

# Aplicação de Técnicas Adaptativas em Reconhecimento de Formas

Leoncio C. Barros Neto, André R. Hirakawa e Antonio M. A. Massola

**Resumo**—Este artigo apresenta a proposta do desenvolvimento de um método para classificação de formas geométricas básicas por meio de Autômatos Adaptativos (AA). O método proposto utiliza a abordagem sintática de reconhecimento de padrões, apresentando algumas vantagens sobre práticas tradicionais permitindo realizar uma identificação das propriedades estruturais das formas contidas na imagem. A característica adaptativa aperfeiçoa o processo de reconhecimento ao considerar pequenas diferenças entre bordas, geradas por distorções residuais nas fases de segmentação e pré-processamento. O artigo apresenta ainda os resultados dos testes de reconhecimento de formas utilizando AA e implementado experimentalmente.

**Palavras Chave**— Reconhecimento de Padrões, Autômatos Adaptativos, Experimentação, Modelagem.

## II. INTRODUÇÃO

No sentido de aumentar a capacidade de percepção do ambiente, elevando o desempenho dos sistemas modernos, sensores mais sofisticados tem sido integrados. Dentre esses estão aqueles que empregam imagens, por exemplo, aplicados em robótica para reconhecimento de formas.

No processo de reconhecimento e classificação de padrões, a primeira etapa é a do processamento de imagens e a última é a da classificação de formas, passando pelo processo de extração de características e pré-processamento. Nesse processo, é inegável que a qualidade da imagem e atributos extraídos [1] interferem no resultado. Por outro lado, a classificação continua sendo considerada um desafio para os pesquisadores, podendo comprometer o processo.

A abordagem sintática é considerada fundamental para o reconhecimento de formas devido a sua característica de incorporar informações estruturais [2]. Entretanto, conforme descrito no item a seguir, as técnicas sintáticas tradicionais apresentam restrições tal como a dificuldade em tratar distorções não solucionadas pelas fases preliminares que alimentam dados para a etapa de classificação.

---

Manuscrito recebido em 11 de Dezembro de 2007. Trabalho desenvolvido no Laboratório de Automação Agrícola (LAA), da Escola Politécnica da USP, São Paulo, Brasil.

L. C. Barros Neto; LAA (número telefônico: 55-11-36824400; e-mail: leoncio.barros@poli.usp.br).

A. R. Hirakawa; LAA (número telefônico: 55-11-30915104; e-mail: andre.hirakawa@poli.usp.br).

A. M. A. Massola; LAA (número telefônico: 55-11-30914289; e-mail: antonio.massola@poli.usp.br).

## III. DEFINIÇÃO DO PROBLEMA

### A. Reconhecimento de Formas

O Reconhecimento de formas está inserido no contexto do reconhecimento de padrões (RP). RP pode ser definido como um processo de identificação de estruturas ou padrões nos dados ou medidas, por meio de comparações com estruturas conhecidas [3]. Trata-se de uma ciência complexa, pois a maioria das informações oriundas do mundo real está sujeita a transformações, deformações ou corrompida por ruído.

A Figura 1 apresenta as etapas típicas do processo de RP, envolvendo sensor, pré-processamento, extração de atributos e classificação.

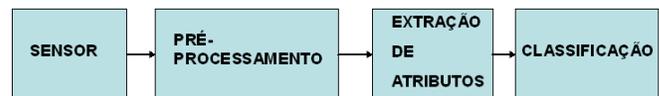


Fig. 1. Estrutura típica de reconhecimento de padrões.

Na etapa inicial do sensoriamento, dados analógicos do mundo físico são coletados por transdutores ou sensores e convertidos em conjuntos de dados digitais. Na fase de pré-processamento, esses dados são tratados convenientemente e encaminhados para a etapa de extração de atributos. A etapa de extração de atributos agrupa os dados recebidos em conjuntos de vetores ou primitivas. No caso de imagens, de modo geral, a extração de atributos agrupa os *pixels* em regiões com características semelhantes num processo denominado segmentação.

