

# Modelo de Identificação de Plantas Usando Algoritmos Genéticos e Rede Perceptron Multicamadas

Thiago H. Borges<sup>1</sup>, Gilberto A. Carrijo<sup>1</sup>, Edna L. Flôres<sup>1</sup>,  
Antônio C. P. Veiga<sup>1</sup>, Milena B. P. Carneiro<sup>1</sup>

<sup>1</sup>Faculdade de Engenharia Elétrica – Universidade Federal de Uberlândia (UFU)  
CEP – 38408-454 – Uberlândia – MG – Brasil

thiagohborges@msn.com, {gilberto,edna,acpveiga}@ufu.br, milenabueno@yahoo.com

**Abstract.** *The biodiversity of species existing in the plant kingdom make the use of traditional models of taxonomy a process very complex and slow. This paper presents a hybrid system for identifying plant. This system is composed by Genetic Algorithm (GA) and Artificial Neural Network (ANN). The role played by the GA is to perform a preselection of plants forming a group that the answer of an unknown leaf is more probable and the purpose of ANN, trained by backpropagation algorithm, is to classify the unknown leaf performing the search only in the group calculated by the AG. Several tests were conducted and the results obtained demonstrate that the hybrid system has a satisfactory recognition rate compared with other architectures.*

**Resumo.** *A biodiversidade de espécies existentes no reino vegetal tornam os modelos tradicionais de taxonomia uma tarefa muito complexa e lenta. Este trabalho apresenta um sistema híbrido e automático de taxonomia de plantas. Esse sistema é composto por Algoritmos Genéticos (AG) e pela Rede Neural Artificial (RNA). O objetivo do AG é realizar uma pré-seleção de plantas formando um grupo cuja resposta da folha desconhecida é a mais provável e a finalidade da RNA, treinada pelo algoritmo backpropagation, é classificar a folha considerando apenas o grupo calculado pelo AG. Vários testes foram realizados e os resultados obtidos mostram que o sistema híbrido apresenta uma taxa de reconhecimento satisfatória se comparada com outros sistemas.*

## 1. Introdução

A identificação de espécies vegetais é uma importante, porém, difícil tarefa para o pesquisador, principalmente devido à biodiversidade das espécies. Este processo que vai desde a aquisição dos exemplares até a lenta etapa de comparação com as amostras já catalogadas exige um grande esforço por parte dos taxonomistas e/ou profissionais de áreas correlatas.

A identificação de uma planta pode ser realizada observando características tais como: flores, caule, frutos, raízes e sementes, porém, devido à complexidade geométrica dessas características, a mais simples de ser analisada na planta é a folha, pois ela é composta de uma estrutura polimórfica e planar. Esses fatores facilitam a transferência de sua imagem para um computador, por meio de fotografias ou digitalização.

Metodologias de visão computacional em conjunto com técnicas de reconhecimento de padrões vêm sendo desenvolvidas em procedimentos automatizados de reconhecimento da planta. [Wu et al. 2007] utilizaram a rede neural probabilística (PNN)

para classificar 32 tipos de folhas. A taxa de reconhecimento obtida por eles foi de 90,31%. Outros estudiosos como [Wang et al. 2005] desenvolveram um extrator de características que analisa a geometria das folhas e um classificador baseado em hiperesfera para identificar 20 espécies. Nesse trabalho, a taxa de reconhecimento obtida por eles foi de 92%. Já [Uluturk and Ugur 2012] desenvolveram um novo método de extração de características e utilizaram a rede PNN, para classificar as mesmas folhas do banco de dados de [Wu et al. 2007]. O resultado obtido por eles foi de 92,5%. Na identificação de 20 espécies de plantas [Gu et al. 2005] combinaram a transformada *wavelet* e a interpolação gaussiana para extrair a borda e as nervuras das folhas. Para classificar essas folhas, eles utilizaram os métodos do vizinho mais próximo, do inglês *1-nearest neighbor* (1-NN), dos *k* vizinhos mais próximos (*k*-NN) e a rede neural probabilística (PNN), e obtiveram as taxas de acerto de 93,17%, 85,47% e 91,18%, respectivamente.

