

MAC499 – Trabalho de Formatura Supervisionado

Combinação de Classificadores

Alunos: Adolpho Pacheco, Camila Pacheco

Orientadora: Nina S. T. Hirata

Introdução

Operadores morfológicos invariantes à translação, quando restritos ao contexto de imagens binárias, podem ser vistos como mapeamentos caracterizados por funções booleanas. As funções booleanas por sua vez podem ser vistas como classificadores de padrões binários. A identificação de padrões em imagens vem sendo explorada por diversas áreas para muitas aplicações. Uma técnica bastante comum é a utilização de algoritmos de aprendizado que, a partir de exemplos de treinamento, são capazes de inferir classificadores que são utilizados para identificar alguns padrões. Em geral, vários classificadores são projetados e aquele com melhor desempenho empírico é escolhido. Este trabalho visa experimentar técnicas de combinação de classificadores no projeto de operadores morfológicos possibilitando melhor capacidade de generalização e resultados mais estáveis que os classificadores individuais, aproveitando as informações fornecidas por todos os classificadores individuais.

Projeto de operadores baseado em aprendizado

De acordo com [1], o projeto de operadores pode ser visto como um processo consistindo das seguintes etapas:

Coleta:

São dados como entrada pares de imagens representando respectivamente as imagens anterior e posterior à transformação desejada. A coleta é feita por meio de uma janela que percorre a imagem anterior, e associa o padrão dos pontos (configuração) dessa janela ao valor do pixel na imagem ideal, correspondente ao centro da janela. A partir dessa coleta é gerada uma tabela de padrões e suas frequências (números de 0's e 1's observados na imagem ideal).

Decisão:

A partir dessa tabela, é escolhido, para cada padrão observado, o valor (0 ou 1) que possui a maior frequência.

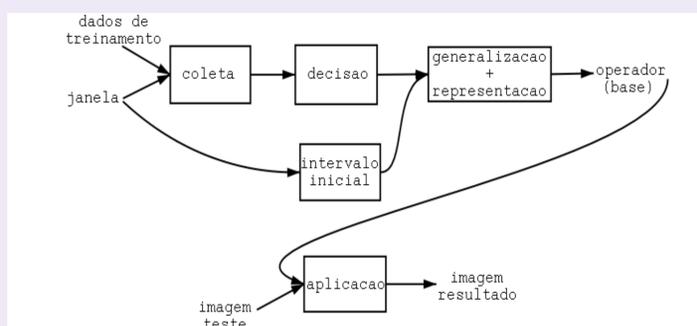
Generalização:

Nem todos os possíveis padrões são observados na coleta de dados. Assim, algoritmos de aprendizado podem ser utilizados para associar valores (0 ou 1) a estes padrões. Um algoritmo tem capacidade de boa generalização se é capaz de associar a classificação correta a esses padrões.

Validação:

Aplicar o operador projetado sobre uma imagem de teste e analisar o desempenho.

Estrutura do processo:



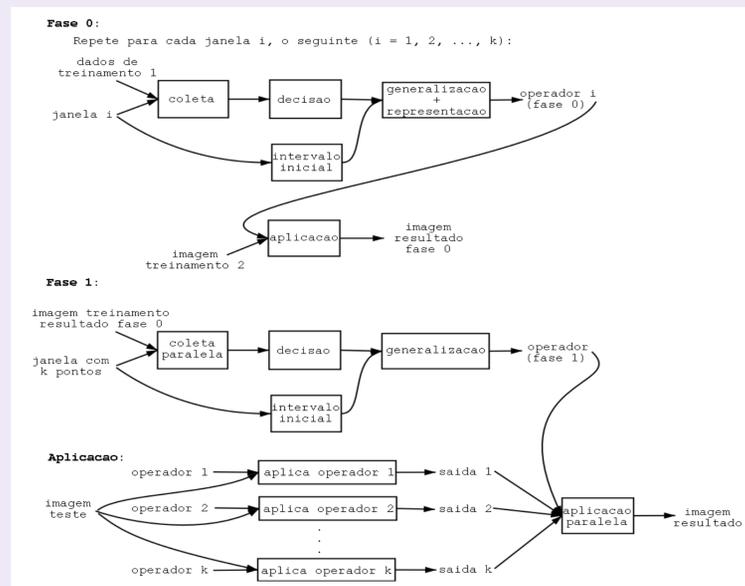
Combinando classificadores

Se a janela considerada é muito grande, o operador é pouco preciso, pois muitos padrões não são observados na imagem. Além disso, o tempo gasto no treinamento é muito grande (pode-se levar vários dias). Não podemos também diminuir muito o tamanho da janela, pois perdemos a capacidade de discriminar formas relativamente grandes.

