

Programação Automática de Máquinas Morfológicas por Aprendizado Computacional

Aluno: Caetano Jimenez Carezzato
Orientador: Junior Barrera

Departamento de Ciência da Computação
Instituto de Matemática e Estatística
Universidade de São Paulo

Introdução

A Morfologia Matemática é uma teoria útil para modelar transformações entre imagens e, por isso, vem sendo empregada com bastante sucesso para resolver diversos problemas de processamento de imagens. Entretanto, para se obter bons resultados em aplicações, o usuário deve possuir conhecimentos específicos a cerca da técnica empregada, bem como uma vasta experiência na área. Para simplificar a tarefa do usuário, vem sendo estudadas técnicas de projeto automático de operadores (i.e., transformações) morfológicas. Neste trabalho estudamos a aplicação da técnica de aprendizado Provavelmente Aproximadamente Correto para o projeto de operadores morfológicos (ou, equivalentemente, programas para Máquinas Morfológicas), que resolvem problemas de Análise de Imagens. Em particular, estudamos um método híbrido de projeto de operadores (heurístico-estatístico) que viabiliza a solução de problemas complexos e extensos para melhorar a sua precisão.

Atividades Realizadas

- No período de março de 1998 a agosto de 1999, realizamos diversas atividades. Primeiramente estudamos o básico do Processamento Morfológico de Imagens, sua representação, operações e operadores, elementos estruturantes, dilatações e erosões, manipulação geométrica, filtros morfológicos, resíduos, gradiente e alguns exemplos de aplicações.
 - Logo em seguida, passamos a estudar as bases necessárias para a Programação Automática de Máquinas Morfológicas:
 - **Elementos da Teoria dos Conjuntos** - Relação de Ordem em Conjuntos, Funções, Elementos da Teoria dos Reticulados, Princípio da Dualidade;
 - **Elementos da Teoria de Probabilidades** - Espaço de Probabilidades, Variáveis Aleatórias, noções de Funções Booleanas, Álgebra Booleana, Funções Booleanas, Representação cúbica de uma função Booleana, Minimização Tabular de QuineMcCluskey e Grupo Abeliano;
 - **Decomposição de Operadores** - Operadores Elementares, Decomposição Canônica, Decomposição Minimal, Operadores vistos através do Isomorfismo, Relações de Isomorfismo, Relações entre Operadores e Funções Booleanas, Decomposição de Operadores com Propriedades Especiais;
 - **Projeto de Operadores** - Conjuntos Aleatórios, Estacionariedade, Transformações Morfológicas de Conjuntos Aleatórios, Projeto de Filtros Morfológicos Ótimos;
 - **Modelos de Aprendizado Computacional** - Modelagem do Processo de Aprendizado, Modelo de Aprendizado PAC, Extensão do Modelo PAC;
 - **Algoritmos para o Aprendizado de Funções Booleanas** - O algoritmo ISI, Quine X ISI, Variantes do ISI;
 - **Geração Automática de Operadores Morfológicos** - Especificação do sistema, Decisão Ótima, Aspectos Práticos;
- Após essa primeira fase, realizamos a visualização de exemplos prontos, inicialmente instalamos e estudamos a interface de desenvolvimento (Khoros 2.2). Estudamos as Toolboxes já implementadas necessárias para fazer uma pré-filtragem e o Aprendizado Computacional propriamente dito. Estudamos exemplos já prontos, como o reconhecimento de bordas, letras impressas, textos, eliminação de ruído etc.

Terminada a fase inicial de leituras e estudo de exemplos, resolvemos implementar um reconhecedor de dígitos escritos à mão. Começamos os experimentos tentando reconhecer o dígito zero. Para tal, utilizamos uma série de formulários já disponíveis (100 no total) cada uma com 672 dígitos do mesmo tipo, escrito por diversas pessoas.

Para o nosso aprendizado, escolhemos um subconjunto desses formulários e aplicamos um filtro morfológico. A ideia básica é normalizar as letras das pessoas e reduzir o contexto estatístico. Após o pré-processamento, realizamos o aprendizado com diversos tamanhos de janela e subconjuntos de formulários.

Os primeiros resultados obtidos estão na tabela abaixo:

Fólias com o dígito	Erro	Erro Percentual
0	264	39,28%
1	1	0,14%
2	44	6,54%
3	38	5,65%
4	35	5,20%
5	38	5,65%
6	57	8,48%
7	29	4,31%
8	73	10,86%
9	46	6,84%

Erro para menos (Não reconheceu o que devia reconhecer): 39,28%
Erro para mais (Reconheceu o que não devia): 5,95%
Erro total: 9,29%

Apesar do erro total ser menor do que 10%, o que não é nada mau para uma primeira tentativa, o erro para menos é quase 40%, o que indica fortes dificuldades para melhorar a taxa de acerto. Partimos então para um momento mais avançado. Continuamos as buscas por novos métodos para obter operadores melhores. A estratégia básica utilizada foi estender métodos já existentes. Para isso estudamos uma série de novos métodos que buscam ampliar a abrangência dos métodos já estudados anteriormente. Além disso, começamos um novo estudo que aplicará as técnicas já aprendidas: o projeto CAGE.

