

Integração de Crescimento de Região e Detecção de Arestas através de Operadores de Agregação

DENISE GULIATO^{1,3}, WALTER A. CARNIELLI², JOÃO ANTÔNIO ZUFFO³

¹Depto de Informática, Universidade Federal de Uberlândia, Campus Santa Mônica
Av. João Naves de Ávila, 2160, 38400-089 Uberlândia, MG, Brazil
guliato@ufu.br

²Centro de Lógica e Epistemologia e IFCH, Unicamp
CP 6133 13081-970 Campinas, SP, Brazil
carniell@cle.unicamp.br

³Laboratório de Sistemas Integráveis, Escola Politécnica de São Paulo, Divisão de Sistemas Digitais
Av. Prof. Luciano Gualberto, travessa 3, nro. 158, 05508-900 São Paulo, SP, Brasil
guliato, jazuffo@lsi.usp.br

Abstract. The objective of this paper is to propose an abstract concept of fusion operator based on finite automata and fuzzy sets. The proposed operator is able to integrate different sources of information, specially interesting for image processing. In particular, we give an example of application of this method to the problem of image segmentation, combining region growing and edge detection segmentations.

1 Introdução

A segmentação de imagens pode ser definida como um processo que agrupa pixels de uma imagem com relação a uma ou mais propriedades locais [7].

Considerando os estudos sobre as preferências do agrupamento perceptual humano [10], as técnicas de segmentação dividem-se em basicamente duas categorias: técnicas que exploram características de descontinuidade e técnicas que exploram características de similaridade. A primeira baseia-se nas mudanças abruptas dos níveis de cinza da imagem enquanto que a segunda baseia-se na homogeneidade das propriedades locais dos pixels.

As técnicas baseadas em características de descontinuidade são conceitualmente simples, no entanto produzem imagens com arestas geralmente desconectadas exigindo a aplicação de um algoritmo de detecção de contornos.

As técnicas baseadas em características de similaridade dividem a imagem em áreas que compartilham propriedades comuns. O resultado da segmentação orientada à região é dependente da seleção do pixel ou subregião inicial e a forma final da região de interesse pode depender da escolha de um algoritmo em particular.

Não existe, portanto, um algoritmo que possa ser considerado bom para todas as imagens, nem todos os algoritmos são igualmente bons para uma imagem em especial.

Uma proposta para melhorar o resultado final da segmentação é combinar os resultados obtidos por algoritmos diferentes que explorem características complementares da imagem de tal maneira que se possa obter um resultado melhor do que seria se apenas um deles fosse

considerado. Esta afirmação se justifica pelo seguinte:

1. métodos de segmentação diferentes exploram características diferentes da imagem, sendo que algumas dessas características são complementares e podem cooperar entre si para a obtenção de resultados mais refinados. Este é o caso, por exemplo, dos métodos de crescimento de região e de detecção de arestas.
2. A aplicação de técnicas isoladas pode resultar em erros. Em particular um algoritmo baseado em crescimento de região pode resultar em três tipos de erros: o limite da região é detectado mas nenhuma aresta real corresponde a esse limite; uma aresta está presente, mas não é detectada; o limite detectado não coincide com as arestas próximas a ele [14]. A combinação adequada de características de homogeneidade e de descontinuidade podem minimizar os erros mencionados.

Explorar características diferentes da imagem pode tornar o processo de segmentação mais robusto. No entanto, não são muitos os trabalhos apresentados com este enfoque [5], [2], [14], [9], [1] [4], [8]. Talvez este fato se deva à dificuldade de se definir mecanismos que avaliem mais de uma propriedade da imagem simultaneamente e/ou à dificuldade de se obter resultados satisfatórios a partir da combinação booleana de dois ou mais conjuntos clássicos, dado que, em termos da Álgebra de Boole, só podemos operar com união, interseção e complemento.

Uma classe de operadores, que generaliza os operadores booleanos, denominados *operadores de agregação* são estudados pela teoria dos conjuntos difusos [12], [15], [13].

Se os resultados da segmentação forem representados por conjuntos difusos [16], então poderemos integrar, através destes operadores, os resultados de segmentações obtidos independentemente um do outro.