Após a segmentação da imagem, a classificação é realizada atuando nesses conjuntos fornecidos pela fase anterior. O algoritmo de classificação compara os atributos encontrados na segmentação dentro de certos limiares de aceitação pré-determinados identificando estruturas similares. As classes resultantes são então associadas às classes de interesse.

Caracterização detalhada dessas fases pode ser encontrada em trabalhos sobre processamento de imagens [4].

Este artigo se concentra na etapa final de classificação utilizando a abordagem sintática, comentada a seguir.

### B. Abordagem Sintática

Na abordagem sintática, detalhes locais da imagem são descritos por intermédio de padrões elementares ou primitivas e os sinais de entrada para classificação através da cadeia concatenada dessas primitivas (*string*) [3]–[6]. A seleção de primitivas é especificada de acordo com o problema e visa facilitar a descrição estrutural dos padrões. A Figura 2

apresenta as primitivas utilizadas neste artigo, convenientes para reconhecimento de quadrados e retângulos. Para simplificar, as dimensões das primitivas são consideradas neste trabalho como uma unidade da dimensão unitária convencionada para a imagem. Portanto, as formas segmentadas são descritas por meio de *strings* relacionados aos contornos das regiões associadas.

Metodologias habituais de reconhecimento dessas seqüências de símbolos aplicam autômatos finitos representando uma gramática formal em que as primitivas são símbolos terminais do alfabeto da gramática. Um aspecto interessante é que poucas primitivas podem representar muitas classes, entretanto, o tópico abaixo enfatiza o requisito de uma máquina de Turing para o reconhecimento dos *strings* correspondentes.

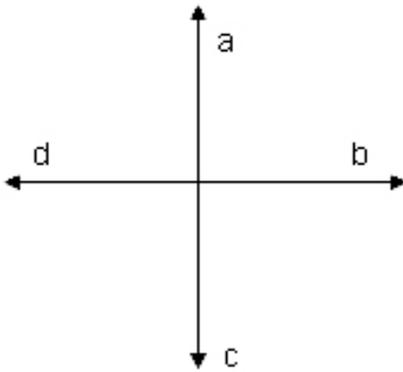


Fig. 2. Exemplo de primitivas.

### C. Stringology e Autômatos Adaptativos (AA)

*Stringology* é o apelido de uma área relativamente nova em Ciência da Computação [5] e engloba a classe de problemas envolvendo cadeias de símbolos, ou passíveis de terem solução por essas cadeias que é o caso do reconhecimento em questão.

Ocorre que as formas a serem reconhecidas envolvem *strings* pertencentes a linguagens dependentes de contexto, requerendo uma máquina de Turing. Em certos casos particulares, um autômato finito pode efetuar o reconhecimento, limitado a formas com dimensões fixadas.

A equivalência dos AA com a máquina de Turing [7] possibilitou o reconhecimento de triângulos equiláteros [8]. A questão seguinte é como diferenciar entre formas com estruturas similares, independente das dimensões, tal como um quadrado de um retângulo. O problema do reconhecimento do *string* ganha em dificuldade ao tratar o caso de *strings* gerados nos casos práticos, ou seja, como padrões não ideais das formas e sujeitos a vários tipos de distorções que ocorrem devido aos erros residuais dos estágios preliminares de segmentação e pré-processamento.

### D. Ruído e Distorção

Ruído e distorção causados pelo posicionamento relativo da câmera ou sensor são questões em qualquer sistema envolvendo imagens. Existe literatura técnica para a solução de cada tipo de distúrbio e para a redução de erros, minimizando a complexidade da etapa final de reconhecimento [9].