Projeto REAL

Na tentativa de melhorar a precisão dos operadores encontrados pelo método PAC, resolvemos criar uma extensão denominada REAL.

O REAL consiste basicamente em calcular a entropia do sistema de bases dos implicantes primos que descrevem o operador obtido e realizar transformações nesse bases a fim de obter um operador mais consistente em relação a determinados parâmetros de entropia. Sendo assim, valoriza-se o agrupamento de determinadas bases que eventualmente podem ser importantes para a identificação de classes de elementos usada por exemplo num OCR.

Participi de diversas reuniões com os demais participantes do projeto. Além disso realizei uma série de implementações do código e sugeri algumas otimizações no código que reduziram a complexidade de algumas partes do algoritmo tornando-o de 20 a 50 vezes mais rápido. Realizei também diversos testes com o novo método.

Novos experimentos e implementações

Após algum tempo estudando o REAL, resolvemos fazer alguns experimentos para testar a sua viabilidade. Notamos uma pequena melhoria na precisão dos operadores do nosso OCR de dígitos manuscritos, mas, achando ainda não ser suficiente bom para termos um OCR de verdade, resolvemos experimentar novas técnicas.

Resolvemos utilizar um novo método chamado pirâmide que busca minimizar a quantidade de erros devido a dados não observados. A ideia toda do método é procurar preencher toda a tabela de don't cares obtida pelo método PAC com algum valor mais significativo usando janelas de diferentes tamanhos.

Tendo estudado esses dois métodos, realizamos uma série de testes com eles para tentar melhorar a precisão do nosso OCR. Ela melhorou um pouco principalmente diminuindo o erro principal que consiste em não reconhecer algumas letras que deveriam ser reconhecidas.

Para os testes, criamos um novo ambiente automatizado que permite uma mais fácil obtenção e visualização dos dados.

Implementei um pequeno programa de escalonamento de imagens para o Khoros que viabiliza a ideia de outro método, o multi-scale, que também busca preencher a tabela de don't cares com valores mais significativos usando escalonamentos nas imagens de treinamento. Quando esse método for implementado por completo poderemos iniciar uma nova bateria de testes que utilizam esses três métodos juntos e mais um, o multi-resolução ainda em estudo.

Leituras

- Durante praticamente todo o tempo de vigência da bolsa, realizei uma série de leituras destinadas ao aprendizado de novos métodos ou preparo para o projeto CAGE. As principais leituras consistiram de:
- Manual do Khoros - Essa leitura foi essencial para que eu pudesse ajudar a instalar o khoros no nosso servidor e participar de diversas implementações como a do projeto REAL ou do multi-scale.
- Releitura da tese de mestrado da Nina - Para uma melhor fixação dos conceitos aprendidos no período anterior.
- Artigo sobre o Aperture Filter - Acreditamos que esse será muito útil no futuro para o nosso OCR e para o projeto CAGE.
- Livro sobre DNA e Proteínas - Utilizado no projeto CAGE.
- Diversos artigos sobre micro-array e aplicativos usados para a seqüencição do DNA bem como seu armazenamento e busca - Essencial para participar do projeto CAGE.

Projeto CAGE

Durante a realização dos experimentos supracitados, surgiu um novo projeto onde muitas das técnicas utilizadas para resolver o já conhecido problema do OCR podem ser reutilizadas no projeto. O projeto consiste basicamente em, em conjunto com a biofísica, construir um sistema de análise de micro-arrays.

Para participar desse novo projeto, realizei uma série de leituras já especificadas e assisti uma série de palestras que visavam introduzir alguns conceitos de biofísica e micro-array. Foram eles:

- 2 seminários introdutórios realizados no IME. Falou-se praticamente de um pouco da linguagem de biofísica, uma visão geral de coisas que tornam essa área uma das mais interessantes atualmente e algumas coisas mais dirigidas à nossa proposta de atuação, enfatizando aspectos de seqüenciamento e microarrays.
- Seminário sobre o Dystrostellum, um organismo que provavelmente dará origem aos primeiros microarrays que poderemos analisar.
- Seminário sobre ciclo celular, fenomenologia e modelagem.
- 2 seminários sobre princípios básicos de Biologia Molecular do Gme. O objetivo foi tornar acessíveis ao pessoal do IME conceitos e métodos importantes no entendimento da análise da expressão gênica através da metodologia de microarrays de DNA. Os tópicos abordados foram: Organização e análise dos genes e Níveis de controle da expressão dos genes.
- Seminário sobre o CAGE falando de "Análise da expressão gênica no Câncer: como o CAGE se relaciona com o projeto FAPESP/LICR de seqüenciamento do genoma humano do câncer".
- Seminário sobre microarrays num nível básico.
- Seminário de Biologia Molecular Computacional - Reconstrução Filogenética: enfoques e perspectivas.

