

# INF1771 - INTELIGÊNCIA ARTIFICIAL

## TRABALHO 3 – APRENDIZADO DE MÁQUINA

O **objetivo do Trabalho 3** é desenvolver um sistema de reconhecimento de plantas através de imagens de folhas. O sistema deve receber como entrada a imagem de uma folha (como a mostrada na Figura 1) e ser capaz de identificar a espécie da planta a qual aquela folha pertence.



**Figura 1:** Exemplo de folha que o sistema deve ser capaz de reconhecer.

Para essa tarefa, você tem a disposição um conjunto de 1907 imagens de folhas de 32 tipos diferentes de plantas. Esse conjunto de imagens foi criado pelos autores do artigo “*A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network*” [1]. Esse artigo também apresenta o método utilizado pelos autores para realizar o processo de classificação, então é uma ótima fonte de informações para o desenvolvimento deste trabalho.

As plantas que compõem o conjunto de imagens são as seguintes:

ID	Nome	Arquivos
1	Pubescent Bamboo	1001-1059
2	Chinese Horse Chestnut	1060-1122
3	Anhui Barberry	1552-1616
4	Chinese Redbud	1123-1194
5	True Indigo	1195-1267
6	Japanese Maple	1268-1323
7	Nanmu	1324-1385
8	Castor Aralia	1386-1437
9	Chinese Cinnamon	1497-1551
10	Goldenrain Tree	1438-1496
11	Big-fruited Holly	2001-2050
12	Japanese Cheesewood	2051-2113
13	Wintersweet	2114-2165

14	Camphortree	2166-2230
15	Japan Arrowwood	2231-2290
16	Sweet Osmanthus	2291-2346
17	Geodar	2347-2423
18	Ginkgo	2424-2485
19	Crepe Myrtle	2486-2546
20	Oleander	2547-2612
21	Yew Plum Pine	2616-2675
22	Japanese Flowering Cherry	3001-3055
23	Glossy Privet	3056-3110
24	Chinese Toon	3111-3175
25	Peach	3176-3229
26	Ford Woodlotus	3230-3281
27	Trident Maple	3282-3334
28	Beale's Barberry	3335-3389
29	Southern Magnolia	3390-3446
30	Canadian Poplar	3447-3510
31	Chinese Tulip Tree	3511-3563
32	Tangerine	3566-3621

Para desenvolver o sistema de reconhecimento de plantas você deve utilizar algum método de **aprendizado de máquina supervisionado**, visto que temos um conjunto rotulado de exemplos para treinamento (imagens). Para isso, você deve seguir os seguintes passos:

- 1) Definir quais serão os atributos que serão usados para descrever os exemplos de treinamento.
- 2) Criar um programa para extrair os atributos das imagens de forma a gerar um conjunto de treinamento e um conjunto de validação. Normalmente gera-se um único conjunto de dados e depois se divide ele em dois conjuntos (treinamento e validação). Cada imagem de folha será um exemplo de treinamento.
- 3) Utilizar 4 algoritmos de aprendizado supervisionado, treinando-os com o conjunto de treinamento e depois realizando classificação do conjunto de testes para verificar qual algoritmo apresenta a melhor taxa de reconhecimento. Devem ser utilizados os seguintes algoritmos:
  - Árvores de Decisão
  - K-Nearest Neighbor (KNN)
  - Support Vector Machine (SVM)
  - Rede Neural (usando backpropagation)

Não é necessário programar os algoritmos, é permitida a utilização de ferramentas de aprendizado de máquina, como por exemplo, o Weka (<http://www.cs.waikato.ac.nz/ml/weka/>).

Se a taxa de reconhecimento estiver muito baixa para todos os algoritmos, deve-se retornar para a etapa 1 e selecionar melhor os atributos para descrever os exemplos de treinamento.

- 4) Escolher um dos 4 algoritmos e implementá-lo na linguagem de sua preferência. Árvores de Decisão e KNN são os mais simples de serem implementados. No caso do SVM, é permitido o uso da biblioteca LibSVM (<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>).

### **Programa Base**

O programa base fornecido demonstra como alguns atributos podem ser extraídos das imagens de treinamento. Os atributos extraídos pelo programa base são:

- 1) Largura do retângulo englobante do maior objeto segmentado encontrado na imagem.
- 2) Altura do retângulo englobante do maior objeto segmentado encontrado na imagem.
- 3) Perímetro do maior objeto segmentado encontrado na imagem.
- 4) Área do maior objeto segmentado encontrado na imagem.
- 5) 1º momento invariante de Hu do maior objeto segmentado encontrado na imagem.
- 6) 2º momento invariante de Hu do maior objeto segmentado encontrado na imagem.
- 7) 3º momento invariante de Hu do maior objeto segmentado encontrado na imagem.
- 8) 4º momento invariante de Hu do maior objeto segmentado encontrado na imagem.
- 9) 5º momento invariante de Hu do maior objeto segmentado encontrado na imagem.
- 10) 6º momento invariante de Hu do maior objeto segmentado encontrado na imagem.
- 11) 7º momento invariante de Hu do maior objeto segmentado encontrado na imagem.