Em geral, todo sistema de reconhecimento de imagens pode ser dividido em dois subgrupos. O primeiro, é responsável pelo processamento das imagens e pela extração de suas características, e o segundo subgrupo, é responsável pela identificação. O desempenho desse segundo está relacionado à qualidade e a eficiência do primeiro subgrupo. Neste trabalho utilizou-se o extrator de características desenvolvido em [Wu et al. 2007]. As 12 características das folhas obtidas por esse extrator são derivadas de 5 características primárias delas: o diâmetro, o comprimento, a largura, a área e o perímetro.

A proposta deste trabalho é desenvolver um sistema de reconhecimento de plantas utilizando uma arquitetura híbrida composta por Algoritmos Genéticos (AG) e pela Rede Neural Artificial (RNA) com treinamento *backpropagation*. A finalidade do AG é realizar uma pré-seleção de plantas que possuem características de folhas semelhantes às da folha desconhecida e o objetivo da RNA, neste trabalho, é identificar a folha desconhecida realizando a busca dentro do grupo pré-selecionado pelo AG.

Este artigo está estruturado da seguinte maneira: as seções 2 e 3 mostram os fundamentos de algoritmos genéticos e redes neurais, respectivamente. A seção 4 descreve a arquitetura híbrida proposta neste trabalho. A seção 5 apresenta os resultados obtidos. E finalmente, na seção 6 são realizadas as conclusões.

## 2. Algoritmos Genéticos

Os algoritmos genéticos (AGs) foram criados em 1975 por John Holland [Holland 1975]. AGs são métodos computacionais de busca, otimização e aprendizado da máquina.

A identificação de um padrão desconhecido por meio dos AGs inicia-se com a extração de suas características. Essas características são a base da criação da população inicial. O primeiro elemento desta população possui exatamente as características extraídas da folha e os demais indivíduos são derivados desse primeiro elemento. Esta derivação é realizada da seguinte forma: 5 características são escolhidas aleatoriamente para serem mantidas conforme sua extração e as outras 7 são geradas, obedecendo uma distribuição normal de média igual ao valor da característica original e desvio padrão igual a 5% do valor da média. Este processo é repetido 59 vezes, gerando, assim, uma população inicial de 60 indivíduos.

A população inicial é avaliada pela função aptidão. A análise desta função utiliza duas tabelas. A primeira armazena os valores máximos e mínimos de cada uma

das características dos padrões de treinamento, e a segunda tabela armazena o valor mais representativo de cada uma das características das plantas conhecidas, baseada nos três vizinhos mais próximos [Duda et al. 2000]. Dessa forma, se a característica que está sendo analisada está entre os valores máximo e mínimo de uma determinada característica de uma classe, um estudo é realizado para conhecer a distância dela até o valor mais representativo. Quanto mais próximo a característica que está sendo analisada está do valor mais representativo, melhor é a sua avaliação. Assim, o valor da aptidão do indivíduo é calculado pela soma da avaliação de cada uma de suas 12 características. Esse processo de comparação é repetido entre todos os elementos do banco de dados, composto de 32 plantas. No final, o indivíduo que possui a maior aptidão é separado e rotulado com o valor da classe que ele mais se aproxima.

A próxima etapa do algoritmo AG é selecionar os pais que serão utilizados no cruzamento. O método de seleção dos pais utilizado neste artigo é o *ranking* [Linden 2008]. Esta seleção ordena todos os indivíduos conforme a sua função de aptidão e os classifica de acordo com um *ranking*, ao invés de usar diretamente o valor da aptidão. O objetivo de utilizar este método de seleção é o de evitar a convergência prematura e o aparecimento de um superindivíduo.

As 12 características presentes nos indivíduos que formam a população do AG podem assumir qualquer valor real maior do que zero. Assim, para aplicar os operadores genéticos, cada uma das características dos indivíduos selecionados para o cruzamento são binarizados em 25 bits. Dessa forma, os cromossomos dessa população depois da codificação passam a ser são compostos por 300 genes ( $12 \times 25$ ).