Uma possível abordagem para contornar esse problema consiste em considerar várias subjanelas da janela original, projetar um operador para cada subjanela e combinar o resultado desses operadores para obter o resultado final.

Para isso, podemos usar o método *stacked generalization* [2]. *Stacked generalization* refere-se a qualquer técnica de aprendizado que utiliza mais de um nível de treinamento. O nível 0 utiliza exclusivamente os dados de entrada, enquanto os níveis superiores podem utilizar resultados de níveis abaixo deles.

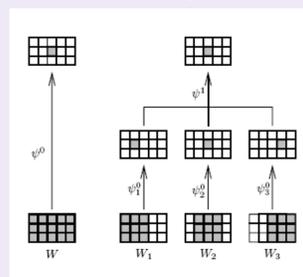
Projeto de operadores baseado em *stacked generalization*



Este trabalho restringiu-se ao uso de *stacked generalization* de 2 fases.

Na figura acima, k subjanelas são utilizadas na fase 0. Para cada subjanela, os operadores da fase 0 são gerados conforme procedimento descrito anteriormente. Na fase 1, a coleta considera as imagens resultantes de cada um dos k operadores da fase 0. As etapas de decisão e generalização são similares à fase 0.

Cada classificador da fase 0 pode ter como alvo um mesmo pixel bem como diferentes pixels na imagem de saída. A figura ao lado mostra a abordagem mais genérica de treinamento em duas fases em comparação com o treinamento em uma única fase.



A coleta da fase 1 é chamada paralela pois cada padrão é obtido a partir das k imagens resultantes dos operadores da fase 0 (compondo-se os valores dos pixels alvos de cada operador).

Tendo os operadores da fase 0 e o operador da fase 1, para fazer a aplicação, deve-se primeiro aplicar na imagem cada um dos k operadores da fase 0, gerando k imagens. Então, aplica-se o classificador da fase 1 nos padrões obtidos de forma similar à coleta paralela.

A implementação dessa técnica foi feita em linguagem C, no ambiente Khoros.

Alguns resultados

Janela considerada para uma fase: 3×7 .

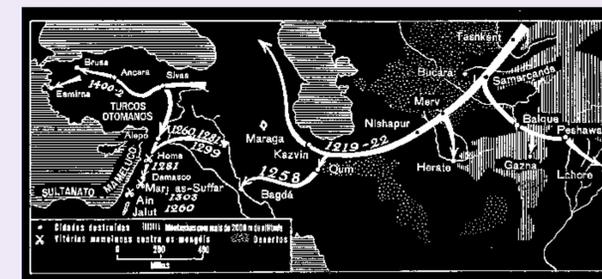
Janela considerada para duas fases: Tres subjanelas 3×5 .

Os testes mostram que a implementação do algoritmo em duas fases tem um tempo de processamento muito menor que o de uma fase, pois o tempo de processamento aumenta exponencialmente conforme aumenta o tamanho da janela.

O operador que apresentou melhores resultados foi a de duas fases com todos os classificadores tendo como alvo um mesmo pixel bem na imagem de saída, como podemos observar na tabela a seguir:

	Erro*	Tempo
1 fase	2,75%	33 horas
2 fases (mesmo pixel)	2,65%	2 minutos
2 fases (pixels diferentes)	2,78%	2 minutos

*O erro é calculado comparando-se a imagem resultante com a imagem ideal e contando quantos pixels são diferentes.



(imagem teste)



(Operador de uma fase com janela 3×7)



(Operador de duas fases, com mesmo pixel alvo)



(Operador de duas fases, com pixels alvos distintos)

Referências

- [1] Nina S. Tomita, Programação automática de máquinas morfológicas binárias baseada em aprendizado PAC, Dissertação de mestrado, IME- USP, 1996.
- [2] D. H. Wolpert, Stacked Generalization, Neural Networks, vol. 5, pp.241-259, 1992.