solução para esse problema se encontra no uso do Filtro de Kalman. Como a rede é um sistema dinâmico, as coordenadas X, Y e Z dos neurônios são fornecidas ao filtro. No momento em que a rede é treinada, ocorre a fase de propagação no tempo, na qual o filtro gera os futuros pesos da rede, baseado nos estados anteriores da mesma e quando os neurônios acabam de unir os padrões de entrada, convergindo para as mesmas, seus valores e os ruídos de medição são fornecidos para o filtro, o qual, na fase de atualização de sua estimativa, corrige o valor a priori. Desse modo, as coordenadas futuras da rede neural são indicadas pelo filtro. Essas estimações são usadas com uma componente a mais da rede, além das cartesianas, obrigando-a a realizar o processo de competição levando em consideração o formato e a direção do caminho já rastreado por cada neurônio. Isso diminui a chance de ocorrer uma falha de rastreamento. A união dessas duas ferramentas se realiza dessa maneira e no fluxograma exposto na figura 10 pode ser observada a implementação de tal acoplamento.

### 3.2.2 – União da Rede Neural Kohonen com a Lógica Fuzzy

Como as trajetórias utilizadas para validação do algoritmo são tridimensionais foi criado nesta etapa um sistema difuso tridimensional semelhante ao sistema criado pelo ex-bolsista Denys de Souza Scheiner em seu trabalho “Descrição Semântica através de Modelagem Difusa (Fuzzy) para Rastreo de Alvos Móveis” realizado em 2006 [63]. Para unir a Rede Neural Kohonen e a Lógica Fuzzy e conseqüentemente efetuar a descrição semântica de rastreamento de objeto-móvel, inicialmente dividiu-se o universo, cujo objeto-móvel está inserido, em 27 volumes conforme se pode observar abaixo e inseriram-se as funções de pertinência de cada volume, conforme ilustra a figura 9.

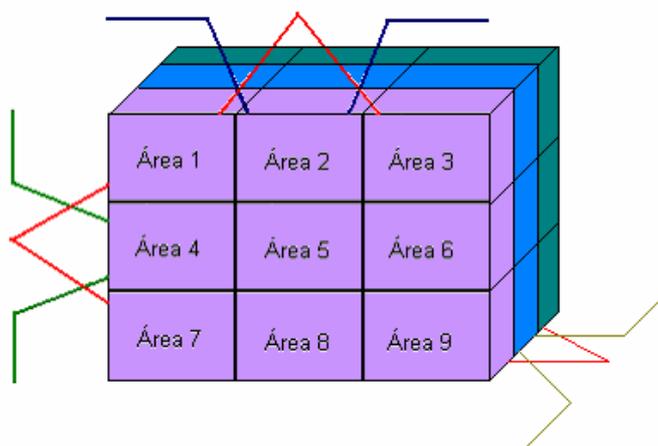


Figura 9 - Localização das áreas e suas funções de pertinência.

As 27 áreas estão listadas em baixo:

As áreas de 1 a 9 encontram na parte frontal referido ao eixo z e estão em roxo na figura acima, assim elas são denominadas:

- (1) Diagonal frontal superior esquerda
- (2) Na frente e em cima
- (3) Diagonal frontal superior direita
- (4) Na frente e à esquerda
- (5) Na frente e no centro
- (6) Na frente e à direita
- (7) Diagonal frontal inferior esquerda
- (8) Na frente e em baixo
- (9) Diagonal frontal inferior direita

As áreas de 10 a 18 encontram na parte mediana referida ao eixo z e estão em azul na figura acima, assim elas são denominadas:

- (10) Diagonal mediana superior esquerda
- (11) No meio da profundidade e em cima
- (12) Diagonal mediana superior direita
- (13) No meio da profundidade e à esquerda
- (14) No meio do espaço
- (15) No meio da profundidade e à direita
- (16) Diagonal mediana inferior esquerda
- (17) No meio da profundidade e em baixo
- (18) Diagonal mediana inferior direita

As áreas de 19 a 27 encontram na parte traseira referida ao eixo z e estão em verde na figura acima, assim elas são denominadas:

- (19) Diagonal traseira superior esquerda
- (20) Atrás e em cima
- (21) Diagonal traseira superior direita
- (22) Atrás e à esquerda
- (23) Atrás e no centro
- (24) Atrás e à direita
- (25) Diagonal traseira inferior esquerda
- (26) Atrás e em baixo
- (27) Diagonal traseira inferior direita

Assim a função do sistema difuso criado consiste em desmembrar, a cada etapa de treinamento, as posições do móvel em suas respectivas coordenadas através das funções de pertinência, determinar a posição espacial do neurônio e descrever lingüisticamente o comportamento dos mesmos.

### 3.2.3 – União da Rede Neural Kohonen com o Filtro de Kalman e com a Lógica Fuzzy

Na implementação das duas etapas anteriores, foi constatado que a conexão entre a rede e o filtro é diferente da conexão entre a rede e o sistema difuso. A primeira se realiza através de uma realimentação, pois o filtro necessita de informações anteriores, incluindo as atuais, para poder estimar o estado da rede e corrigir o rastreamento dos alvos. Logo após o treinamento dos neurônios, o estado do sistema é informado a rede para que esta possa atualizar suas estimações. Já a segunda é realizada utilizando os pesos dos neurônios logo após o seu treinamento para descrever seus comportamentos. A descrição semântica, então, é aplicada diretamente ao rastreamento efetuado pela rede.

Em vista dessas conclusões, o acoplamento da rede com o filtro e com o sistema difuso pode ser implementado, utilizando a rede o filtro como um mecanismo de rastreamento mais aperfeiçoado, o qual a cada convergência para os valores de entrada, fornece o estado dos neurônios para o sistema difuso, e este através de regras SE – ENTÃO e de funções de pertinência descreve lingüisticamente o comportamento da rede.

O fluxograma do algoritmo desenvolvido em conjunto com ex-bolsistas de iniciação científica [60-63] se encontra exposto na figura 10 para um melhor entendimento:

Os passos para realizar o treinamento da rede juntamente com as outras duas ferramentas são descritos a seguir:

- 1 - São estabelecidas condições iniciais para os pesos dos neurônios  $w^i$  de forma aleatória.
- 2 - É estabelecido um quociente de aprendizado  $\alpha$  para os pesos. Inicialmente, foi definido  $\alpha = 0,01$ .
- 3 – São estabelecidas condições iniciais para as matrizes do Filtro de Kalman, baseado nos dados do sistema dinâmico em questão, ou seja, a Rede Neural.
- 4 - Normalizam-se as entradas e os pesos de forma que, fique como a equação (3). Os resultados da normalização serão os vetores sinapses (no caso dos pesos dos neurônios) e os vetores medidas (no caso dos padrões de entrada).
- 5 - Dado uma entrada, é calculado um neurônio vencedor, o qual possui a maior variável interna  $u(i)$ , que será o neurônio que mais se aproxima da posição atual da entrada.
- 6 - O peso do neurônio vencedor deverá ser atualizado da seguinte forma:

$$\text{sinapse}(i) = \text{sinapse}(i) + \alpha(\text{medida}(po) - \text{sinapse}(i))$$

Sendo que “i” representa o índice da sinapse vencedora e “po” a entrada apresentada para a rede, ambas normalizadas no passo 4.