O operador de cruzamento implementado neste trabalho é o *crossover* de dois pontos com probabilidade de cruzamento de 90%. Em cada um dos pais esse cruzamento escolhe dois pontos de corte, sendo que esses pontos devem ser múltiplos de 25 para não separar os genes de uma mesma característica. Assim, o primeiro filho é formado pela parte do primeiro pai fora dos pontos de corte e pela parte do segundo pai entre os pontos de corte. O segundo filho é formado pelas partes restantes.

O operador de mutação foi empregado com a probabilidade de ocorrência de 10%, com a finalidade de garantir a continuidade da existência de diversidade genética na população. O funcionamento desse operador baseia-se na troca de bits do cromossomo. Nesta troca, 3 bits da cadeia do cromossomo são escolhidos aleatoriamente para serem invertidos.

O ciclo do AG termina na conversão de binário para decimal de cada um dos genes e na reavaliação de todos os novos indivíduos pela função aptidão. Com o objetivo de estimular o crescimento do AG na direção da resposta foi empregada a técnica do elitismo. Esta técnica substitui o pior indivíduo da população corrente pelo melhor indivíduo da população anterior.

### 3. Redes Neurais Artificiais

O algoritmo de aprendizado mais conhecido para o treinamento de redes neurais perceptron multicamadas, do inglês *MultiLayer Perceptron* (MLP), é o *backpropagation*.

O treinamento da MLP utilizando o algoritmo *backpropagation* é realizado em duas fases, em que cada uma delas percorre a rede em um sentido. Estas fases são de-

nominadas *forward* e *backward*. A *forward* é usada para definir a saída da rede para um determinado padrão de entrada. A *backward* utiliza as saídas desejada e calculada pela rede para atualizar os pesos de suas conexões [Fausett 1994].

A fim de garantir uma rápida convergência da resposta, a inicialização dos pesos e dos *bias* no algoritmo *backpropagation* não deve ser feita de forma totalmente aleatória, pois dependendo dos valores assumidos por eles, as funções de ativação ou suas derivadas podem se tornar iguais a zero. Desta forma, os valores iniciais não devem ser muito grandes, pois eles terão a finalidade de evitar que a derivada da função de ativação entre na região de saturação e assumam valor muito pequeno. Por outro lado, se os pesos iniciais são muito pequenos, o somatório de entrada das camadas pode assumir valores próximos de zero, promovendo um aprendizado muito lento.

Após vários testes com diferentes métricas de inicialização, o critério que obteve os melhores resultados para o problema analisado neste artigo é inicializar os pesos pelo critério de Nguyen e Widrow [Nguyen and Widrow 1990]. Este critério consiste em inicializar os pesos para a camada de saída com valores aleatórios entre  $-0,5$  e  $0,5$  e os pesos para a camada oculta com valores que estimulem um aprendizado mais rápido dos neurônios dessa camada. Este estímulo é definido primeiramente calculando o fator de escala  $\beta$  como mostrado na Equação (1).

$$\beta = 0.7 \sqrt{p} \quad (1)$$

onde:

- $n$  - número de entradas; e
- $p$  - número de neurônios da camada oculta.

Após calcular o fator de escala  $\beta$  cada peso que chega à camada oculta ( $W_{ij}(0)$ ) deve ser inicializado com valores aleatórios entre  $-0,5$  e  $0,5$ . Finalmente, o valor inicial dos pesos é encontrado pela Equação (2).

$$W_{ij} = \frac{\beta W_{ij}(0)}{|W_{ij}(0)|} \quad (2)$$

onde:

- $|W_{ij}(0)|$  - módulo do vetor  $W_{ij}(0)$ .

Os *bias* dos neurônios das camadas oculta ( $v_{0j}$ ) e de saída ( $\omega_{0k}$ ) são inicializados com valores aleatórios entre  $-\beta$  e  $\beta$  e entre  $-0,5$  e  $0,5$ , respectivamente.