Conclusão

Após realizarmos uma série de experimentos, concluímos que o melhor a fazer seria uma série de leituras sobre novos métodos. A maior parte do tempo foi dedicada a essas leituras.

Paramos um pouco com implementações e experimentos visando uma nova fase de iniciação científica constituída por um aprendizado mais aprofundado numa série de novas técnicas que surgiram e outras que estão sendo desenvolvidas pelos meus colegas.

Além disso, encontramos um novo horizonte representado pelo projeto CAGE que força uma nova abordagem dos métodos já aprendidos e de outros ainda não estudados o que pode vir a ser uma ótima aplicação para o estudo já realizado.

O OCR, objeto de estudo dessa bolsa e da anterior, teve uma ligeira melhora com os novos métodos mas ainda não o achamos suficientemente bom. Paramos um pouco de desenvolvê-lo para estudar técnicas que possam torná-lo viável.

Portanto, estamos agora numa nova fase que consiste basicamente no estudo de novos métodos e problemas de aplicações para os métodos já aprendidos.

Atualmente estou me dedicando mais ao projeto CAGE, na área de clustering.

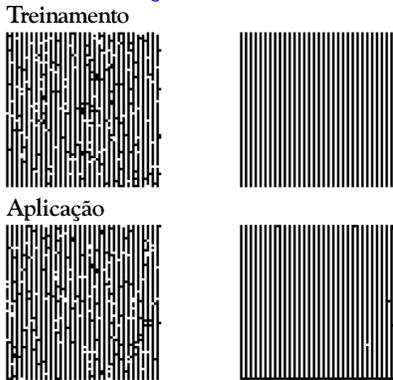
Além disso, temos diversos exemplos de aplicação e resultados do OCR desenvolvidos.

Contato

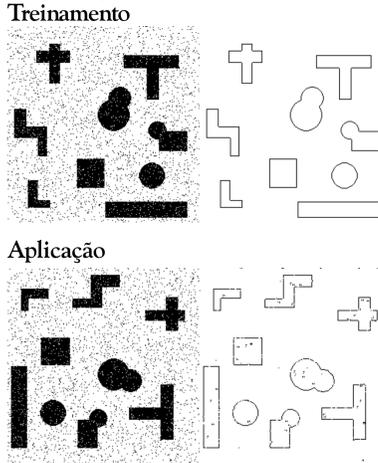
Maiores informações e referência bibliográficas:
cetaz@vision.ime.usp.br
<http://www.linux.ime.usp.br/~caetano/ic/relatorios/>

Exemplos de Aplicação

Redução de Ruídos



Extração de Ruídos e Bordas



Redução de Ruídos

Treinamento

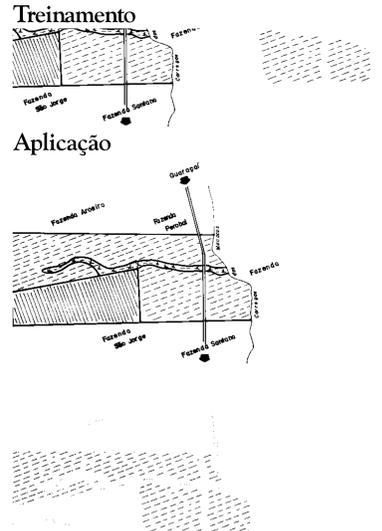
ABCDEFGHIJ
KLMNOPQRS
TUVXZ WY
a b c d e f g h i j k l m
n o p q r s t u v x z w y

Aplicação

CHARLES BABBAGE Everyone from bankers to navigators depended on mathematical tables during the Industrial Revolution. However, these hand-calculated tables were usually full of errors. After discovering that his own tables were riddled with mistakes, Charles Babbage envisioned a steam powered differential engine and then an analytical engine that would perform tedious calculations accurately.

CHARLES BABBAGE Everyone from bankers to navigators depended on mathematical tables during the Industrial Revolution. However, these hand-calculated tables were usually full of errors. After discovering that his own tables were riddled with mistakes, Charles Babbage envisioned a steam powered differential engine and then an analytical engine that would perform tedious calculations accurately.

Reconhecimento de Texturas



Resultado do Experimento