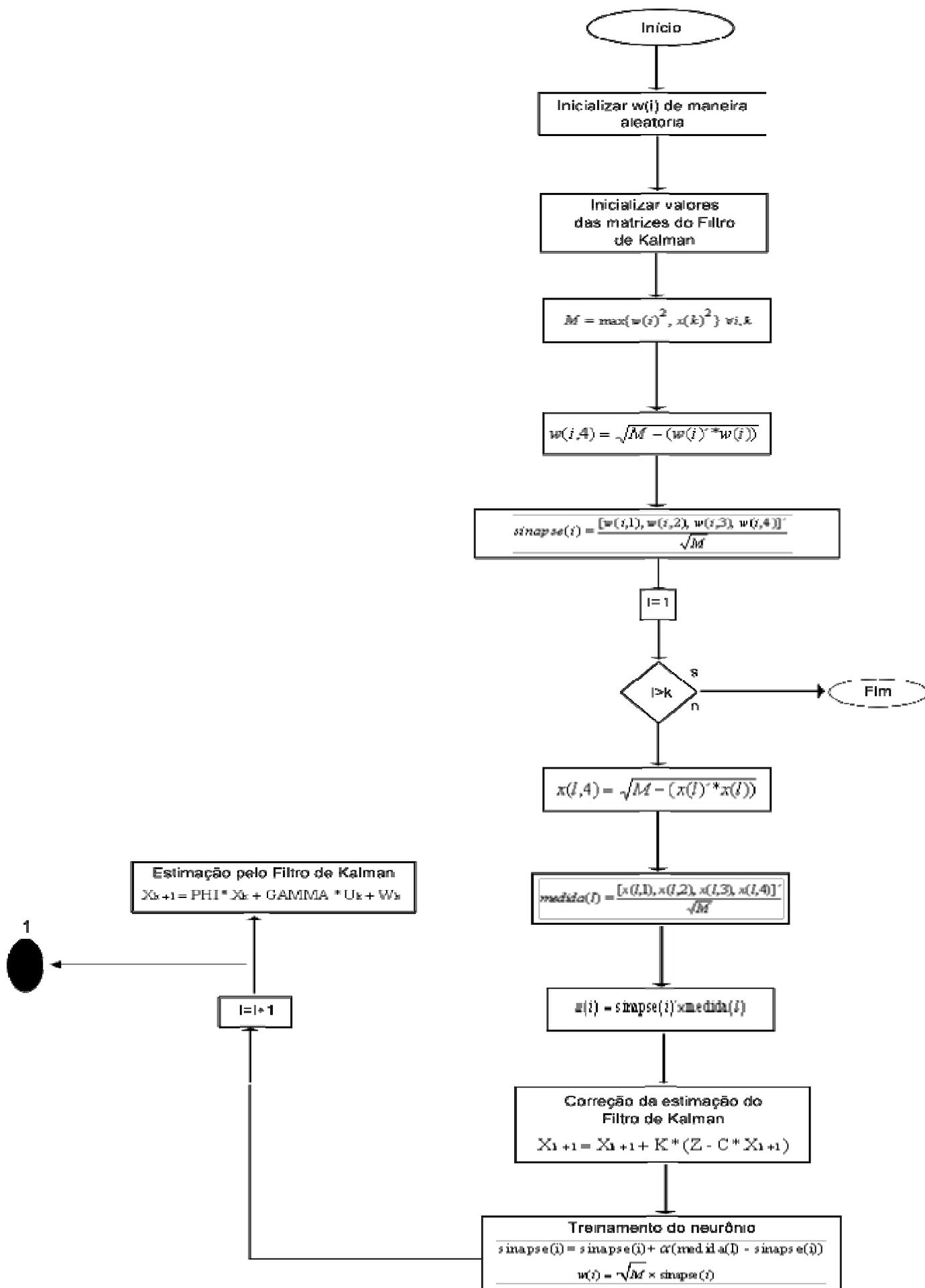
7 – Realiza-se a correção da estimativa do Filtro de Kalman com a equação (9)

8 – Os limites mínimos e máximos do campo visual são determinados. Para isso, é feita uma varredura de todas as entradas e todos os neurônios para detectar qual dessas variáveis possui os maiores e menores valores para cada eixo coordenado.

9 – O campo visual é dividido em 27 cubos iguais usando os valores mínimos, médios e máximos encontrados para cada eixo coordenado.

10 – Através das funções de pertinência para cada região do espaço visual, as descrições lingüísticas são geradas para cada neurônio.

11 – Retorna-se ao passo 4, até que todas as entradas tenham sido apresentadas à rede.



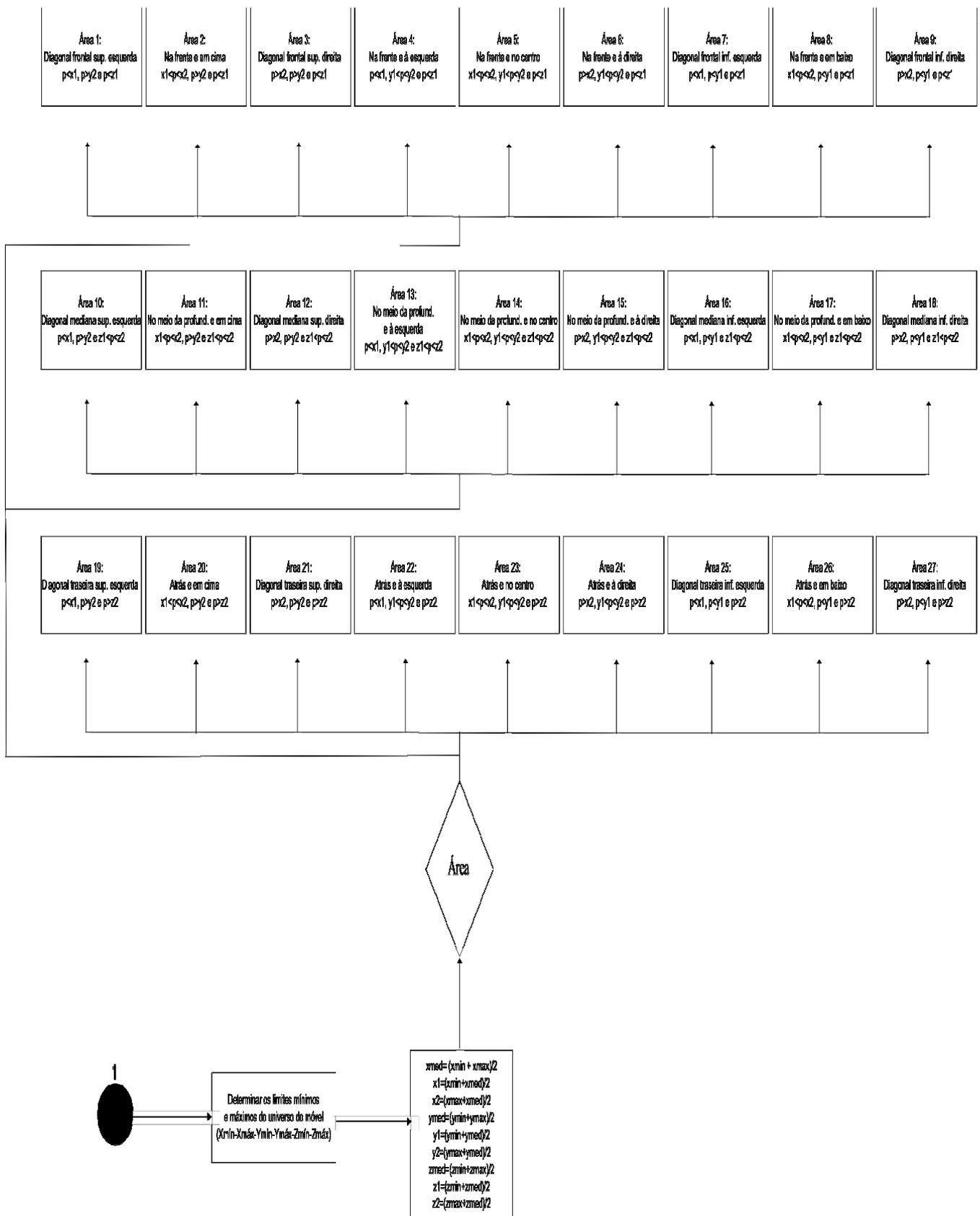


Figura 10 - Fluxograma do algoritmo tridimensional com Kohonen, Filtro de Kalman e Lógica Fuzzy juntos.