Outro parâmetro importante que contribui para a eficiência de uma RNA é a sua função de ativação. Essa função deve ser contínua, diferenciável e monotonicamente não decrescente. Com o objetivo de obter uma melhor eficiência computacional, a derivada da função de ativação deve ser fácil de calcular [Fausett 1994]. Assim, após vários testes utilizando diferentes funções de ativação, identificou-se que a que obteve os melhores resultados para este trabalho foi a função logarítmica definida pela Equação (3).

$$f(x) = \begin{cases} \log(1+x) & \text{para } x > 0 \\ -\log(1-x) & \text{para } x < 0 \end{cases} \quad (3)$$

Ao derivar a Equação (3) obtém-se a Equação (4).

$$f'(x) = \begin{cases} \frac{1}{1+x} & \text{para } x > 0 \\ \frac{1}{1-x} & \text{para } x < 0 \end{cases} \quad (4)$$

O principal problema de uma rede *backpropagation* é o seu elevado tempo de treinamento. Uma forma de minimizar este problema é considerar os efeitos de segunda ordem para o gradiente descendente. Outro problema indesejável nessa rede e que ocorre frequentemente é a convergência para mínimos locais. Estes mínimos são pontos na superfície de erro que apresentam uma solução estável, embora não sejam as saídas mais adequadas para a RNA. Para minimizar esses problemas, foi introduzido o termo *momentum* à equação de atualização de pesos da camada oculta, obtendo-se a Equação (5). O *momentum* acelera o treinamento da rede em regiões muito planas da superfície de erro e suprime a oscilação de pesos em vales e ravinas [Morales 2010].

$$W_{ij}(t+1) = W_{ij}(t) + \eta \delta_j(t) x_i(t) + \alpha (W_{ij}(t) - W_{ij}(t-1)) \quad (5)$$

onde:

- $\eta$  - termo *momentum* que deve assumir valores entre 0 e 1.

#### 4. Método Proposto

As redes neurais podem ser utilizadas em reconhecimento de padrões, como por exemplo, no reconhecimento de voz, caracteres, imagens, etc. Os AGs podem ser usados em problemas de otimização, como por exemplo, roteamento, escalonamento de datas e horários, etc. Neste trabalho foi desenvolvida uma arquitetura híbrida composta de AG e rede neural *backpropagation* com o objetivo de identificar plantas por meio de suas folhas. A Figura 1 mostra o diagrama de blocos dessa arquitetura.

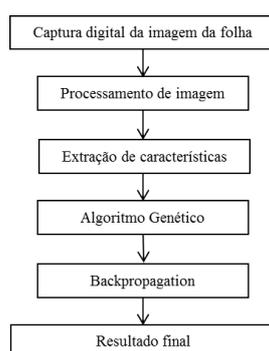


Figura 1. Esquema da arquitetura proposta

O processo de reconhecimento da arquitetura híbrida mostrado na Figura 1 inicia-se com a captura digital da imagem da folha, seguida pelas etapas de processamento e extração de características do padrão a ser reconhecido. Essas etapas são discutidas detalhadamente em [Wu et al. 2007]. Em seguida, o AG seleciona um grupo de classes cuja resposta do padrão a ser reconhecido é a mais provável e a RNA classifica o padrão, produzindo o resultado final.

A escolha por empregar os algoritmos genéticos com pré-seletor foi motivada após a observação dos resultados dos vários testes realizados somente com o AG atuando como classificador, em que verificou-se que a classe correta de uma folha desconhecida estava frequentemente presente na população final desse algoritmo, porém, ela não era apresentada como a melhor solução.

Para adaptar o AG para realizar a função de pré-seletor foi necessário apenas alterar o critério de parada do algoritmo, pois observou-se que não havia necessidade de o número de ciclos do AG ser alto, pois, com o avanço das épocas, a população final tende a ser uniforme, com poucas classes de plantas sendo representadas. Dessa forma, para satisfazer a situação em que a população inicial é homogênea e necessita ser diversificada por meio dos cruzamentos e das mutações do AG, como também para a situação contrária em que a população inicial é muito heterogênea e é necessário eliminar as piores classes de plantas, o critério de parada do AG foi alterado para o momento em que o número de indivíduos da população se torna menor ou igual a 5, ou o número de ciclos do algoritmo for maior do que 5.

