

# **UM SISTEMA DE INFERÊNCIA ADAPTATIVO NEURO-FUZZY NO RECONHECIMENTO DE IMAGENS COMO APLICAÇÃO NA VISÃO COMPUTACIONAL**

**Luiz Eduardo Nicolini do Patrocínio Nunes**  
UNITAU - Departamento de Engenharia Mecânica  
Laboratório de Informática  
Taubaté – SP Brasil  
E-mail: [lgpnunes@uol.com.br](mailto:lgpnunes@uol.com.br)

**Victor Orlando Gamarra-Rosado**  
UNESP – Campus de Guaratinguetá  
Departamento de Mecânica, CP 205  
12516-410 Guaratinguetá – SP Brasil  
E-mail: [victor@feg.unesp.br](mailto:victor@feg.unesp.br)

**Resumo.** Este trabalho propõe o emprego de um sistema de inferência neuro-fuzzy na classificação de objetos de formas geométricas contidos em imagens bidimensionais. A tarefa de reconhecimento de imagens foi dividida em duas etapas: pré-processamento e classificação através de um sistema neuro-fuzzy. Na etapa de pré-processamento, as imagens foram processadas por um algoritmo de detecção de bordas, utilizando o método de Sobel. As imagens binárias resultantes desse algoritmo foram, então, processadas pela Transformada de Fourier log-polar, para tornar o sistema invariante a efeitos de translação, rotação e escala. Os resultados do algoritmo log-polar foram as entradas para o sistema fuzzy. Das imagens pré-processadas, duas amostras foram utilizadas para treinar o sistema, e as restantes, para os testes de classificação. Os resultados do trabalho demonstraram que o sistema de inferência neuro-fuzzy pode ser empregado como ferramenta eficiente na tarefa de reconhecimento de formas geométricas.

**Palavras-chave:** Sistema de inferência neuro-fuzzy, Visão computacional, Método de Sobel.

## **1. INTRODUÇÃO**

Cada técnica de inteligência artificial possui características que a torna adequada a determinadas aplicações. Por causa do alto paralelismo na sua estrutura, as redes neurais têm resposta mais rápida e com melhor desempenho do que os computadores seqüenciais na emulação do cérebro humano. Assim processamentos de grandes quantidades de dados tornam-se fáceis mediante *hardware* adequado.

Por outro lado, a lógica *fuzzy* é uma poderosa ferramenta para modelar o pensamento e a percepção humana. O raciocínio humano é de alguma forma nebuloso, pois é capaz de trabalhar com dados incertos, incompletos e ambíguos. As imagens na vida real, freqüentemente contêm dados que são incompletos ou ambíguos. A utilidade da lógica *fuzzy* está na sua habilidade de tratar esses dados que são encontrados freqüentemente na prática.

Os sistemas híbridos combinam duas ou mais técnicas de maneira a aproveitar suas vantagens, e também superar algumas limitações que as técnicas individuais possuírem. Assim, para habilitar um sistema a manipular situações da vida real de uma forma mais parecida com a do ser humano, uma forma é incorporar os conceitos dos conjuntos *fuzzy* dentro de redes neurais.

Este trabalho propõe a implementação de um simulador, através do *software* Matlab (Mathworks Inc., 1999), de um sistema de visão computacional, utilizando um sistema adaptativo *neuro-fuzzy* para o reconhecimento de objetos em imagens bidimensionais. Inicialmente, as imagens foram submetidas a um pré-processamento com base em um algoritmo de detecção de bordas (Método de Sobel), para transformar o cenário de fundo em preto, mantendo, apenas, o contorno do objeto a ser reconhecido. Em seguida, foi aplicada a Transformada de Fourier Log-Polar (Loesch, 1998), com o objetivo de manter as características identificadoras do objeto independentes de sua translação, rotação e escala. As entradas do sistema *neuro-fuzzy* foram as matrizes geradas pela Transformada Log-Polar. Este sistema tem a função de classificar a imagem apresentada.

## 2. SISTEMA DE INFERÊNCIA FUZZY

O Sistema de Inferência *Fuzzy* (FIS) é o processo que formula o mapeamento de uma dada entrada em uma saída usando lógica *fuzzy*. Esse mapeamento fornece a base na qual decisões podem ser tomadas ou padrões reconhecidos. O processo de inferência *fuzzy* envolve funções de pertinência, operadores lógicos *fuzzy* e regras *if-then*.