No caso de padrões rotacionados, Costa [8] comentou não existir possibilidade de classificá-los, pois as primitivas estarão deslocadas de um ângulo determinado. Uma solução relativamente simples desse efeito implica na rotação de todas as formas por um eixo comum da imagem antes de aplicar as primitivas. Com isto, todas as formas terão um eixo comum. O erro de paralaxe pode ser minimizado pela rotação da imagem em relação a algum eixo perpendicular ao plano da lente da câmera, preliminarmente à aplicação das primitivas.

Há situações caracterizadas por freqüentes alterações dos parâmetros indicativos da orientação dos sensores, tal como descrito em [10]. Nesse caso, soluções encontradas buscam integrar modelos já existentes formando módulos, levando em conta as necessidades dos usuários, dados reais disponíveis, particularidades dos sensores e os requisitos de precisão e tolerância a fim de obter uma solução ótima.

De modo geral, os algoritmos de classificação têm que reconhecer *strings* representando padrões não ideais, ou seja, formas com pequenas diferenças das formas corretas. Requisitos como esses geram as principais restrições para uso dos procedimentos sintáticos tradicionais em classificação.

## IV. O MÉTODO

Foram identificadas três estruturas computacionais principais tendendo a se repetir em qualquer reconhecedor, denominadas pilha (*stack*), laço (*loop*) e ponteiro (*pointer*), que facilitam a modelagem do autômato. Considerando-se uma única forma, um modelo básico contempla as seguintes etapas:

- 1) De acordo com a forma, elaborar o modelo topológico do reconhecedor introduzindo as estruturas computacionais (laços, pilhas e ponteiros) conforme necessário.
- 2) Projetar a máquina não-adaptativa.
- 3) Dividir o projeto em unidades de acordo com as estruturas computacionais e projetar as funções adaptativas para acesso às estruturas.
- 4) Integrar as funções adaptativas a uma ou mais cópias da máquina não-adaptativa para atuação com cada estrutura.
- 5) Teste das unidades.
- 6) Integração geral das unidades e teste global.
- 7) Documentar o processo.

Sendo possível o projeto de um autômato para cada forma, torna-se viável a criação de banco de autômatos para reconhecer várias modalidades de formas.

As funções adaptativas que manipulam pilhas costumam ser simples e normalmente de duas espécies. Uma espécie é de incremento da pilha em que a função gera uma nova transição e um novo estado que passam a ser os últimos da seqüência. A segunda espécie é de redução da pilha, de modo que a última transição é removida e o penúltimo estado passa a ser o último. Quando as pilhas são construídas entre dois estados fixos, as funções adaptativas correspondentes são similares às descritas, com poucas variações.

Para reconhecer formas quadradas, sem considerar distorções, à medida que o autômato consome a seqüência dos *tokens* relativos ao primeiro lado, o AA deve gerar três grupos

de transições unidas, agrupadas em três pilhas, apresentando número igual de transições ao *token* lido (no caso *token a*). A forma estará correta se o autômato consumir essas três pilhas com sucesso, quando do recebimento dos símbolos dos demais lados.

A Figura 3 apresenta exemplos de pilhas construídas pelo autômato durante seu processo adaptativo. A primeira pilha é formada entre os estados 101 e 102 e duas pilhas interligadas serão construídas entre os estados 301 e 302 e entre os estados 401 e 402.

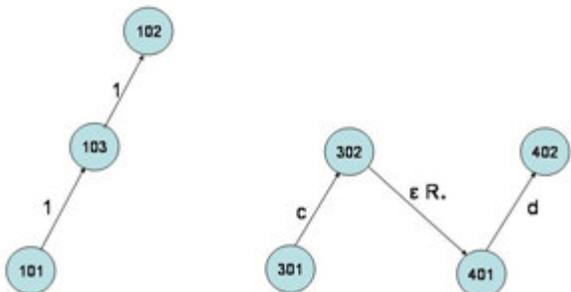


Fig. 3. Três pilhas construídas pelo AA.