Finalmente, a rede *backpropagation* é treinada para classificar a folha desconhecida, considerando apenas as classes obtidas pelo AG com a vantagem de que a procura pela resposta final ocorre em um universo de busca mais enxuto, que antes era de 32 classes de plantas e após a pré-seleção passou a ser de no máximo 5 classes.

Em resumo, para cada folha que se deseja fazer o reconhecimento por meio da arquitetura proposta por este trabalho, é necessário que o sistema realize a pré-seleção das folhas (executado pelo AG), tomando como base os padrões disponíveis para o treinamento, para, em seguida, a RNA ser treinada com as classes resultantes da etapa anterior. Dessa forma, não existe a possibilidade de criar RNAs especializadas em conjuntos de classes, pois na etapa de pré-seleção não se pode garantir que sua saída seja sempre a mesma para um mesmo padrão desconhecido.

A Tabela 1 mostra os parâmetros utilizados para construir o AG e a RNA na arquitetura híbrida proposta.

**Tabela 1. Parâmetros do AG e da RNA**

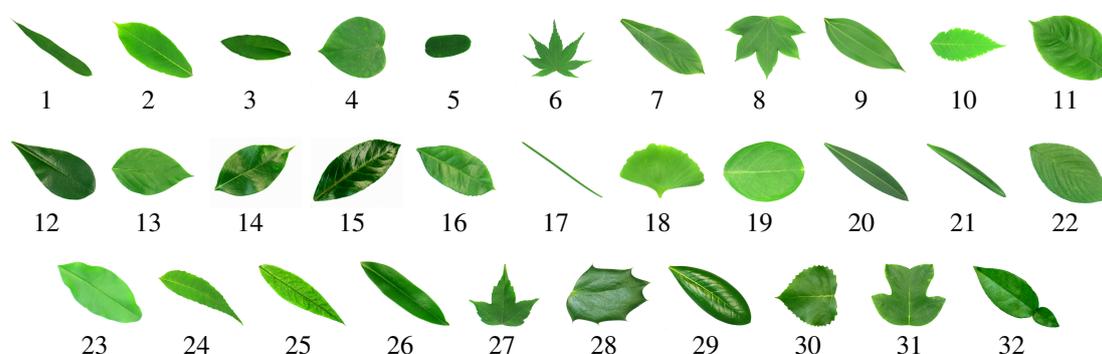
Parâmetros do AG	Valores	Parâmetros da RNA	Valores
Número de ciclos	25	Termo <i>momentum</i> ( $\eta$ )	0,3
Número de genes	12	Taxa de aprendizagem ( $\alpha$ )	0,6
Número de bits na binarização	25	Número de neurônios na camada oculta	60
Tamanho da população	60	Erro máximo acumulado	0,05
Probabilidade de cruzamento	90%		
Elitismo	1		
Probabilidade de mutação	10%		

## 5. Resultados Obtidos

Este trabalho foi desenvolvido utilizando o software MATLAB versão 7.8.0 e o mesmo banco de dados de folhas usado por [Wu et al. 2007] e [Uluturk and Ugur 2012]. Este banco é constituído por folhas de plantas da região de Shanghai na China e é composto por 32 grupos de plantas totalizando 1907 folhas com resolução de 800 x 600 *pixels*, sem restrições em relação à translação, a rotação e ao escalonamento da folha. A Tabela 2 apresenta o nome científico e a quantidade dessas folhas e a Figura 2 apresenta uma folha de cada classe descrita na tabela.