Vários trabalhos sobre operadores de agregação têm sido apresentados na literatura, que, no entanto, não são gerais o suficiente para serem diretamente aplicados ao problema de segmentação de imagens [15], [13], [3].

O objetivo deste trabalho é propor um novo operador de agregação orientado por autômatos finitos que seja capaz de integrar os resultados obtidos por algoritmos de segmentação difusa, clássica em particular, aplicados independentemente um do outro, e que explorem características complementares da imagem.

Este trabalho está organizado da seguinte forma: A seção 2 apresenta os conceitos básicos sobre comportamentos e classificações de operadores de agregação e a seção 3 introduz o conceito de autômatos de estados finitos com o objetivo de familiarizar o leitor com relação a este assunto. A seção 4 propõe uma extensão para o comportamento e a classificação dos operadores de agregação apresentados na seção 3. A seção 5 apresenta formalmente o novo operador de agregação orientado por autômatos finitos. Ressaltamos que esta definição é bastante abrangente, podendo ser aplicada a uma vasta gama de problemas. A seção 6 particulariza o operador de agregação proposto de tal forma que seja capaz de combinar resultados de segmentações que explorem características diferentes da imagem. A seção 7 apresenta os resultados obtidos e a seção 8 faz algumas considerações finais sobre o trabalho.

2 Conceitos Básicos sobre Operadores de Agregação

Operações de agregação sobre conjuntos difusos são operações através das quais vários conjuntos difusos, os clássicos em particular, são combinados de forma apropriada para produzir um único conjunto difuso [6], [15].

Formalmente, qualquer operação de agregação sobre conjuntos difusos é definido pela função:

$$h : [0, 1]^n \rightarrow [0, 1], \text{ onde } n \geq 2$$

Particularizando, os conectivos de união e a interseção são operadores de agregação para $n = 2$.

Os operadores de agregação podem ser classificados de acordo com seu comportamento que podem ser (1) conjuntivo, (2) disjuntivo, (3) de compromisso. Os trabalhos [3], [12], [15] apresentam vários operadores de agregação levando em consideração esta classificação.

1. **Operadores Conjuntivos** - são aqueles que representam um consenso entre as informações sendo combinadas. Confiam na fonte que fornece a menor medida e buscam satisfação simultânea de critérios e objetivos. O comportamento do operador conjuntivo é *severo*. Os operadores conjuntivos são represen-

tados pelos operadores chamados T-norma e se caracterizam pelas seguintes propriedades: são comutativos, associativos, monotonicamente não decrescente e o elemento identidade é 1 [13]. Portanto um operador é dito conjuntivo se

$$h(a_1, a_2, \dots, a_n) \leq \min\{a_1, a_2, \dots, a_n\}, \text{ onde } a_i \in [0, 1].$$

2. **Operadores Disjuntivos** - são aqueles que aumentam a certeza que temos sobre uma das fontes. Confiam na fonte que fornece a maior medida e expressam redundância entre critérios. Têm comportamento *indulgente*. Os operadores disjuntivos são representados pelos operadores chamados T-conorma e se caracterizam pelas seguintes propriedades: são comutativos, associativos, monotonicamente não decrescente e o elemento identidade é 0 [13]. Portanto um operador é disjuntivo se

$$h(a_1, a_2, \dots, a_n) \geq \max\{a_1, a_2, \dots, a_n\}, \text{ onde } a_i \in [0, 1].$$

3. **Operadores de Compromisso** - são aqueles que resultam numa medida intermediária entre as medidas obtidas por diversas fontes. Apresentam um comportamento *cauteloso*. Um operador é de compromisso se

$$\min\{a_1, \dots, a_n\} \leq h(a_1, \dots, a_n) \leq \max\{a_1, \dots, a_n\}, \text{ onde } a_i \in [0, 1]$$

Bloch em [3] apresenta uma classificação que descreve os operadores de agregação não apenas como conjuntivo, disjuntivo ou de compromisso, mas também em termos de seu comportamento com relação aos valores das variáveis a serem combinadas (variáveis de entrada). A autora divide os operadores em três classes:

1. **Operadores com Comportamento Constante Independente de Contexto** - são os operadores que mantêm o mesmo comportamento, independente dos valores das variáveis de entrada. O operador de agregação deve satisfazer, de forma mutuamente exclusiva, um dos três comportamentos descritos acima.
2. **Operadores com Comportamento Variável Independente de Contexto** - são operadores independentes de contexto como no item anterior, mas cujo comportamento depende dos valores das variáveis de entrada. Por exemplo, um operador desta classe pode ter comportamento severo se os valores de entrada de todas as fontes forem altas, pode ter comportamento indulgente se todos os valores de entrada forem baixos e de compromisso no caso em que nenhuma das condições acima forem satisfeitas.
3. **Operadores Dependentes de Contexto** - são operadores que levam em consideração não apenas os

valores das variáveis de entrada a serem combinadas mas também dependem de um conhecimento global ou de parâmetros que indiquem conflito ou confiabilidade das fontes. Por exemplo, é possível definir operadores que se comportem de maneira conjuntiva se as fontes forem consonantes (baixo conflito), de maneira disjuntivas se elas forem dissonantes (alto conflito) e como compromisso se elas forem parcialmente conflitantes.

3 Conceitos Básicos sobre Autômatos Finitos

O autômato finito é um modelo matemático de um sistema, com entradas e saídas discretas cujo principal interesse é sua capacidade de especificar finitamente linguagens (ou procedimentos) infinitários. O sistema pode ter qualquer número finito de configurações internas ou *estados*. O estado de um sistema resume informações relacionadas a entradas anteriores necessárias para determinar o comportamento do sistema nas entradas subsequentes [11]. Os autômatos finitos podem ser vistos como modelos abstratos de computabilidade, dos quais as máquinas de Turing são especializações.

Formalmente um autômato finito [11] consiste de um quintupla $(Q, \Sigma, \delta, q_0, F)$, onde:

1. Q é um conjunto finito de estados
2. Σ é o alfabeto finito de entradas
3. δ é a função de transição que mapeia $Q \times \Sigma \rightarrow Q$
4. q_0 é o estado inicial, sendo que $q_0 \in Q$
5. F é o conjunto de estados finais, sendo que $F \subseteq Q$

A interpretação da função de transição δ de um autômato finito é que $\delta(q_i, a) = q_j$ se e somente se o autômato pode ir do estado q_i para o estado q_j pela entrada do símbolo a . Eventualmente, q_i e q_j podem coincidir. Desta forma a *diagrama de transição* de um autômato finito é uma representação gráfica da função de transição do autômato. Se há uma transição de q_i para q_j com entrada a então há um arco dirigido do estado q_i para o estado q_j rotulado com a no diagrama de transição, veja figura 1.

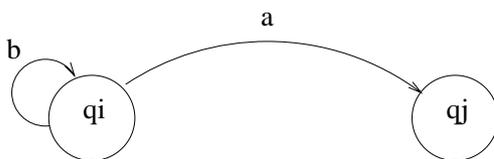


Figura 1: representação gráfica da função de transição $\lambda(q_i, a) = q_j, \lambda(q_i, b) = q_i$

4 Proposta de Extensão para os Operadores de Agregação

Os comportamentos e classificações apresentados na seção 2, em nossa opinião, não são suficientes para enquadrar todo tipo de operadores de agregação. Existem situações em que as fontes não são capazes de fornecer quaisquer informações indicando uma situação de incerteza. Neste caso, o comportamento do operador de agregação deve ser tal que reflita esta situação. Este operador não está necessariamente limitado a um comportamento conjuntivo, disjuntivo ou de compromisso, conforme definido acima. Sendo assim definimos um outro comportamento para os operadores de agregação: *operadores que indicam incerteza*. Conforme a classificação apresentada por Bloch em [3], os operadores de agregação podem ter comportamentos diferentes dependendo dos valores das variáveis de entrada. No entanto, determinar este comportamento nem sempre é possível por algoritmos simples. Em algumas situações o comportamento do operador de agregação pode depender não apenas dos valores das variáveis de entrada como também das operações executadas por ele até aquele momento, ou seja, o comportamento pode depender das variáveis de entrada e do estado atual do operador de agregação. Para atender estas exigências propomos uma extensão para o comportamento e para a classificação de operadores de agregação

4.1 Proposta de Extensão para o Comportamento dos Operadores de Agregação

Operadores que Indicam Incerteza - são aqueles que refletem a falta de informações das fontes sendo combinadas. A imagem (*range*) destes operadores não está limitada aos intervalos impostos pelos operadores T-norma e T-conorma. Na seção 6.1 apresentaremos um exemplo deste comportamento.