Os laços são entendidos como um caso particular de pilha formando grupo de transições unidas que começam e terminam no mesmo estado, convenientes para armazenamento de dados ou instruções a serem acessadas ciclicamente. A Figura 4 mostra um laço em torno do estado 302. Os ponteiros são comumente usados pelo AA para acessar dados ou instruções existente nos laços e pilhas. A Figura 5 mostra exemplos de ponteiros.

De acordo com as formas a serem reconhecidas, pode-se iniciar o projeto do classificador a partir de uma descrição topológica do autômato, possibilitando estimativas de memória utilizada e dimensões do autômato. Esse procedimento é análogo com o seguido pela engenharia de *software* em que cada *design pattern* [11] apresenta uma solução para um determinado tipo de problema recorrente.

## V. IMPLEMENTAÇÃO EFETUADA

A título de exemplo, foi implementado um reconhecedor de quadrados e retângulos em que a característica adaptativa constrói dinamicamente uma função linear (concretizada através de laços) relacionando a quantidade de estados auxiliares gerados em pilhas às tolerâncias das dimensões das formas. Conhecimentos estruturais das formas estão implícitos nas funções adaptativas que constroem as pilhas.

Não considerando distorções, qualquer forma associada a um *string* do tipo  $a^n b^m c^n d^m$  deve ser classificado como quadrado. O classificador tem que diferenciar essa linguagem dos quadrados da linguagem  $a^n b^m c^n d^m$   $m \neq n$ , referente aos retângulos.

Entretanto, levando em conta a existência de distorções, as suposições abaixo são pertinentes:

1) A situação prática operacional implica em formas apresentando lados maiores que um determinado limiar  $k$

(número inteiro) de unidades de primitivas, correspondendo à região de distorção e ruído das etapas de pré-processamento e de segmentação.

2) O processo de reconhecimento iniciando pelo menor lado da forma.

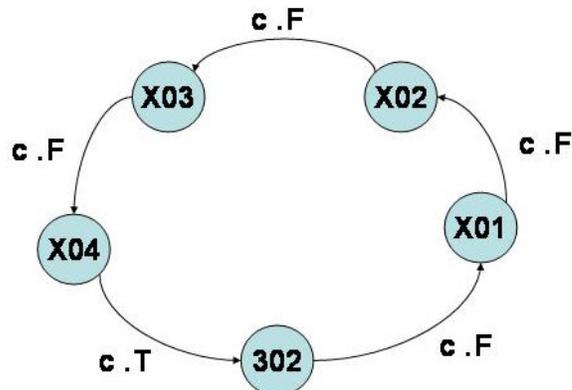


Fig. 4. Laço que inicia e termina no estado 302.

Assim, qualquer forma representada por linguagem do tipo  $a^n b^m c^l d^u$  será aceita desde que satisfeitas as condições  $n - k \leq l \leq n + k$  e  $m - k \leq u \leq m + k$ . Adicionalmente, a forma será classificada como quadrado quando atendida a condição seguinte  $n - k \leq l, m, u \leq n + k$ . Tendo o *string* atendido todas as condições, trata-se de um quadrado. Se apenas as duas primeiras condições forem satisfeitas, a forma é retangular. Este é apenas um exemplo em que retângulos e quadrados são identificados em um único processo adaptativo, passível de variações nos seus detalhes de acordo com as especificações.

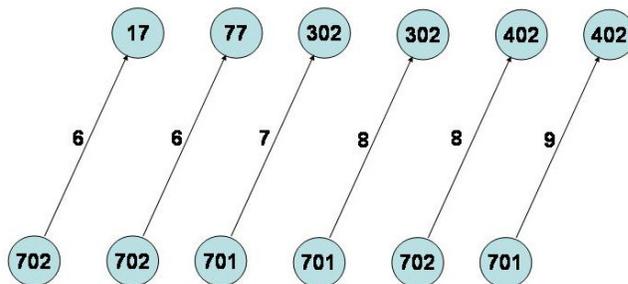


Fig. 5. Exemplo de seis ponteiros.