**Tabela 2. Nome científico e quantidade de folhas**

	Nome Científico	Quant. de Folhas		Nome Científico	Quant. de Folhas
1	<i>Phyllostachys edulis</i>	59	17	<i>Cedrus deodara</i>	77
2	<i>Aesculus chinensis</i>	63	18	<i>Ginkgo biloba</i>	62
3	<i>Berberis anhwaiensis</i>	65	19	<i>Lagerstroemia indica</i>	61
4	<i>Cercis chiensis</i>	72	20	<i>Nerium oleander</i>	66
5	<i>Indigofera tinctoria</i>	73	21	<i>Podocarpus macrophyllus</i>	60
6	<i>Acer dalmatum</i>	56	22	<i>Prunus yedoensis</i>	55
7	<i>Phoebe zhennan</i>	62	23	<i>Ligustrum lucidum</i>	55
8	<i>Kalopanax septemlobus</i>	52	24	<i>Tonna sinensis</i>	65
9	<i>Cinnamomum japonicum</i>	55	25	<i>Prunus pérsica</i>	54
10	<i>Koelreuteria paniculata</i>	59	26	<i>Manglietia fordiana</i>	52
11	<i>Ilex macrocarpa</i>	50	27	<i>Acer buergerianum</i>	53
12	<i>Pittosporum tobira</i>	63	28	<i>Mahonia bealei</i>	55
13	<i>Chimonanthus praecox</i>	52	29	<i>Magnolia grandiflora</i>	57
14	<i>Cinnamomum camphora</i>	65	30	<i>Populus x canadensis</i>	64
15	<i>Viburnum awabuki</i>	60	31	<i>Liriodendron chinense</i>	53
16	<i>Osmanthus fragrans</i>	56	32	<i>Citrus reticulata</i>	56

**Figura 2. Amostra de folhas utilizadas neste trabalho correspondente à Tabela 2.**

Pode-se observar na Tabela 2, que as plantas possuem quantidades diferentes de folhas. Devido a isso, em todos os testes realizados neste trabalho, os padrões de treinamento por classe foram variados de 5 em 5 até alcançar o número 40, e o número de padrões desconhecidos que passam um a um pelo sistema de reconhecimento, é o complemento do número de padrões de treinamento. Por exemplo, se a classe possui 60 folhas e 5 estão sendo utilizadas como padrões de treinamento, 55 são os padrões desconhecidos.

O número de 40 padrões de treinamento por classe na penúltima seção de testes foi escolhido porque existe uma classe (*Ilex macrocarpa*), cuja quantidade de folhas é igual a 50. Como 40 padrões estão sendo utilizados para realizar o treinamento, os 10 restantes serão os padrões desconhecidos. Com esse número de padrões desconhecidos, é possível comparar os resultados obtidos neste artigo, com os resultados do trabalho de [Wu et al. 2007], que reconheceu 10 padrões por classe. Na última seção de testes, todas as classes possuem 10 padrões desconhecidos.

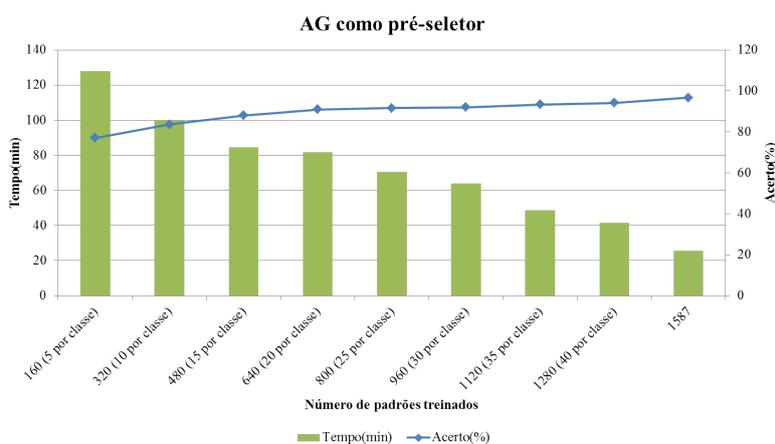
Em todos os testes executados, as folhas usadas como padrões de treinamento em cada classe foram escolhidas aleatoriamente.

A Tabela 3 e a Figura 3 mostram os resultados obtidos quando o AG foi utilizado como um pré-seletor de plantas.