## 4 – ANÁLISE DOS RESULTADOS

Nesta seção serão aplicadas as três abordagens, Redes Neurais Kohonen, Filtro de Kalman e Lógica Fuzzy de forma seqüencial para mostrar a utilidade de cada uma. Simulações realizadas mostraram que é válido unir as três abordagens em um mecanismo apenas. Ao mesmo tempo em que a Rede Neural Kohonen rastreava o objeto, o Filtro realizava a função de estimar os valores futuros da entrada e corrigir o rastreamento e era realizada também descrição semântica.

### 4.1 – Validação e Resultado do Algoritmo

Os baricentros das classes, que representam as posições do objeto-alvo, foram gerados a partir de funções  $f(x)$ , e a partir daí foram gerados pontos a redor desses baricentro, para representar as medições feitas pelos sensores elétricos de posição. Para realizar a validação, os valores do baricentro foram atualizados após cada etapa de treinamento. Após alguns treinamentos iniciais do neurônio, o seu valor começa a ser informado ao Filtro de Kalman, o qual realiza uma estimação do valor futuro do neurônio e após o próximo treinamento ele corrige essa estimação. Em seguida, a posição do neurônio é desmembrada em suas coordenadas e cada uma dessas é analisada pelo sistema difuso criado, que utiliza a Lógica Fuzzy, para informar, utilizando uma lingüística mais humana, onde o neurônio se encontra. Em cada treinamento, as condições iniciais dos neurônios são iguais às condições finais do treinamento anterior, ou seja, o neurônio vencedor é inicializado de forma aleatória somente antes de qualquer treinamento.

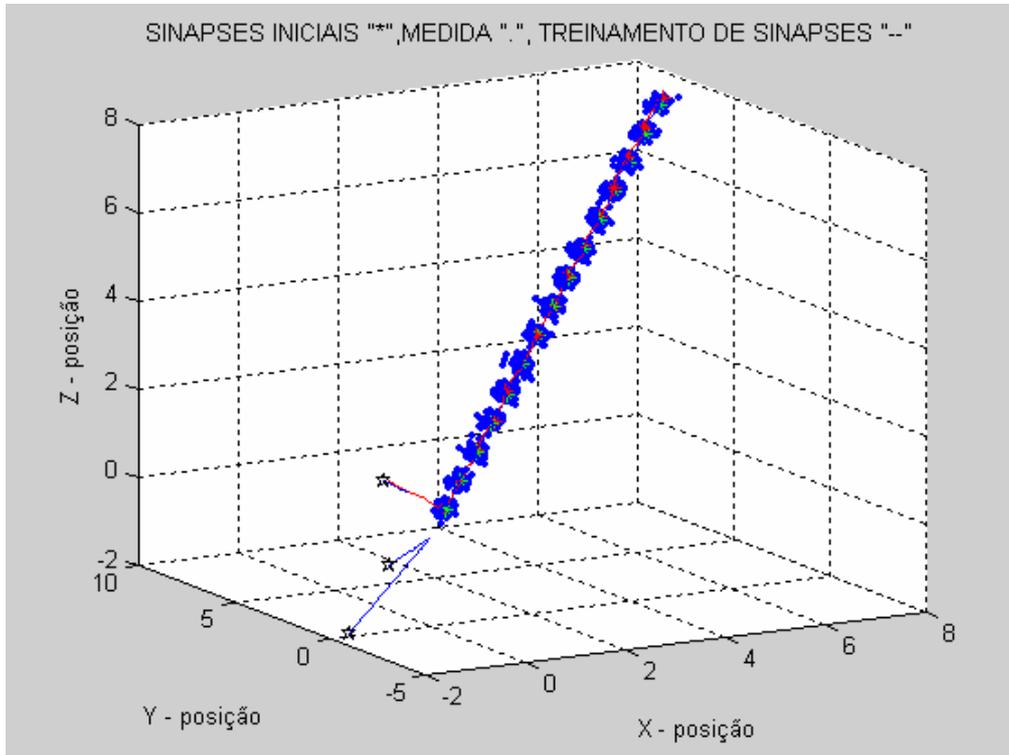
Aqui, o algoritmo foi validado de forma a realizar a identificação da trajetória do objeto alvo pela rede neural, a estimação da posição do neurônio vencedor pelo Filtro de Kalman e a descrição semântica do posicionamento do neurônio. Foram simuladas três trajetórias distintas com as equações das seguintes curvas: Reta, Logaritmo e Seno e em seguida foi simulado um rastreamento com mais de uma trajetória ao mesmo tempo.

#### 4.1.1 – Rastreamento do Objeto Sobre a Reta

A primeira simulação de rastreamento foi realizada sobre um objeto alvo com uma trajetória segundo uma reta tridimensional. Em torno dos baricentros foram criados pontos que são as medições realizadas pelos sensores elétricos de posição e a cada passo a variável T foi incrementada de “1” para que os pontos se deslocassem sobre a trajetória. Os valores dos baricentros foram governados pela seguinte equação:

$$X(T) = T \qquad Y(T) = 2 * T \qquad Z(T) = T \qquad (13)$$

Pela teoria desenvolvida na seção 3, o neurônio vencedor deve ser capaz de realizar o rastreamento dos pontos, enquanto os perdedores devem se afastar do neurônio vencedor. Já o Filtro de Kalman deve identificar a forma da rede neural e a Lógica Fuzzy deve descrever corretamente o comportamento do neurônio rastreador.



**Figura 11 - Resultado do algoritmo com trajetória retilínea.**

A figura 11 mostra o resultado do rastreamento. As estrelas pretas representam as condições iniciais das sinapses, os pontos em vermelho representam os baricentros e os traços azuis representam o treinamento das sinapses. Os pontos em azul representam a nuvem de pontos e a linha tracejada em vermelho representa a estimativa do Filtro de Kalman.

Pela figura, pode-se observar que a rede neural rastreia o alvo corretamente, enquanto o Filtro de Kalman realiza a estimativa e correção da mesma. No início, o Filtro de Kalman possui estimativas com certo grau de erro, mas ele converge para a rede neural. Em seguida foi aplicado sistema difuso, obtendo-se:

Posição: 1 - diagonal frontal inferior esquerda	Posição: 26 - no meio do espaço
Posição: 2 - diagonal frontal inferior esquerda	Posição: 27 - no meio do espaço
Posição: 3 - diagonal frontal inferior esquerda	Posição: 28 - no meio do espaço
Posição: 4 - diagonal frontal inferior esquerda	Posição: 29 - no meio do espaço
Posição: 5 - diagonal frontal inferior esquerda	Posição: 30 - no meio do espaço
Posição: 6 - diagonal frontal inferior esquerda	Posição: 31 - no meio do espaço
Posição: 7 - diagonal frontal inferior esquerda	Posição: 32 - diagonal traseira superior direita

Posição: 8 - diagonal frontal inferior esquerda	Posição: 33 - diagonal traseira superior direita
Posição: 9 - diagonal frontal inferior esquerda	Posição: 34 - diagonal traseira superior direita
Posição: 10 - diagonal frontal inferior esquerda	Posição: 35 - diagonal traseira superior direita
Posição: 11 - diagonal frontal inferior esquerda	Posição: 36 - diagonal traseira superior direita
Posição: 12 - diagonal frontal inferior esquerda	Posição: 37 - diagonal traseira superior direita
Posição: 13 - diagonal frontal inferior esquerda	Posição: 38 - diagonal traseira superior direita
Posição: 14 - diagonal frontal inferior esquerda	Posição: 39 - diagonal traseira superior direita
Posição: 15 - diagonal frontal inferior esquerda	Posição: 40 - diagonal traseira superior direita
Posição: 16 - diagonal frontal inferior esquerda	Posição: 41 - diagonal traseira superior direita
Posição: 17 - diagonal frontal inferior esquerda	Posição: 42 - diagonal traseira superior direita
Posição: 18 - diagonal frontal inferior esquerda	Posição: 43 - diagonal traseira superior direita
Posição: 19 - no meio do espaço	Posição: 44 - diagonal traseira superior direita
Posição: 20 - no meio do espaço	Posição: 45 - diagonal traseira superior direita
Posição: 21 - no meio do espaço	Posição: 46 - diagonal traseira superior direita
Posição: 22 - no meio do espaço	Posição: 47 - diagonal traseira superior direita
Posição: 23 - no meio do espaço	Posição: 48 - diagonal traseira superior direita
Posição: 24 - no meio do espaço	Posição: 49 - diagonal traseira superior direita
Posição: 25 - no meio do espaço	Posição: 50 - diagonal traseira superior direita

Pela listagem de posições, vê-se que a Lógica Fuzzy também operou corretamente.