4.2 Proposta de Extensão para a Classificação dos Operadores de Agregação

Operadores Dependentes de Contexto e do seu Estado Atual - são operadores dependentes de contexto e das variáveis de entrada e que dependem também das operações executadas pelo operador até o presente momento. Esta classificação é particularmente interessante quando as fontes sendo combinadas apresentam informações algebricamente diferentes.

5 Operadores Orientados por Autômatos Finitos

Formalmente, o operador de agregação orientado por autômatos finitos que agrega n fontes pode ser definido como sendo um par ordenado $\langle H, M \rangle$, onde:

1. $H = \{h_1, h_2, \dots, h_k\}$ é um conjunto finito de operadores de agregação, sendo que h_i são operadores de $[0, 1]^n \rightarrow [0, 1]$ para $n \geq 2$
2. $M = (Q, \Sigma, \delta, q_0, F)$ é um autômato de estados finitos, onde:
 - (a) Q é um conjunto finito de estados
 - (b) Σ é o alfabeto finito de entradas
 - (c) δ é a função de transição que mapeia $Q \times \Sigma \rightarrow Q$
 - (d) q_0 é o estado inicial, sendo que $q_0 \in Q$
 - (e) F é o conjunto de estados finais, sendo que $F \subseteq Q$

Para atender especificamente ao problema proposto devemos definir um conjunto H de operadores de agregação, um alfabeto finito de entradas Σ e um diagrama de transição adequados para este propósito. Em particular, o alfabeto de entrada do autômato deve ser finito. No entanto as informações a serem combinadas são graus de pertinência representados no intervalo $[0, 1]$, produzindo, portanto, um conjunto infinito de entradas. Para tornar o conjunto de entradas um conjunto finito, sem perder a generalização, propomos que este seja obtido por uma função Γ que mapeia o produto cartesiano de partições finitas de $[0, 1]$ para Σ .

Formalmente o alfabeto de entrada deve ser constituído da seguinte forma: Sejam Π_i partições finitas de $[0, 1]$, então existe uma função Γ , tal que $\Gamma : \Pi_1 \times \Pi_2 \times \dots \times \Pi_n \rightarrow \Sigma$ onde n é o número de fontes a serem combinadas.

6 Integração de Segmentações Complementares através do Operador de Agregação Orientado por Autômatos Finitos

Nesta seção apresentaremos o operador orientado por autômatos finitos proposto aplicado ao problema de segmentação de imagens. Neste caso, o operador deve ser capaz de integrar os resultados obtidos por duas segmentações obtidas de uma mesma imagem, uma que explora características de continuidade e outra que explora características de descontinuidade na variação dos níveis de cinza. Os resultados das segmentações devem ser representados por conjuntos difusos, sendo que cada elemento destes conjuntos é referenciado como grau de pertinência difusa [15]. Formalmente,

Seja Φ uma função de pertinência difusa.

Seja $\Phi : I \rightarrow [0, 1]$ uma função de pertinência Φ que mapeia I em $[0, 1]$. Para qualquer pixel p da imagem I , $\Phi(p)$ é denominado grau de pertinência de p em Φ .

Como resultado, o operador deve produzir um conjunto difuso que reflita as certezas e incertezas identi-

ficadas durante esta integração. Surge aqui um outro conceito discutido na teoria dos conjuntos difusos: como representar a incerteza?

Como os graus de pertinência estão sendo representados no intervalo real unitário, a certeza máxima ocorre quando $\Phi(p) = 0$ ou $\Phi(p) = 1$. Por outro lado, a incerteza máxima ocorre quando $\Phi(p) = 0.5$, o que significa que a função de pertinência não foi capaz de indicar qualquer evidência quanto a pertinência do pixel sendo analisado

6.1 Características Gerais do Operador de Agregação sendo Proposto

Para definirmos formalmente o operador de agregação para o problema em questão, devemos fazer algumas considerações gerais quanto ao seu comportamento. Para tanto, consideremos o exemplo da figura 2. Neste exemplo, sobreposemos os resultados de duas segmentações, uma que identifica a região (região hachurada) e a outra que identifica o contorno de uma região de interesse fictícia. Os resultados não coincidem, o que significa dizer que algoritmos de segmentação diferentes aplicados a uma mesma imagem produziram resultados diferentes.