Pode-se verificar na Tabela 3 e na Figura 3 que, à medida que cresceu o número de

**Tabela 3. AG utilizado como pré-seletor de plantas**

Treinamento	Probabilidade da resposta estar presente na população final do AG (%)	Tempo(min)
160 (5 por classe)	77,1	128
320 (10 por classe)	83,6	100
480 (15 por classe)	88,0	84,4
640 (20 por classe)	91,0	81,6
800 (25 por classe)	91,6	70,5
960 (30 por classe)	92,1	64,1
1120 (35 por classe)	93,4	48,8
1280 (40 por classe)	94,2	41,6
1587	96,7	25,5

**Figura 3. Gráfico dos resultados obtidos utilizando o AG como um pré-seletor de plantas.**

padrões de treinamento, aumentou a probabilidade da classe da planta da folha desconhecida estar dentro da população final do AG, como, por exemplo, para o treinamento com 5 padrões por classe essa probabilidade foi de 77,1%; já para o treinamento com 1587, padrões essa probabilidade foi de 96,7%.

Observa-se, também, nessa tabela e nessa figura que o tempo de processamento do algoritmo genético diminuiu. Isso é justificado pelo fato de que, para cada padrão desconhecido, o AG necessita realizar todas as etapas do algoritmo para cada uma das folhas desconhecidas. Por exemplo, no último teste de reconhecimento existiam apenas 10 padrões desconhecidos por classe (resultando em 320 padrões desconhecidos) o tempo para pré-selecionar as classes de maior aptidão de cada uma das folhas foi, em média, próximo ou igual a 4,8 segundos ( $\frac{25,5 \cdot 60}{320}$  segundos).

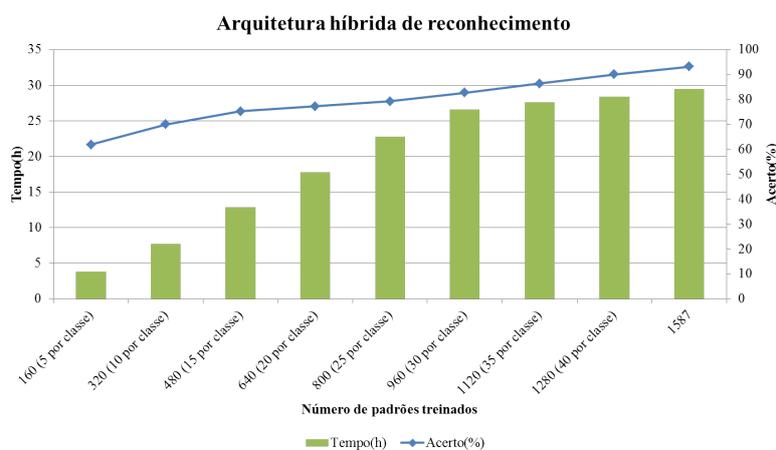
A Tabela 4 e a Figura 4 ilustram os resultados obtidos no reconhecimento das folhas das plantas, utilizando o sistema híbrido proposto neste trabalho.

Pode-se verificar na Tabela 4 e na Figura 4 que, à medida que cresceu o número de padrões de treinamento, a taxa de reconhecimento aumentou. O motivo desse aumento foi que o sistema híbrido ficou mais generalista e melhorou sua capacidade de reconhecimento.

O tempo de processamento do sistema híbrido é composto pelos tempos de processamento do AG e da RNA. O tempo de processamento do AG nesse sistema é o mesmo

**Tabela 4. Resultados obtidos no reconhecimento das folhas das plantas utilizando o sistema híbrido.**

Treinamento	Porcentagem de acerto (%)	Tempo(h)
160 (5 por classe)	61,9	3,8
320 (10 por classe)	70,0	7,7
480 (15 por classe)	75,3	12,9
640 (20 por classe)	77,2	17,8
800 (25 por classe)	79,3	22,8
960 (30 por classe)	82,7	26,6
1120 (35 por classe)	86,4	27,6
1280 (40 por classe)	90,1	28,4
1587	93,2	29,5

**Figura 4. Gráfico do reconhecimento das folhas das plantas utilizando o sistema híbrido proposto neste trabalho.**

tempo de processamento de quando o AG foi utilizado como pré-seletor de plantas, ou seja, 25,5 minutos para o caso do treinamento de 1597 padrões. A diferença observada é o tempo de treinamento da RNA que é muito menor se comparado com o tempo de treinamento de uma RNA, sem que as entradas sejam pré-selecionadas pelo AG.