## VI. RESUMO DOS TESTES

A fim de validar o classificador, a tabela I indica alguns *strings* de teste, associados às respostas do autômato.

De acordo com a metodologia do experimento, os referidos *strings* correspondem a formas distorcidas, simulando os estágios de RP anteriores ao classificador. O teste procura varrer a faixa de aceitação do classificador variando as dimensões dos lados das formas conforme o expoente de cada símbolo dos *strings*. A coluna Resposta do AA mostra que o autômato reconhece os padrões adequadamente mesmo havendo aumento ou diminuição de qualquer lado da forma, até uma dimensão limite, a partir da qual as seqüências passam

a ser rejeitadas. Havendo aceitação do *string*, o estado final 7 indica o retângulo e o estado final 10, o quadrado. A rejeição de qualquer *string* corresponde aos casos em que a forma é desconhecida.

TABELA I

STRING DE ENTRADA E CORRESPONDENTE ESTADO FINAL E RESPOSTA DO AA

| STRING DE ENTRADA          | ESTADO FINAL | RESPOSTA DO AA |
|----------------------------|--------------|----------------|
| $a^{10}b^{15}c^{13}d^{16}$ | 7            | ACEITO         |
| $a^{10}b^{15}c^{13}d^{18}$ | 7            | ACEITO         |
| $a^{10}b^{15}c^{13}d^{20}$ |              | REJEITADO      |
| $a^{10}b^{12}c^{12}d^{14}$ | 10           | ACEITO         |
| $a^{10}b^{12}c^{10}d^{14}$ | 10           | ACEITO         |
| $a^{10}b^{12}c^9d^{14}$    | 10           | ACEITO         |
| $a^{10}b^{12}c^8d^{14}$    | 10           | ACEITO         |
| $a^{10}b^{12}c^7d^{14}$    |              | REJEITADO      |
| $a^{10}b^{12}c^9d^9$       |              | REJEITADO      |
| $a^{10}b^{12}c^9d^{10}$    | 10           | ACEITO         |

Sob o enfoque de ferramental de *software*, destaque-se a simplicidade para o operador dentre as principais funcionalidades desta implementação. Basta o usuário introduzir os quatro expoentes do *string* via teclado do microcomputador para obter na tela as respostas aceito ou rejeitado e estado final. Quanto a aspectos de simulação de condições variadas de ruído bem como de interface com o sistema de RP, o autômato deve incorporar o respectivo limiar de distorção e ruído, especificado em número de primitivas. Basicamente, tal atividade é efetuada alterando-se a quantidade de estados em estrutura *loop* da máquina não-adaptativa, mantendo-se as funções adaptativas protegidas de acesso e aptas a atuarem em novas condições conforme as máquinas não-adaptativas.

## VII. CONCLUSÃO

Através dos resultados dos experimentos, constatou-se que o método proposto pode ser utilizado como referência para outros classificadores a serem desenvolvidos, auxiliando também na modelagem de problemas recorrentes em reconhecimento de formas.

Apesar de haver uma alteração de paradigma com os AAs, as funções adaptativas se mostraram sem grandes dificuldades, sendo a simplicidade e facilidade de implementação duas das principais vantagens constatadas com a proposta.

Os AAs preservam a característica positiva da abordagem sintática de favorecer o tratamento de padrões estruturais além de reconhecer as formas em qualquer escala. O uso de técnicas costumeiras exigiriam autômatos finitos grandes amarrados a uma única dimensão da forma.

Devido a ser um assunto relativamente novo, não se tem encontrado trabalhos em reconhecimento de padrões utilizando AA, em particular envolvendo *strings* sujeitos a ruído. Extensão do trabalho será voltada a potenciais aplicações do método e respectivas técnicas adaptativas.