Considerando estes resultados o operador de agregação deve produzir um terceiro resultado que leve em consideração os resultados originais e que seja melhor que cada um individualmente. Considerando a posição espacial dos pixels durante o processo de combinação, podemos tirar, deste exemplo, algumas conclusões sobre qual deve ser o comportamento esperado do operador de agregação em face às diferentes situações possíveis.

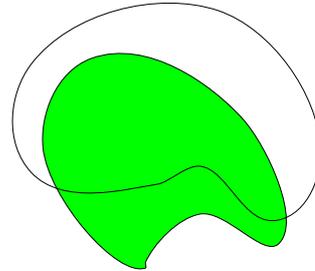


Figura 2: sobreposição de duas segmentações que exploram características complementares

Destacamos quatro diferentes situações que devem ser adequadamente tratadas pelo operador de agregação. Estes situações são indicadas na figura 3 e discutidas a seguir:

1. *pixels que pertencem a intersecção das duas segmentações.* Neste caso os pixels devem pertencer com certeza ao resultado final da integração. O operador deve ter um *comportamento indulgente*.

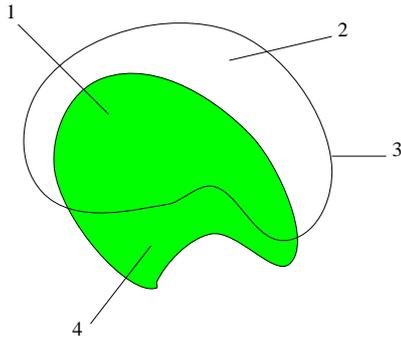


Figura 3: diferentes situações tratadas pelo operador de agregação

2. *pixels que não pertencem à região, mas encontram-se no interior do contorno.* Faltam, neste caso, informações sobre a pertinência ou não do pixel ao resultado final, indicando uma situação de incerteza, que deve ser representada no resultado final. O operador deve ter um *comportamento que indique incerteza*.
3. *pixels que pertencem somente ao contorno.* Indicam uma situação de transição, e esta transição deve ser adequadamente representada no resultado da integração. O *comportamento deve ser de compromisso*.
4. *pixels que pertencem à região mas que já transpuseram os limites impostos pelo contorno.* Neste caso, como no item 2, configura-se uma situação de incerteza. Portanto o operador deve ter um *comportamento que indique incerteza*.

Considerando o estudo feito, percebemos que, devido ao fato de estarmos integrando conjuntos com dados complementares, a informação espacial do pixel sendo analisado é uma informação relevante ao processo de combinação, ou seja, o tipo de operação de agregação a ser executada depende desta posição. Este fato pode ser exemplificado da seguinte maneira: a segmentação baseada em região representa os pixels que pertencem à região com graus de pertinência “alto”, enquanto que a segmentação baseada em contorno representa os pixels do interior do contorno (portanto, pixels que pertencem à região, segundo ele) com graus de pertinência “baixo”. Sendo assim, pixels que pertencem à região e são internos ao contorno são representados por graus de pertinência “alto” na segmentação baseada em região enquanto que estes mesmos pixels são representados por graus de pertinência “baixo” na segmentação baseada em contorno. Nesse caso, como mostrado na figura 3, indicação 1, a situação é de certeza máxima. Notamos que, a segmentação baseada em contorno também representa os pixels localizados no exterior do contorno (pixels não pertencentes à região), por graus de pertinência “baixo”. No caso da região transpor o contorno (pixels com graus de pertinência “alto”

para região e “baixo” para contorno), conforme mostrado na figura 3 indicação 4, a situação identificada pelo operador deve ser de incerteza. Verificamos que para uma mesma entrada (pixels com graus de pertinência “alto” para região e “baixo” para contorno), o operador de agregação deve executar procedimentos diferentes. Esta observação nos leva a pensar na posição espacial do pixel sendo analisado como estados que determinam qual operação de agregação deverá ser realizada. Sendo assim, propomos que a escolha das operações de agregação a serem executadas durante o processo de integração seja orientada por um autômato de estados finitos [11]. Os estados do autômato são determinados pelas diferentes posições espaciais que um pixel pode assumir dentro deste contexto, e o diagrama de transições depende da forma como os pixels da imagem serão percorridos.