Os FIS têm sido aplicados nas mais diversas áreas, tais como: controle automático, classificação de dados, análise de decisão, sistemas especialistas e visão computacional [4].

### 2.1. Arquitetura ANFIS

A combinação da Lógica *Fuzzy* e Redes Neurais Artificiais (RNA's) permite todas as vantagens de ambos os sistemas (Garzon et al, 2001). A arquitetura ANFIS (*Adaptative Neural-Fuzzy Inference System*) foi proposta com o objetivo de modelar o raciocínio *fuzzy*. O real poder do sistema ANFIS é sua habilidade de representar regras do tipo proposto por Takagi e Sugeno:

**Regra 1:** Se  $x$  é  $A_1$  e  $y$  é  $B_1$ , então  $f_1 = p_1x + q_1y + r_1$

**Regra 2:** Se  $x$  é  $A_2$  e  $y$  é  $B_2$ , então  $f_2 = p_2x + q_2y + r_2$

sendo  $x$  e  $y$  variáveis de entrada,  $A$  e  $B$  são funções de pertinência,  $f$  é a variável de saída,  $p$ ,  $q$  e  $r$  são parâmetros consequentes.

A estrutura ANFIS deste raciocínio é mostrada na Figura 1.

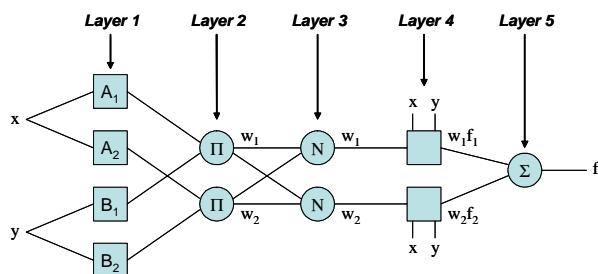


Figura 1. Estrutura ANFIS

Na camada (*layer*) 1 é realizada a *fuzzificação* da entrada. Todos os nós nesta camada contêm funções de pertinência. Normalmente, as funções triangulares ou gaussianas são escolhidas (Grigore et al, 2001). A função para o  $i$ -ésimo nó pode ser uma função de pertinência do tipo gaussiana:

$$O_i^1 = \mu_{A_i}(x) = e^{-\frac{1}{2}\left(\frac{x-c_i}{a_i}\right)^2} \quad (1)$$

onde  $O_i^j$  denota a saída do  $i$ -ésimo nó na  $j$ -ésima camada,  $x$  é a entrada para o nó  $i$ ,  $A_i$  são vetores de entrada associados ao  $i$ -ésimo nó e  $\{a_i, b_i, c_i\}$  são seus conjuntos de parâmetros que mudam a forma da função de pertinência.

Na camada 2 é computada a ativação das regras, ou seja é realizada a parte antecedente das regras, e é realizada através das  $t$ -normas para modelar o operador OR.

$$O_i^2 = w_i = \mu_{A_i}(x) + \mu_{B_i}(y) - \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1, 2 \quad (2)$$

Na camada 3 é realizada a normalização dos valores de ativação das regras e é calculada por:

$$O_i^3 = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2 \quad (3)$$

A camada 4 é referente à parte consequente da regra, calculando o produto do nível de ativação  $z$  normalizado pela saída das respectiva regra:

$$O_i^4 = \bar{w}_i f_i = w_i(p_i x + q_i y + r_i) \quad (4)$$

Na camada 5 é realizada a soma de todas as saídas, fornecendo a resposta da rede.

$$O_i^5 = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i f_i} \quad (5)$$

### 3. AQUISIÇÃO DE DADOS E PRÉ-PROCESSAMENTO

O simulador do sistema de visão computacional tem o objetivo de classificar, através de um sistema adaptativo neuro-fuzzy, objetos de formas geométricas diversas, independente de suas variações em relação à rotação, translação e escalonamento. As classes de objetos utilizadas nesse trabalho foram o quadrado, o círculo, o triângulo e o retângulo, sendo que, para cada classe, foram adquiridas, através de um *scanner*, quarenta (40) imagens com posições, rotações e escalas diferentes. Todas as imagens foram gravadas no formato *BMP* para posterior utilização. Foram utilizadas duas imagens de cada classe para treinar o sistema, e as outras, para a classificação propriamente dita.