#### 4.1.2 – Rastreamento do Objeto Sobre o Logaritmo

Nesta simulação o objeto alvo se move segundo uma trajetória com comportamento logarítmico. Os valores da variável T foram incrementados de “1” para que os pontos se deslocassem sobre a trajetória e o valor do baricentro era governado pela seguinte equação:

$$X(T) = T \quad Y(T) = LN(T) \quad Z(T) = T \quad (14)$$

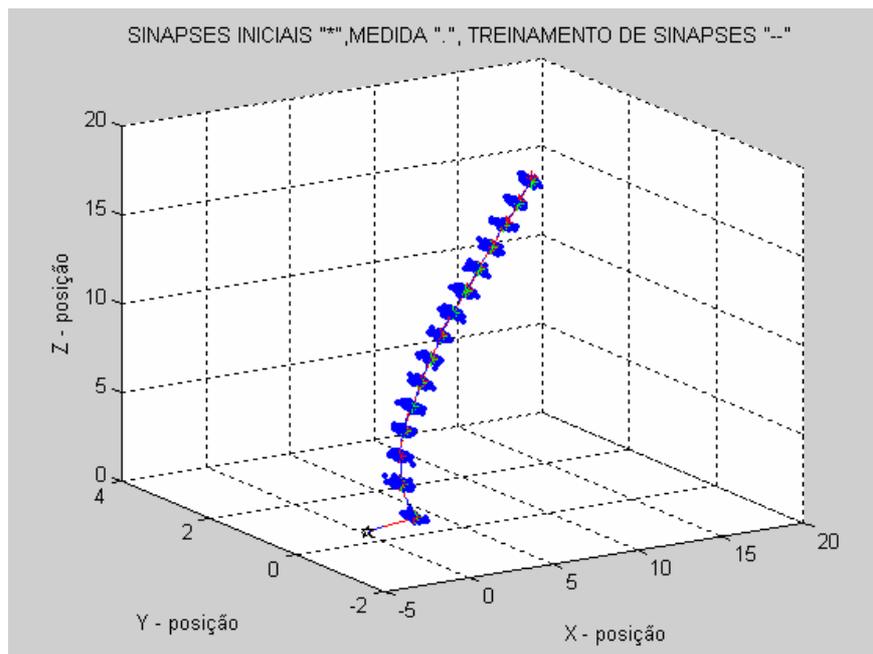


Figura 12 - Resultado do algoritmo com trajetória logarítmica.

A figura 12 mostra os resultados da validação do algoritmo, e, a partir da mesma pôde-se analisar o comportamento do sistema.

Pela figura, é possível observar que tanto a rede neural quanto o Filtro de Kalman se comportam como foi previamente estipulado. Apesar de esta ser uma trajetória curva, o algoritmo funcionou como esperado. Em seguida o sistema difuso foi aplicado, obtendo-se:

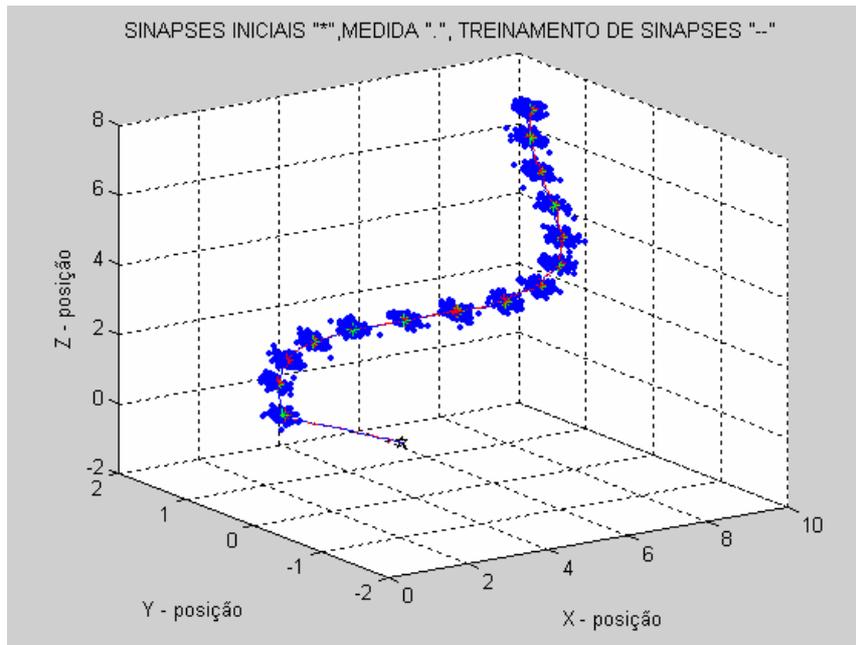
Posição: 1 - diagonal frontal inferior esquerda	Posição: 26 - no meio da profundidade e em cima
Posição: 2 - diagonal frontal inferior esquerda	Posição: 27 - no meio da profundidade e em cima
Posição: 3 - diagonal frontal inferior esquerda	Posição: 28 - no meio da profundidade e em cima
Posição: 4 - diagonal frontal inferior esquerda	Posição: 29 - no meio da profundidade e em cima
Posição: 5 - na frente e à esquerda	Posição: 30 - no meio da profundidade e em cima
Posição: 6 - na frente e à esquerda	Posição: 31 - no meio da profundidade e em cima
Posição: 7 - na frente e à esquerda	Posição: 32 - diagonal traseira superior direita
Posição: 8 - na frente e à esquerda	Posição: 33 - diagonal traseira superior direita
Posição: 9 - na frente e à esquerda	Posição: 34 - diagonal traseira superior direita
Posição: 10 - na frente e à esquerda	Posição: 35 - diagonal traseira superior direita
Posição: 11 - na frente e à esquerda	Posição: 36 - diagonal traseira superior direita
Posição: 12 - diagonal frontal superior esquerda	Posição: 37 - diagonal traseira superior direita
Posição: 13 - diagonal frontal superior esquerda	Posição: 38 - diagonal traseira superior direita
Posição: 14 - diagonal frontal superior esquerda	Posição: 39 - diagonal traseira superior direita
Posição: 15 - diagonal frontal superior esquerda	Posição: 40 - diagonal traseira superior direita
Posição: 16 - diagonal frontal superior esquerda	Posição: 41 - diagonal traseira superior direita
Posição: 17 - diagonal frontal superior esquerda	Posição: 42 - diagonal traseira superior direita
Posição: 18 - diagonal frontal superior esquerda	Posição: 43 - diagonal traseira superior direita
Posição: 19 - no meio da profundidade e em cima	Posição: 44 - diagonal traseira superior direita
Posição: 20 - no meio da profundidade e em cima	Posição: 45 - diagonal traseira superior direita
Posição: 21 - no meio da profundidade e em cima	Posição: 46 - diagonal traseira superior direita
Posição: 22 - no meio da profundidade e em cima	Posição: 47 - diagonal traseira superior direita
Posição: 23 - no meio da profundidade e em cima	Posição: 48 - diagonal traseira superior direita
Posição: 24 - no meio da profundidade e em cima	Posição: 49 - diagonal traseira superior direita
Posição: 25 - no meio da profundidade e em cima	Posição: 50 - diagonal traseira superior direita

Pela listagem acima, pode-se concluir que o sistema difuso funcionou corretamente.