O programa base está disponível no seguinte link:

[http://edirlei.3dgb.com.br/aulas/ia\\_2012\\_2/Trabalho3ProgramaBase.zip](http://edirlei.3dgb.com.br/aulas/ia_2012_2/Trabalho3ProgramaBase.zip)

O programa base foi desenvolvido utilizando a biblioteca de visão computacional OpenCV (<http://opencv.willowgarage.com/wiki/>). Mais detalhes sobre o funcionamento do programa base e como utilizar o OpenCV serão apresentados durante a aula.

### **Informações Adicionais:**

- Durante a etapa de teste dos 4 algoritmos não é necessário implementar todos os classificadores para realizar os experimentos. É permitida a utilização de ferramentas de aprendizado de máquina, como por exemplo, o Weka (<http://www.cs.waikato.ac.nz/ml/weka/>).

- O download das imagens de treinamento pode ser feito acessando o seguinte link:

<http://sourceforge.net/projects/flavia/files/Leaf%20Image%20Dataset/1.0/Leaves.tar.bz2/download>

Ou copiado durante as próximas aulas (tragam um pen drive com pelo menos 1 GB livre).

- A divisão dos exemplos de treinamento e teste deve ser feita de forma aleatória. Normalmente 70% dos exemplos devem ser utilizados para treinamento e 30% para testes. Você deve garantir também que os exemplos de cada classe sejam distribuídos nestas mesmas proporções nos dois conjuntos.
- A implementação de um dos algoritmos pode ser feita em qualquer linguagem (C/C++, C#, Java...), mas aconselha-se utilizar a linguagem C/C++ usando o programa base para ler a imagem e realizar a classificação.
- Você deve decidir quais atributos serão utilizados pelos classificadores. A classificação pode piorar ou melhorar dependendo do conjunto de atributos utilizado. Os atributos extraídos pelo programa base são apenas exemplos de atributos que podem ser extraídos. Você deve escolher quais atributos serão utilizados e fazer testes para verificar qual conjunto de atributos apresenta melhores resultados.
- Para se conseguir melhores resultados será necessário extrair mais características das imagens de treinamento. O programa base extrai apenas algumas características estruturais, mas existem muitas outras informações úteis que podem ser extraídas computacionalmente das imagens.

### **Forma de Avaliação:**

Será avaliado se todas as etapas do processo foram cumpridas corretamente. A avaliação também será baseada na **apresentação dos resultados** durante a aula.

Essa apresentação deverá conter:

- Descrição da modelagem dos exemplos de treinamento:
  - Atributos selecionados para descrever os exemplos;
  - Justificativa para a escolha dos atributos;
  - Estrutura dos exemplos;
- Descrição dos experimentos realizados:
  - Variações na modelagem dos exemplos;
  - Variações no conjunto treinamento e testes;
  - Variação nos parâmetros dos algoritmos;
- Comparação dos algoritmos analisados:
  - Taxa de reconhecimento;
  - Tempo gasto no processo de treinamento;
  - Tempo gasto no processo de classificação de um exemplo desconhecido;
- Implementação:
  - Descrição de como o algoritmo escolhido foi implementado;
  - Resultados alcançados pelo algoritmo que foi implementado (tempo gasto no processo de treinamento, tempo gasto no processo de classificação de um exemplo desconhecido, taxa de reconhecimento).

### **Bônus:**

- O trabalho que conseguir uma taxa de reconhecimento acima de 91% receberá 2.0 pontos extras na nota.
- O trabalho que implementar uma Rede Neural ou SVM sem utilizar nenhuma biblioteca externa receberá 2.0 pontos extras na nota.

### **Data de Entrega:**

10/12

### **Forma de Entrega:**

Os trabalhos devem ser **apresentados** na aula do dia 10/12 (segunda) e 12/12 (quarta).

Todos devem enviar o trabalho até o dia 12/12 para o email [edirlei.slima@gmail.com](mailto:edirlei.slima@gmail.com).

**Não serão aceitos trabalhos enviados depois desta data.**

**Referências:**

[1] Stephen Gang Wu, Forrest Sheng Bao, Eric You Xu, Yu-Xuan Wang, Yi-Fan Chang and Qiao-Liang Xiang. **A Leaf Recognition Algorithm for Plant classification Using Probabilistic Neural Network.** In: IEEE 7th International Symposium on Signal Processing and Information Technology, 2007. <http://arxiv.org/pdf/0707.4289.pdf>