O pré-processamento da imagem constituiu na transformação do cenário de fundo em preto (permanecendo apenas o contorno do objeto a ser identificado) através de um algoritmo de detecção de bordas, utilizando o método de Sobel e na aplicação de uma transformada discreta baseada na transformação log-polar, para permitir a invariância em relação à translação, rotação e escala, permanecendo apenas as características identificadoras do objeto.

### 4. PROJETO DO CLASSIFICADOR NEURO-FUZZY

Para este trabalho, foi utilizada a interface gráfica ANFIS, que adota o tipo Sugeno, do programa Matlab. Foi criado um conjunto de treinamento contendo duas amostras de cada classe de objetos. Quatro amostras de cada classe foram separadas para os testes.

Em primeiro lugar foram definidas as funções de pertinência para cada classe de entrada (quadrado, círculo, triângulo e retângulo), conforme ilustra a Figura 2. Foi adotada a função do tipo gaussiana por ter apresentado melhores resultados que a do tipo triangular.

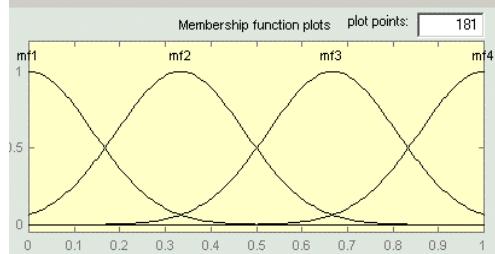


Figura 2. Funções de pertinência para as entradas do sistema

O ambiente ANFIS oferece dois tipos de funções de pertinência para a saída do sistema: linear e constante. Para este trabalho, foi adotado o tipo constante. A seguir, foram definidas as regras de ativação, utilizando-se o operador *OR* e peso 1 para as conexões (Fig. 3).

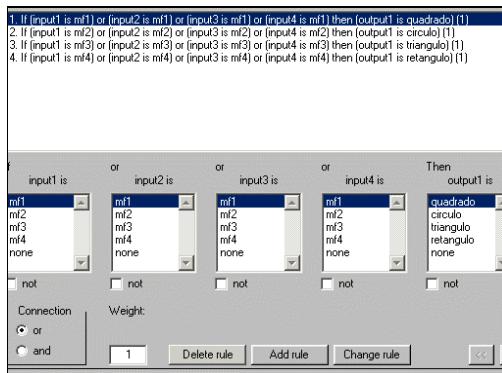


Figura 3. Regras de ativação

O sistema oferece dois modos de treinamento: *Backpropagation* ou híbrido, que é uma mistura dos métodos dos Mínimos Quadrados e *Backpropagation*. O sistema foi treinado com o método *Backpropagation* e 800 épocas, conforme ilustra a Figura 4. Após aproximadamente 50 épocas, a taxa de erro se estabilizou na faixa de 6%.

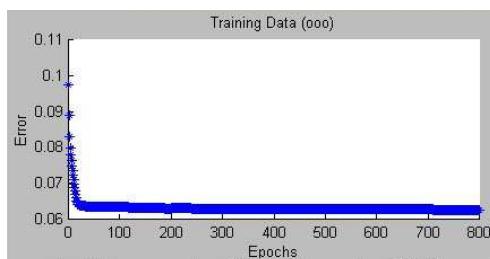


Figura 4. Treinamento do sistema

A Figura 5 apresenta: a) o conjunto de treinamento e b) o resultado gerado pelo sistema após o treinamento.