#### 4.1.3 – Rastreamento do Objeto Sobre o Seno

Nesta simulação, o objeto alvo se move com uma trajetória com comportamento senoidal. A forma de construção da simulação foi igual a da simulação anterior, exceto que nessa, o alvo percorre uma trajetória senoidal.

$$X(T) = T \quad Y(T) = \text{Sen}(T) \quad Z(T) = T \quad (15)$$



**Figura 13 - Resultado do algoritmo com trajetória senoidal.**

A figura 13 mostra que o rastreamento foi bem sucedido como o rastreamento das outras trajetórias. Mesmo com grandes variações em algumas regiões, o Filtro de Kalman conseguiu convergir para o sistema dinâmico, que é a Rede Neural Kohonen. Em seguida o sistema difuso foi aplicado à rede, obtendo-se a lista abaixo, a qual mostra que este sistema também trabalhou apropriadamente para este caso.

Posição: 1 - diagonal frontal inferior esquerda	Posição: 26 - no meio da profundidade e em cima
Posição: 2 - diagonal frontal inferior esquerda	Posição: 27 - no meio da profundidade e em cima
Posição: 3 - diagonal frontal inferior esquerda	Posição: 28 - no meio da profundidade e em cima
Posição: 4 - diagonal frontal inferior esquerda	Posição: 29 - no meio da profundidade e em cima
Posição: 5 - na frente e à esquerda	Posição: 30 - no meio da profundidade e em cima
Posição: 6 - na frente e à esquerda	Posição: 31 - no meio da profundidade e em cima
Posição: 7 - na frente e à esquerda	Posição: 32 - diagonal traseira superior direita
Posição: 8 - na frente e à esquerda	Posição: 33 - diagonal traseira superior direita
Posição: 9 - na frente e à esquerda	Posição: 34 - diagonal traseira superior direita
Posição: 10 - na frente e à esquerda	Posição: 35 - diagonal traseira superior direita
Posição: 11 - na frente e à esquerda	Posição: 36 - diagonal traseira superior direita
Posição: 12 - diagonal frontal superior esquerda	Posição: 37 - diagonal traseira superior direita
Posição: 13 - diagonal frontal superior esquerda	Posição: 38 - diagonal traseira superior direita
Posição: 14 - diagonal frontal superior esquerda	Posição: 39 - diagonal traseira superior direita
Posição: 15 - diagonal frontal superior esquerda	Posição: 40 - diagonal traseira superior direita
Posição: 16 - diagonal frontal superior esquerda	Posição: 41 - diagonal traseira superior direita
Posição: 17 - diagonal frontal superior esquerda	Posição: 42 - diagonal traseira superior direita
Posição: 18 - diagonal frontal superior esquerda	Posição: 43 - diagonal traseira superior direita
Posição: 19 - no meio da profundidade e em cima	Posição: 44 - diagonal traseira superior direita
Posição: 20 - no meio da profundidade e em cima	Posição: 45 - diagonal traseira superior direita
Posição: 21 - no meio da profundidade e em cima	Posição: 46 - diagonal traseira superior direita
Posição: 22 - no meio da profundidade e em cima	Posição: 47 - diagonal traseira superior direita
Posição: 23 - no meio da profundidade e em cima	Posição: 48 - diagonal traseira superior direita
Posição: 24 - no meio da profundidade e em cima	Posição: 49 - diagonal traseira superior direita
Posição: 25 - no meio da profundidade e em cima	Posição: 50 - diagonal traseira superior direita

#### 4.1.4 – Rastreamento do Objeto com Mais de Uma Trajetória

Nesta seção, a simulação ocorreu com mais de uma trajetória ao mesmo tempo. Essas trajetórias são governadas por equação matemáticas diferentes.

Para validar o algoritmo foi considerada uma situação na qual dois objetos se movem segundo trajetórias diferentes e governadas pelas seguintes equações:

$$X1(T) = 1 + T \quad Y1(T) = 2 * \text{sen}(T) + 1 \quad Z1(T) = T \quad (16)$$

$$X2(T) = T \quad Y2(T) = 0,3 * T \quad Z2(T) = T \quad (17)$$

Para mostrar claramente o efeito que o Filtro de Kalman traz ao algoritmo, foram feitas duas simulações com os mesmos objetos. Primeiro foi usada apenas a Rede Neural Kohonen e depois foram usados a Rede Neural Kohonen e o Filtro de Kalman juntos.

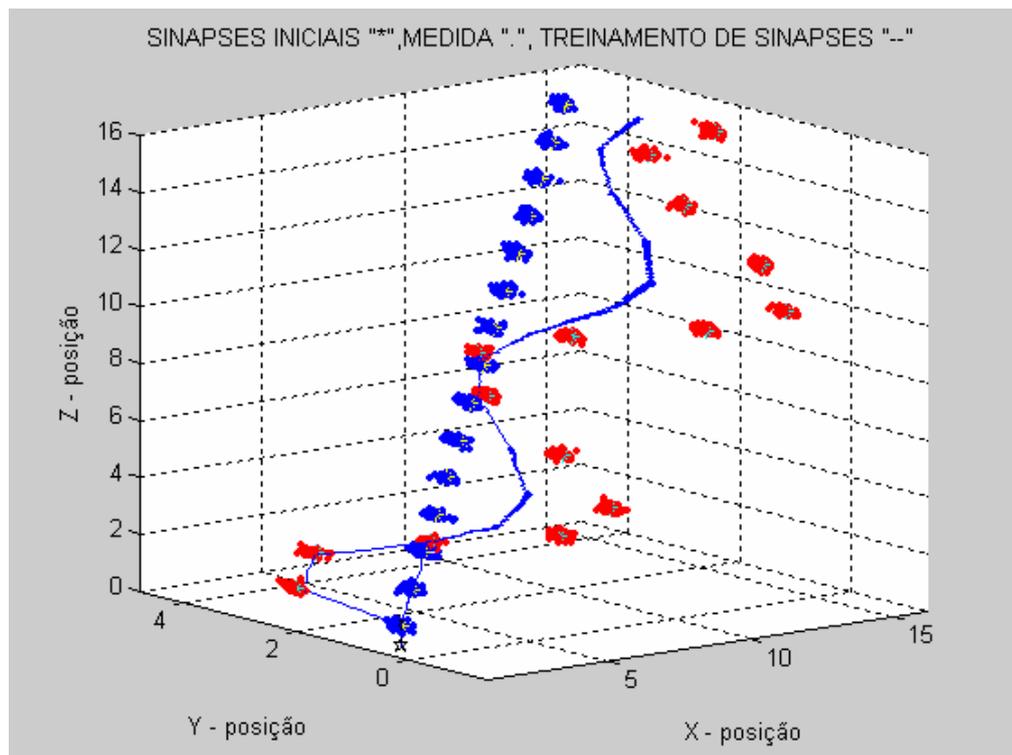


Figura 14 - Rastreamento mal sucedido de dois objetos.

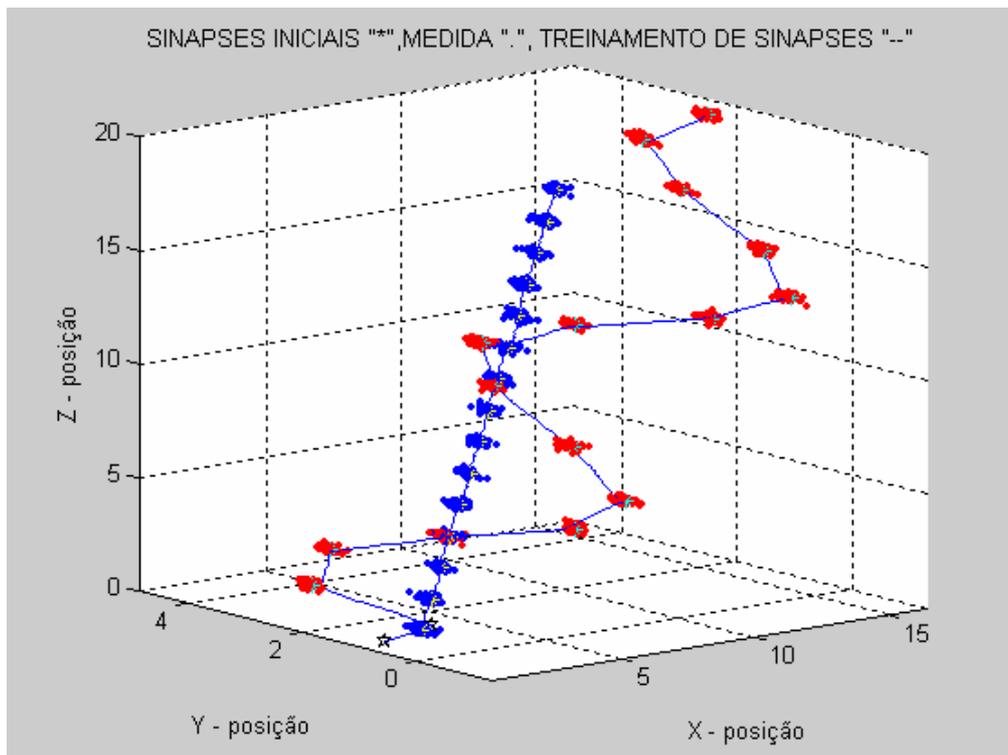
Sem o Filtro de Kalman o rastreamento pode falhar como aconteceu no rastreamento ilustrado na figura 14, isso acontece porque ao chegar nos cruzamentos entre as trajetórias, as entradas passam a pertencer à mesma classe, o que faz os neurônios trocarem de trajetória. Em seguida, foi aplicado o sistema difuso, obtendo-se a lista a seguir:

Posição: 1 - diagonal frontal inferior esquerda	Posição: 26 - no meio da profundidade e em baixo
Posição: 2 - diagonal frontal inferior esquerda	Posição: 27 - no meio da profundidade e em baixo
Posição: 3 - diagonal frontal inferior esquerda	Posição: 28 - no meio do espaço
Posição: 4 - diagonal frontal inferior esquerda	Posição: 29 - no meio do espaço
Posição: 5 - diagonal frontal inferior esquerda	Posição: 30 - no meio da profundidade e em baixo
Posição: 6 - diagonal frontal inferior esquerda	Posição: 31 - no meio da profundidade e em baixo
Posição: 7 - diagonal frontal inferior esquerda	Posição: 32 - no meio da profundidade e em baixo
Posição: 8 - diagonal frontal inferior esquerda	Posição: 33 - atrás e à direita
Posição: 9 - diagonal frontal inferior esquerda	Posição: 34 - atrás e à direita
Posição: 10 - diagonal frontal inferior esquerda	Posição: 35 - atrás e à direita
Posição: 11 - diagonal frontal inferior esquerda	Posição: 36 - atrás e à direita
Posição: 12 - diagonal frontal inferior esquerda	Posição: 37 - atrás e à direita
Posição: 13 - diagonal frontal inferior esquerda	Posição: 38 - atrás e à direita
Posição: 14 - diagonal frontal inferior esquerda	Posição: 39 - atrás e à direita
Posição: 15 - diagonal frontal inferior esquerda	Posição: 40 - atrás e à direita
Posição: 16 - diagonal frontal inferior esquerda	Posição: 41 - atrás e à direita
Posição: 17 - diagonal frontal inferior esquerda	Posição: 42 - atrás e à direita
Posição: 18 - diagonal frontal inferior esquerda	Posição: 43 - atrás e à direita
Posição: 19 - diagonal frontal inferior esquerda	Posição: 44 - atrás e à direita
Posição: 20 - diagonal frontal inferior esquerda	Posição: 45 - atrás e à direita
Posição: 21 - no meio da profundidade e em baixo	Posição: 46 - atrás e à direita
Posição: 22 - no meio da profundidade e em baixo	Posição: 47 - atrás e à direita
Posição: 23 - no meio da profundidade e em baixo	Posição: 48 - atrás e à direita
Posição: 24 - no meio da profundidade e em baixo	Posição: 49 - atrás e à direita
Posição: 25 - no meio da profundidade e em baixo	Posição: 50 - atrás e à direita

Como o objetivo do sistema difuso é descrever o comportamento do neurônio, pode-se dizer que ele operou corretamente, porém a descrição gerada não é válida, pois o rastreamento falhou. Esse exemplo também serve para deixar claro, que o sistema difuso criado é dependente da rede, aplicando a descrição aos neurônios.

A seguir, é mostrada a figura 15, a qual ilustra o rastreamento com as mesmas trajetórias anteriores, só que agora, com a rede neural e filtro.

Pode-se observar pela figura 15 que o rastreamento foi bem sucedido. O Filtro de Kalman inserido como uma componente a mais da Rede Neural Kohonen promove à rede uma prevenção à trocas de trajetórias pelos neurônios. Em seguida, foi aplicado o sistema difuso, obtendo-se a lista logo após a figura 15, a qual descreve corretamente o comportamento dos neurônios.



**Figura 15 - Rastreamento bem sucedido de dois objetos.**

Para a trajetória senoidal obteve-se:

Posição: 1 - diagonal frontal inferior esquerda	Posição: 26 - no meio da profundidade e em baixo
Posição: 2 - diagonal frontal inferior esquerda	Posição: 27 - no meio da profundidade e em baixo
Posição: 3 - diagonal frontal inferior esquerda	Posição: 28 - no meio da profundidade e em baixo
Posição: 4 - diagonal frontal inferior esquerda	Posição: 29 - no meio da profundidade e em baixo
Posição: 5 - diagonal frontal inferior esquerda	Posição: 30 - no meio da profundidade e em baixo
Posição: 6 - diagonal frontal inferior esquerda	Posição: 31 - no meio da profundidade e em baixo
Posição: 7 - diagonal frontal inferior esquerda	Posição: 32 - diagonal traseira inferior direita
Posição: 8 - diagonal frontal inferior esquerda	Posição: 33 - diagonal traseira inferior direita
Posição: 9 - diagonal frontal inferior esquerda	Posição: 34 - diagonal traseira inferior direita
Posição: 10 - diagonal frontal inferior esquerda	Posição: 35 - diagonal traseira inferior direita
Posição: 11 - diagonal frontal inferior esquerda	Posição: 36 - diagonal traseira inferior direita
Posição: 12 - diagonal frontal inferior esquerda	Posição: 37 - diagonal traseira inferior direita
Posição: 13 - diagonal frontal inferior esquerda	Posição: 38 - diagonal traseira inferior direita
Posição: 14 - diagonal frontal inferior esquerda	Posição: 39 - diagonal traseira inferior direita
Posição: 15 - diagonal frontal inferior esquerda	Posição: 40 - atras e à direita
Posição: 16 - diagonal frontal inferior esquerda	Posição: 41 - atras e à direita
Posição: 17 - diagonal frontal inferior esquerda	Posição: 42 - atras e à direita
Posição: 18 - diagonal frontal inferior esquerda	Posição: 43 - atras e à direita
Posição: 19 - no meio da profundidade e em baixo	Posição: 44 - atras e à direita
Posição: 20 - no meio da profundidade e em baixo	Posição: 45 - atras e à direita
Posição: 21 - no meio da profundidade e em baixo	Posição: 46 - atras e à direita
Posição: 22 - no meio da profundidade e em baixo	Posição: 47 - atras e à direita
Posição: 23 - no meio da profundidade e em baixo	Posição: 48 - atras e à direita
Posição: 24 - no meio da profundidade e em baixo	Posição: 49 - atras e à direita
Posição: 25 - no meio da profundidade e em baixo	Posição: 50 - atras e à direita

Para a trajetória linear obteve-se:

Posição: 1 - diagonal frontal inferior esquerda	Posição: 26 - no meio do espaço
Posição: 2 - diagonal frontal inferior esquerda	Posição: 27 - no meio do espaço
Posição: 3 - diagonal frontal inferior esquerda	Posição: 28 - no meio do espaço
Posição: 4 - diagonal frontal inferior esquerda	Posição: 29 - no meio do espaço
Posição: 5 - diagonal frontal inferior esquerda	Posição: 30 - no meio do espaço
Posição: 6 - diagonal frontal inferior esquerda	Posição: 31 - no meio do espaço
Posição: 7 - diagonal frontal inferior esquerda	Posição: 32 - diagonal traseira superior direita
Posição: 8 - diagonal frontal inferior esquerda	Posição: 33 - diagonal traseira superior direita
Posição: 9 - diagonal frontal inferior esquerda	Posição: 34 - diagonal traseira superior direita
Posição: 10 - diagonal frontal inferior esquerda	Posição: 35 - diagonal traseira superior direita
Posição: 11 - diagonal frontal inferior esquerda	Posição: 36 - diagonal traseira superior direita
Posição: 12 - diagonal frontal inferior esquerda	Posição: 37 - diagonal traseira superior direita
Posição: 13 - diagonal frontal inferior esquerda	Posição: 38 - diagonal traseira superior direita
Posição: 14 - diagonal frontal inferior esquerda	Posição: 39 - diagonal traseira superior direita
Posição: 15 - diagonal frontal inferior esquerda	Posição: 40 - diagonal traseira superior direita
Posição: 16 - diagonal frontal inferior esquerda	Posição: 41 - diagonal traseira superior direita
Posição: 17 - diagonal frontal inferior esquerda	Posição: 42 - diagonal traseira superior direita
Posição: 18 - diagonal frontal inferior esquerda	Posição: 43 - diagonal traseira superior direita
Posição: 19 - no meio do espaço	Posição: 44 - diagonal traseira superior direita
Posição: 20 - no meio do espaço	Posição: 45 - diagonal traseira superior direita
Posição: 21 - no meio do espaço	Posição: 46 - diagonal traseira superior direita
Posição: 22 - no meio do espaço	Posição: 47 - diagonal traseira superior direita
Posição: 23 - no meio do espaço	Posição: 48 - diagonal traseira superior direita
Posição: 24 - no meio do espaço	Posição: 49 - diagonal traseira superior direita
Posição: 25 - no meio do espaço	Posição: 50 - diagonal traseira superior direita

## **5 – CONCLUSÃO E SUGESTÃO PARA TRABALHOS FUTUROS**

### **5.1 – Conclusão**

Neste trabalho, foi proposto e estudado um mecanismo que realiza a identificação e o rastreamento através da Rede Neural Kohonen em conjunto com o Filtro de Kalman e que realiza a descrição lingüística dos objetos-alvos usando a Lógica Fuzzy.

Para tentar concretizar a união dessas três abordagens foi realizado um estudo sobre os tipos de rede neurais existentes, suas formas de treinamento e aplicações. Também foi realizado um estudo sobre o Filtro de Kalman e suas aplicações e sobre a Lógica Fuzzy e sistemas de decisão baseados nela. Também foram pesquisados artigos relacionados a qualquer um dos três conceitos.

Para validar o mecanismo foi usado o software Matlab®, no qual foram simuladas trajetórias a partir de funções matemáticas. Como as validações foram efetuadas usando alvos que se locomoviam tridimensionalmente, foi necessário criar um sistema de decisão difuso tridimensional, análogo ao que foi criado pelo ex-bolsista Denys de Souza Scheiner [63]. No rastreamento de uma trajetória apenas, foi constatado que a rede, o filtro e a descrição lingüística operaram com sucesso e para o rastreamento com mais de uma trajetória, a rede sozinha não conseguiu fazer sua tarefa corretamente, porém com ajuda do filtro os neurônios conseguiram rastrear as trajetórias e a Lógica Fuzzy também foi aplicada com êxito.

### **5.2 – Sugestões Para Trabalhos Futuros**

Outros trabalhos podem ser desenvolvidos usando a mesma teoria já estudada, porém adicionando novos recursos que tornem o sistema de rastreamento melhor ou modificando-o através de outras ferramentas. Assim os trabalhos futuros sugeridos são:

1. Estudo teórico e implementação de outras formas do Filtro de Kalman em conjunto com a Rede Neural Kohone e a Lógica Fuzzy, como o Filtro de Kalman estendido, que pode ser aplicado no caso de trajetórias com grandes variações de direção, o Filtro de Kalman-Bucy e outras variações.
2. Estudo teórico e implementação de outros tipos de redes neurais como a Back-propagation, a Perceptron e a Hopfield para realizaram o rastreamento de alvos móveis.
3. Adição de diferentes técnicas de inteligência artificial a Rede Neural Kohonen para realizar o rastreamento de alvos móveis.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Krose, Ben; Smagt, Patrick Van Der – **An Introduction to Neural Networks**, eighth edition.
- [2] Wu, S. J., Lin, C. T., **Global Optimal Fuzzy Tracker Design Based on Local Concept Approach**, IEEE Transaction On Fuzzy Systems, 10 (2): 128-143, 2002.
- [3] Schram, G., vanderLinden, F.X., Krose, B.J.A., Groen, F.C.A., **Visual tracking of moving objects using a neural network controller**, Robotics And Autonomous Systems, 18 (3): 293-299 AUG 1996.
- [4] Gaudiano, P., Zalama, E., Coronado, J.L., **An unsupervised neural network for low-level control of a wheeled mobile robot: Noise resistance, stability, and hardware implementation**, IEEE Transactions On Systems Man And Cybernetics Part B-Cybernetics, 26 (3): 485-496 JUN 1996.
- [5] Chen, Y. M., Hsueh, C.S., **Complementary Data Fusion in Vision-guide and Control of Robotic Tracking**, Robotica, 19: 53-58 Part 1, 2001.
- [6] Ma, C.W., Teng, C.C., **Tracking a near-field moving target using fuzzy neural networks**, Fuzzy Sets And Systems, 110 (3): 365-377 MAR 16 2000.
- [7] Yu, X., Azimi-Sadjadi, M.R., **Neural network directed bayes decision rule for moving target classification**, IEEE Trans. on Aerospace and Electronic Systems, 36 (1): 176-188, 2000.
- [8] Yang, S. X., Meng, M., **Neural Network Approaches to Dynamic Collision-Free Trajectory Generation**, IEEE Transaction on System Man and Cybernetic Part B-Cybernetics, 31 (3): 302-318, 2001.
- [9] Cameron, S., Grossberg, S., Guenther, F.H., **A self-organizing neural network architecture for navigation using optic flow**, Neural Computation, 10 (2): 313-352, 1998.
- [10] Missler, J.M., Kamangar, F.A., **A Neural-Network For Pursuit Tracking Inspired By The Fly Visual-System**, Neural Networks, 8 (3): 463-480 1995.
- [11] Dror, I. E., Florer, F. L., Rios, D., Zagaeski, M., **Using Artificial Bat Sonar Neural Networks for Complex Pattern Recognition: Recognizing Faces and the Speed of a Moving Target**, Biological Cybernetics, 74 (4): 331-338, 1996.
- [12] Navabi, H., Agarwal, A., **Adaptative Response Organizer Network for Space-time Patterns in Low Level Vision**, Neural Networks, 11 (5): 825-836, 1998.
- [13] Calôba, Luis P., **Introdução à Computação Neuronal**, IX Congresso Brasileiro de Automação, 1992, Vitória, ES. Mini-curso 2. p. 25-38.
- [14] Yi Li, Songde Ma, Hanqing Lu, **Human posture recognition using multi-scale morphological method and Kalman motion estimation**, Proceedings of the IEEE International Conference on Pattern Recognition, 1998, pp. 175-177.
- [15] Pereira, Guilherme A. S., **Filtro de Kalman: Teoria e Aplicações**, CPDEE - UFMG
- [16] Nascimento Jr.,C. L.; Yoneyama, Takashi – **Inteligência Artificial em Controle e Automação**, Ed. Edgard Blucher Ltda, 2000.
- [17] Kovács, Zsolt L. – **Redes Neurais Artificiais: Fundamentos e Aplicações**, 2ª ed São Paulo: Edição Acadêmica, 1996.
- [18] Masters, Timothy – **Practical Neural Network Recipes in C++**, Academic Press.
- [19] LiMin Fu – **Neural Networks in Computer Intelligence**, University of Florida, Gainesville.