Ao comparar a taxa de acerto no reconhecimento de 10 padrões do trabalho de [Wu et al. 2007] com a taxa de reconhecimento do sistema híbrido proposto neste trabalho pode-se verificar que elas foram de 90,31% e 93,2%, respectivamente e com um tempo de convergência médio de, aproximadamente, 5,5 minutos ( $\frac{29,5 \cdot 60}{320}$  minutos).

A Tabela 5 apresenta uma comparação entre o número de espécies de plantas, o algoritmo de classificação usado e a taxa de acerto obtida por cada um dos trabalhos referenciados e proposto neste artigo.

## 6. Conclusão

Para reconhecer 10 padrões por classe no sistema híbrido, o algoritmo genético atuando como pré-seletor insere um o erro de 3,3%. A consequência desse erro é que durante o processamento do AG a verdadeira classe da folha desconhecida não estará presente no grupo formado. Desse modo, conhecendo o erro inserido pelo AG, é possível inferir o erro produzido pela RNA, por meio dos gráficos apresentados. Esse erro é de 3,5%. Assim, o erro acumulado do sistema é de 6,8%, consequentemente, a taxa de reconhecimento é de

**Tabela 5. Comparação entre os métodos referenciados e proposto por este trabalho.**

Trabalho	Número de espécies	Algoritmo de reconhecimento	Taxa de acerto
[Wu et al. 2007]	32	PNN	90,31%
[Wang et al. 2005]	20	Classificador hiperesférico	92%
[Uluturk and Ugur 2012]	32	PNN	92,5%
[Gu et al. 2005]	20	1-NN	93,17%
[Gu et al. 2005]	20	k-NN	85,47%
[Gu et al. 2005]	20	PNN	91,18%
Proposta deste artigo	32	AG e RNA <i>backpropagation</i>	93,2%

93,2%, com um tempo de convergência de 5,5 minutos. Com esses números a arquitetura apresentada por este trabalho está entre os melhores desempenhos propostos na literatura.

## Referências

- Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classi.* Wiley-Interscience.
- Fausett, L. (1994). *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications.* Prentice-Hall international editions. Prentice-Hall.
- Gu, X., Du, J.-X., and Wang, X.-F. (2005). Leaf recognition based on the combination of wavelet transform and gaussian interpolation. In *Proceedings of the 2005 international conference on Advances in Intelligent Computing - Volume Part I, ICIC'05*, pages 253–262, Berlin, Heidelberg. Springer-Verlag.
- Holand, J. H. (1975). Adaptation in natural and artificial systems. *University of Michigan Press.*
- Linden, R. (2008). *Algoritmos Genéticos.* BRASPORT.
- Moraes, E. C. (2010). Reconhecimento de padrões e redes neurais artificiais em predição de estruturas secundárias de proteínas. Master's thesis, Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro.
- Nguyen, D. and Widrow, B. (1990). Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In *Proceedings of the international joint conference on neural networks*, volume 3, pages 21–26. Washington.
- Uluturk, C. and Ugur, A. (2012). Recognition of leaves based on morphological features derived from two half-regions. In *Innovations in Intelligent Systems and Applications (INISTA), 2012 International Symposium on*, pages 1–4.
- Wang, X.-F., Du, J.-X., and Zhang, G.-J. (2005). Recognition of leaf images based on shape features using a hypersphere classifier. In *Proceedings of the 2005 international conference on Advances in Intelligent Computing - Volume Part I, ICIC'05*, pages 87–96, Berlin, Heidelberg. Springer-Verlag.
- Wu, S., Bao, F., Xu, E., Wang, Y.-X., Chang, Y.-F., and Xiang, Q.-L. (2007). A leaf recognition algorithm for plant classification using probabilistic neural network. In *Signal Processing and Information Technology, 2007 IEEE International Symposium on*, pages 11–16.