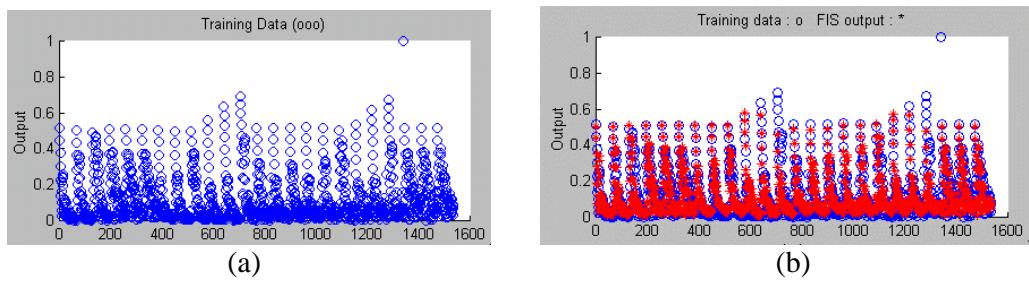


Figura 5. (a) Conjunto de treinamento (duas amostras de cada classe), (b) Sistema treinado

A Figura 6 ilustra o ajuste sofrido pelas funções de pertinência das entradas após o treinamento do sistema.

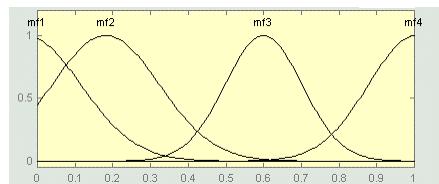


Figura 6. Funções de pertinência ajustadas após o treinamento

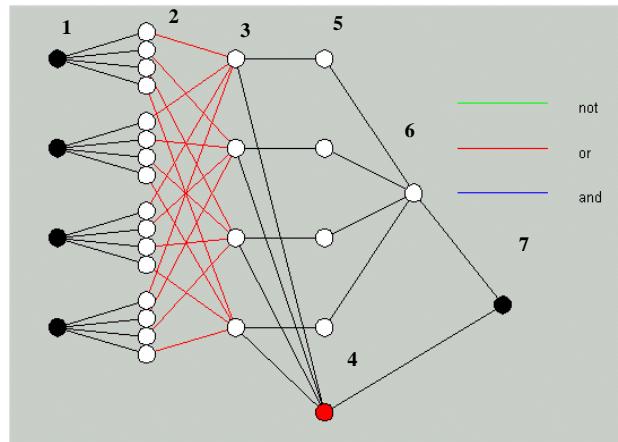


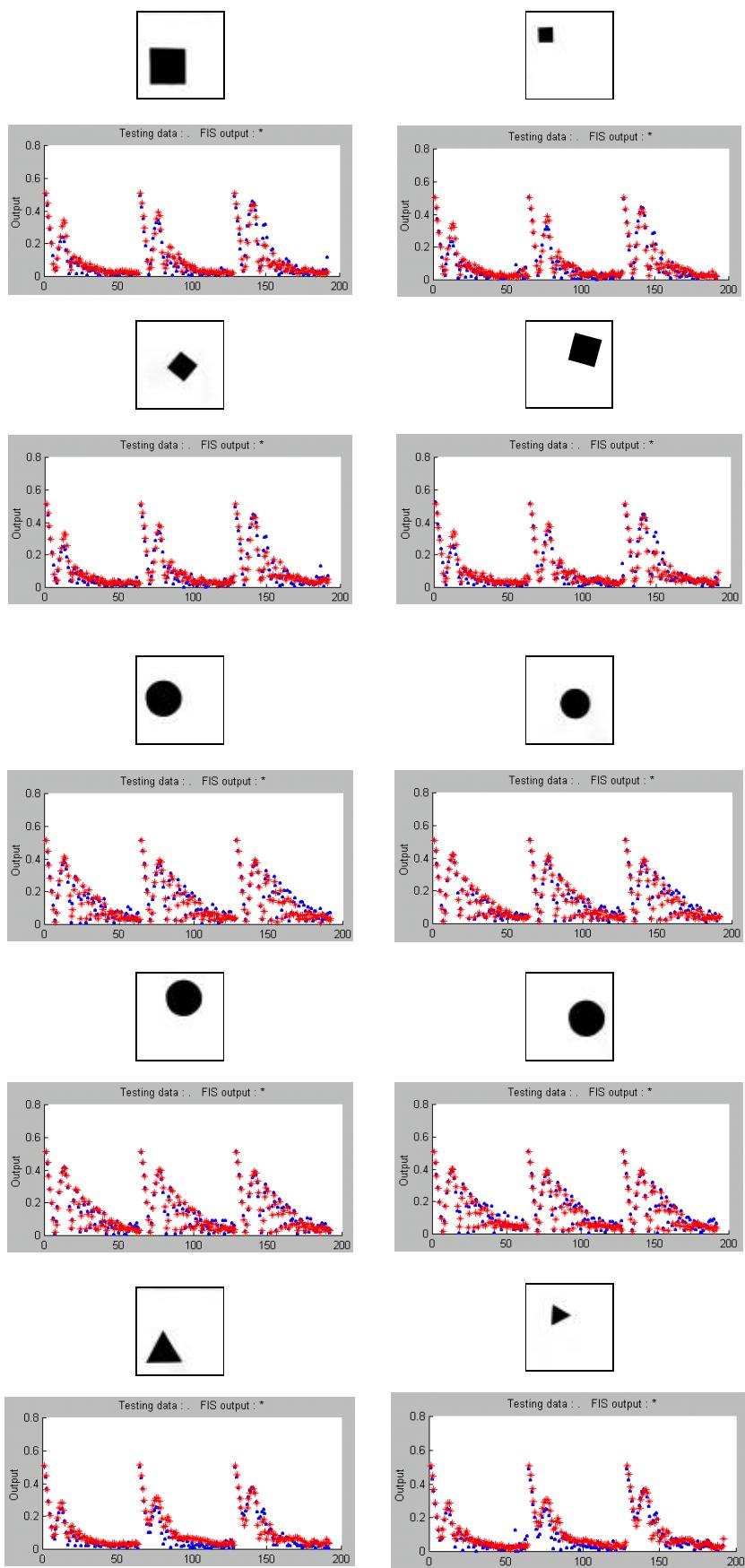
Figura 7. Estrutura do sistema

A Figura 7 ilustra a estrutura do sistema, sendo as camadas descritas e enumeradas a seguir:

- 1 – Entradas do Sistema;
- 2 – Fuzzificação das Entradas (funções de pertinência);
- 3 – Ativação das Regras;
- 4 – Fator de Normalização;
- 5 – Funções de Pertinência para as Saídas;
- 6 – Soma Ponderada das Saídas;
- 7 – Saída do Sistema.

## 5. RESULTADOS

Devido à grande quantidade de imagens adquiridas, 160 no total, foram utilizadas quatro imagens de cada classe para a apresentação dos resultados obtidos. O sistema obteve bons resultados nos testes, com taxas de erro variando entre 1% e 10%, conforme ilustra a Figura 8.