- [20] Rojas, Raúl – **Neural Networks: A Systematic Introduction**, Springer.
- [21] Balakrishnan, S. N.; Weil, R.D. – **Neurocontrol: A Literature Survey**, Elsevier Science Ltd, 1996.
- [22] Agbinya, Johnson I.; Rees, David – **Multi-Object Tracking in Video**, Academic Press, 1999.
- [23] Matthews, N.D., Na, P.E., Charnley, D., Harris, C.J., **Vehicle detection and recognition in greyscale imagery**, *Control Engineering Practice*, 4 (4): 473-479 APR 1996.
- [24] Kocjancic, R., Zupan, J. **Application of a feed-forward artificial neural network as a mapping device**, *Journal Of Chemical Information And Computer Sciences*, 37 (6): 985-989 Nov-Dec 1997.
- [25] Tipping, M.E., Lowe, D., **Shadow targets: A novel algorithm for topographic projections by radial basis functions**, *Neurocomputing*, 19 (1-3): 211-222 Mar 1998.
- [26] Amoozegar, F., **Neural-network-based target tracking state-of-the-art survey**, *Optical Engineering*, 37 (3): 836-846, 1998.
- [27] Wann, C.D., Thomopoulos, S.C.A., **Application of self-organizing neural networks to multiradar data fusion**, *Optical Engineering*, 36 (3): 799-813, 1997.
- [28] Yu, N., Wu, H., Wang, R.S., **Morphological neural networks with applications to image target detection**, *Chinese Journal Of Electronics*, 10 (1): 85-90, 2001.
- [29] Shams, S. **Neural Network Optimization for Multi-Target Multi-Sensor Passive Tracking**, proceedings of the IEEE, vol. 84, No. 10, October, 1996.
- [30] Gurney, K., **Neural Nets**, Dpt. Human Sciences, Brunel University, Uxbridge, Middx, UK.
- [31] Fu, LiMin, **Neural Networks in Computer Intelligence**, c1994 by McGraw-Hill, Inc.
- [32] Junior, B.M., Anido, R.O., **Object Detection with Multiple Cameras**, CAPES and PRONEX-SAI, grant number 2346.
- [33] Fioretti, S., **Signal processing in movement analysis (a state-space approach)**, Ancona: Elsevier Science B.V., c1996.
- [34] Fagiani, C., **An Evaluation of Tracking Methods for Human-Computer Iteration**, Senior Thesis 2002, Boston College – Computer Science Department.
- [35] Roux, J. Le, **An introduction to Kalman filter**, University of Nice, 3 janiver 2003.
- [36] Çavusoglu, M.C., Tendick, F., **Kalman Filter Analysis for Quantitative Comparison of Sensory Schemes in Bilateral Teleoperation Systems**, Presence, December 2002.
- [37] Welch, G., Bishop, G., **An Introduction to the Kalman Filter**, Copyright 2001 by ACM, Inc. Disponível em: <http://www.cs.unc.edu/~welch>.
- [38] Gelb, A., **Applied Optimal Estimation**, M.I.T. Press 1996.
- [39] Brown, R.G., Hwang, P.Y.C., Wiley, J., and Sons, **Introduction to Random Signals and Applied Kalman Filtering**, Inc. New York, 1992.
- [40] Shet, K.C., Rao, B.V., **An Overview Of The Kalman Algorithm**, *International Journal Of Electronics*, 59 (6). 657-665 Dec. 1985.
- [41] Willet, D., Busch, C., Siebert, F., - **Fast Image Analysis Using Kohonen Maps**, Darmstadt Computer Graphics Center.
- [42] Feldkamp, L. A., Prokhorov, D. V., Eagen, C. F., Yuan, F. – **Enhanced Multi-Stream Kalman Filter Training for Recurrent Networks**, For Research Laboratory, Michigan, USA.
- [43] Cheng, H.D., Jiang, X.H., Sum, Y, Wang, J. – **Color image segmentation: advances and**

**prospects**, Department of Computer Science, Utah State University, Logan, USA.

- [44] Cho, Sung-Bae – **Pattern Recognition With Neural Networks Combined by Genetic Algorithm**, Department of Computer Science, Yonsei University, Seoul, South Korea.
- [45] Srihari, S. N., **Representation of Three-Dimensional Digital Images**, Department of Computer Science, State University of New York at Buffalo.
- [46] Srinivasan, R., Kinser, J. M. – **A Foveating Fuzzy Scoring Target Recognition System**, Center for Applied Optical Sciences, Department of Physics, Alabama A&M University.
- [47] Brijnesh, J. Jain, Wysotzki, Fritz, **Competitive Winner-Takes-All Clustering in the Domain of Graphs**, Department of Computer Science, Technical University of Berlin, Germany.
- [48] Vaidehi, V., Chitra, N., Chokkalingam, M., Krishnan, C.N. – **Neural Network Aided Kalman Filtering for multitarget target applications**, School of Instrumentation and Electronics, Madras Institute of Technology, India. 20 May 1999.
- [49] Zadeh, Lofti A. – **Fuzzy Logic**, University of California, Berkeley.
- [50] Kojima, et al., **Generating Natural Language Description of Human Behaviors from Video Images**, Proceedings of the International Conference on Pattern Recognition, 2000, pp.728-731.
- [51] P. Remagnino, T. Tan, K. Baker, **Agent orientated annotation in model based visual surveillance**, Proceedings of the International Conference on Computer Vision, 1998, pp.857–862.
- [52] S. Intille, A. Bobick, **Representation and visual recognition of complex, multi-agent actions using belief networks**, Technical Report 454, Perceptual Computing Section, MIT Media Lab, 1998.
- [53] G. Herzog, K. Rohr, **Integrating vision and language: towards automatic description of human movements**, Proceedings of the 19th Annual German Conference on Artificial Intelligence, 1995, pp. 257–268.
- [54] A. Penland, A. Liu, **Modeling and prediction of human behaviors**, Neural Comput. 11 (1999) 229–242.
- [55] M. Thonnat, N. Rota, **Image understanding for visual surveillance applications**, Proceedings of the Third
- [56] N. Rota, M. Thonnat, **Video sequence interpretation for visual surveillance**, Proceedings of the Workshop on Visual Surveillance, Ireland, 2000, pp. 59–67.
- [57] H. Kollnig, H. H. Nagel, and M. Otte, **“Association of motion verbs with vehicle movements extracted from dense optical flow fields,”** in *Proc. European Conf. Computer Vision*, 1994, pp. 338–347.
- [58] G. Herzog and P. Wazinski, **“Visual translator: linking perceptions and natural language descriptions,”** *Artific. Intell. Rev.*, vol. 8, no. 2, pp. 175–187, 1994.
- [59] Araujo, E., – **Entendendo a Lógica Fuzzy aplicada a Controle de Processos e a Sistemas de Decisão.**
- [60] Garcia, Francis; Filho, José E. de A. – **Estudo de Redes Neurais Artificiais na Identificação de Alvos Móveis em Processamento Digital de Imagens**, LIT, INPE, Maio de 2003.
- [61] Silva, Cassiano Rodrigo; Filho, José E. de A. – **Estudo de Redes Neurais Artificiais na Identificação de Alvos Móveis com Auxílio do Filtro de Kalman**, LIT, INPE, Junho de 2004.
- [62] Silva, Cassiano Rodrigo; Filho, José E. de A. – **Redes Neurais e Filtro de Kalman para**

**Rastreamento de Alvos com Trajetórias Tridimensionais**, LIT, INPE, Junho de 2005.

- [63] Scheiner, Denys de Souza; Filho, José E. de A. – **Descrição Semântica através de Modelagem Difusa (Fuzzy) para Rastreamento de Alvos Moveis**, LIT, INPE, Agosto de 2006.
- [64] Wilson, D. Randall; Martinez, Tony R. – **The Need for Small Learning Rates on Large Problems**, In Proceedings of the 2001 International Joint Conference on Neural Networks (IJCNN'01), 115-119
- [65] Pham, D. T.; Liu, Xing – **Neural Networks for Identification, Prediction and Control**, pp. 1-3, 17-18.