## REFERÊNCIAS

- [1] Rodrigo de O. Plotze, Odemir M. Bruno. Análise de Formas e Reconhecimento de Padrões por Meio da Assinatura Fractal Multiescala. Journal of Computer Science, University of Lavras (Brazil), VOL.6, N.2, June, 2007.
- [2] Wu, W-Y; Two-dimensional Object Recognition through String Matching. Image Science Journal vol. 49, September 2001.
- [3] Fu, K. S. Syntactic Methods in Pattern Recognition. New York: Academic Press, 1974.
- [4] Ritter, G. X.; Wilson, J. N. Handbook of computer vision and algorithms in image algebra. 2nd. ed. 2000.
- [5] A. Apostólico; Z. Galil. Pattern Matching Algorithms. Oxford University Press, New York, 1997.
- [6] Schalkoff, R. J. Pattern Recognition: statistical, structural and neural approaches. John Wiley & Sons, Inc., 1992.
- [7] Rocha, R. L. de Azevedo; Neto, J. J. Autômato adaptativo, limites e complexidade em comparação com máquina de Turing. Proceedings of the second Congress of Logic Applied to Technology - LAPTEC'2000, São Paulo, Brasil, 2001.
- [8] Costa, E. R.; Hirakawa, A. R.; Neto, J. J. An Adaptive Alternative for Syntactic Pattern Recognition. Proceeding of 3rd International Symposium on Robotics and Automation, Toluca, Mexico, 2002.
- [9] Javidi, Bahram; Okano, Fumio. Three-Dimensional Television, Video, and Display Technologies, Springer-Verlag, 2002.
- [10] Michael Breuer, Jörg. Geometric Correction of Airborne Whiskbroom Scanner Imagery Using Hybrid Auxiliary Data, 19th ISPRS congress, Amsterdam, 2000.
- [11] Gamma, E., Helm, R., Johson, R., Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional, 1995.

**Leoncio C. Barros Neto.** Nascido em Campo Grande, MS, em 14 de setembro de 1954. Recebeu os títulos de Engenheiro Eletricista pela Escola de Engenharia Mauá, de São Caetano do Sul, SP, em 1979. Em 1994 recebeu o título de Mestre em Engenharia Elétrica da Universidade de São Paulo. Em 1981, após atuação em Telecomunicações e Sistemas de Controle de Processos em empresas como Siemens e Duratex, foi aprovado em concurso público de âmbito nacional, sendo integrado ao Corpo de Engenheiros Navais da Marinha do Brasil (MB). Na MB, atingiu o mais alto posto de Oficial Superior, exercendo atividades técnicas em Setores de Manutenção e Pesquisa. Especializou-se através de cursos na Alemanha e Inglaterra sobre sistemas eletrônicos aplicados em segurança e defesa.

Após transferência para a reserva, iniciou programa de doutorado no Departamento de Engenharia de Computação e Sistemas Digitais da USP em 2006, sob orientação do Prof. Dr. Massola, passando também a exercer atividades profissionais de consultoria. Suas áreas de interesse e pesquisa abrangem Automação, sensores inteligentes, eletrônica de potência, sistemas embarcados, Sistemas de Detecção, Guerra Eletrônica e suas aplicações.

**André R. Hirakawa.** Nascido em São Paulo, Brasil, em 15 de junho de 1965. Recebeu os títulos de Engenheiro Eletricista e correspondente Mestrado da Universidade de São Paulo (USP), em 1990 e 1992 respectivamente. Em 1992, como pesquisador, foi integrado no programa de doutorado da Yokohama National University, em Yokohama, Japan, tendo recebido o título de PhD em 1997.

Em 1998, iniciou atividades no Departamento de Engenharia de Computação e Sistemas Digitais da USP, sendo atualmente Professor Assistente. Suas áreas de interesse e pesquisa abrangem Automação e Robótica, sensores inteligentes, eletrônica de potência, planejamento de rotas, AVG's, e sistemas sem fio, tanto aplicados à automação agrícola quanto em outros ambientes.

**Antonio M. A. Massola.** Professor Titular da USP e orientador do trabalho de doutorado do primeiro autor.