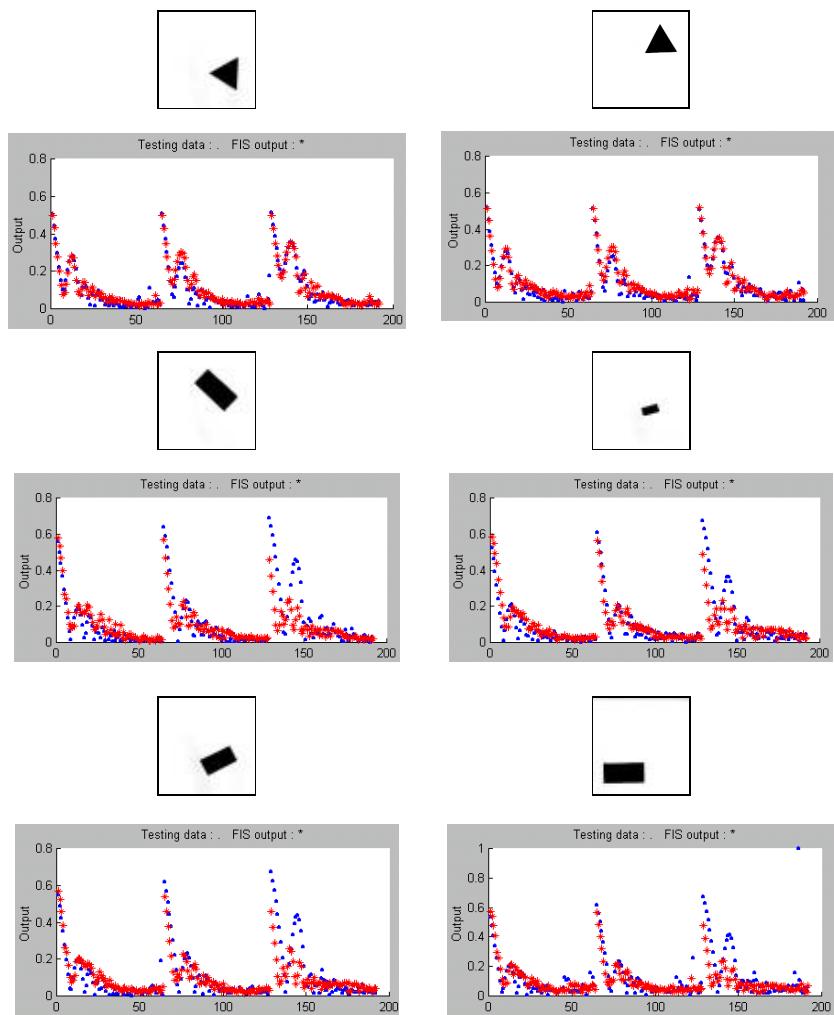


Figura 8. Resultados obtidos

## 6. CONCLUSÕES

Conforme os gráficos apresentados, os resultados foram considerados satisfatórios, atingindo, assim, o objetivo proposto pelo trabalho;

A aplicação de sistemas neuro-fuzzy para a classificação de imagens é viável;

Trata-se de um sistema que pode ser implementado e utilizado em aplicações reais.

## 7. REFERÊNCIAS

- Garzon, M.H., Ankaraju, P., Drumwright, E., Kozma, R., 2001, "Neurofuzzy Recognition and Generation of Facial Features in Talking Heads", Computer Science, University of Memphis, USA.
- Grigore, O., Florescu, A., Vasile, A., Stoicescu, V.A., 2001, "Fuzzy or Neuro-Fuzzy Design in DC Motor Speed Control? Part B: Neuro-Fuzzy Design For DC Motor Speed Control. University of Bucharest, Eletronics and Telecommunications Faculty, Bucharest, Romania.
- Loesch, C., 1998, "Reconhecimento de Objetos em Imagens Bidimensionais". Revista Ciências Exatas, Taubaté, Vol. 4, No. 1/2, p.31-8.
- The Mathworks Inc. Fuzzy Logic Toolbox – User's Guide, 1999.

## **8. DIREITOS AUTORAIS**

Os autores são os únicos responsáveis pelo conteúdo do material impresso incluído no seu trabalho.

### **AN ADAPTATIVE NEURO-FUZZY INFERENCE SYSTEM IN THE IMAGES RECOGNITION AS APPLICATION IN THE COMPUTATIONAL VISION**

**Luiz Eduardo Nicolini do Patrocínio Nunes**

UNITAU - Departamento de Engenharia Mecânica  
Laboratório de Informática  
Taubaté – SP Brasil  
E-mail: [lgpnunes@uol.com.br](mailto:lgpnunes@uol.com.br)

**Victor Orlando Gamarra-Rosado**

UNESP – Campus de Guaratinguetá  
Departamento de Mecânica, CP 205  
12516-410 Guaratinguetá – SP Brasil  
E-mail: [victor@feg.unesp.br](mailto:victor@feg.unesp.br)

***Abstract.** This paper shows the employment of a neuro-fuzzy inference system for the geometric shapes classification contained in bidimensional images. The task of recognition of images was divided in two stages: pre-processing and classification through a neuro-fuzzy system. In the pre-processing stage, the images were processed by an borders detection algorithm, using the Sobel method, that eliminated the image background, just leaving the contour of the object to be recognized.. The resulting binary images of that algorithm were, then, processed by the Fourier log-polar transform, to make the system invariant to translation, rotation and scale effects. The results of the log-polar algorithm were the inputs for the fuzzy system. Two samples of the processed images were used to train the system, and the remaining ones, for the classification tests. The results of the work demonstrated that the neuro-fuzzy inference system can be used as efficient tool in the geometric shapes recognition task.*

**Key-words.** Neuro-fuzzy inference system, Computational vision, Sobel method.