Outra consideração importante é a questão da confiabilidade das fontes sendo combinadas. Nesta aplicação em particular, uma das segmentações, pode apresentar maior confiabilidade que a outra. Neste caso o operador de agregação precisa considerar mais esta informação.

Para permitir uma ponderação entre as fontes, consideramos que a *força da região*, FR somada com a *força do contorno*, FC deve ser igual a 2, isto é, $FR + FC = 2$. Isto significa dizer que:

1. $FR = 2$ e $FC = 0$, então as informações dadas pela segmentação baseada em continuidade são privilegiadas.
2. $FR = 0$ e $FC = 2$, então as informações dadas pela segmentação baseada em descontinuidade são privilegiadas.
3. $FR = 1$ e $FC = 1$, então o resultado da combinação deve levar em consideração as informações apresentadas pelas duas fontes igualmente.
4. qualquer outra combinação para FC e FR , satisfazendo $FC + FR = 2$, o resultado da agregação deve ponderar as informações apresentadas de acordo com a confiabilidade de cada fonte.

Como a confiabilidade das fontes é expressa por pesos complementares em 2, utilizaremos no operador de agregação apenas a confiabilidade dada por FC .

6.2 Características Específicas do Operador de Agregação sendo Proposto

O operador de agregação orientado por autômatos finitos $OP = \langle H, M \rangle$ aplicado ao problema de integração de duas segmentações complementares será definido a seguir.

Sejam S_1 e S_2 duas imagens segmentadas a serem integradas, a primeira baseada em crescimento de região

e a segunda baseada em detecção de arestas com contorno fechado.

Seja p_{ij} um pixel da imagem S_i , para $j = 1, m$, onde m é o número de pixels da imagem, onde $p_{ij} \in [0, 1]$.

Os pixels da imagem serão analisados a partir de um *pixel semente* selecionado interativamente pelo usuário. Este pixel semente deve pertencer à intersecção das duas segmentações sob consideração. Esta restrição não compromete a generalização do operador, uma vez que o objetivo é integrar segmentações com certo grau de confiabilidade. Se for difícil ou impossível localizar um pixel em comum, podemos concluir que uma ou ambas as segmentações não representam adequadamente o objeto de interesse, devendo ser, portanto, reprocessadas. Os operadores de agregação do conjunto H de OP serão apresentados a seguir:

1. $h_1 = 1$
2. $h_2 = 0.5 * ((p_{1j} * (2 - FC) + (p_{2j} * FC))$
3. $h_3 = \max\{0.5 * FC, p_{1j}\}$
4. $h_4 = \min\{1 - (0.5 * FC), p_{1j}\}$

O autômato finito $M = (Q, \Sigma, \delta, q_0, F)$, é definido como:

1. $Q = \{a, b, c\}$, sendo que os estados indicam a posição espacial do pixel sendo analisado:
 - (a) o estado “a” indica que o pixel sendo analisado pertence ao interior do contorno
 - (b) o estado “b” indica que o pixel sendo analisado pertence ao contorno
 - (c) o estado “c” indica que o pixel sendo analisado pertence à região que transpôs o contorno
2. $\Sigma = \{1, 2, 3, 4\}$. Devido à restrição imposta pelo pixel semente a entrada inicial do autômato deve ser sempre igual a ‘1’.

O alfabeto de entrada Σ é constituído da seguinte maneira:

Sejam Π_1 e Π_2 duas partições finitas de $[0, 1]$, onde $\Pi_1 = \Pi_2 = \{alto, baixo\}$

Para uma dada segmentação S_i , então:

$alto = \{p_{ij} | \Phi(p_{ij}) \geq 0.5\}$ para $j = 1, m$ e

$baixo = \{p_{ij} | \Phi(p_{ij}) < 0.5\}$ para $j = 1, m$

Pelo fato de

$\Gamma : \Pi_1 \times \Pi_2 \rightarrow \Sigma$

Temos que

$\Gamma : \{alto, baixo\} \times \{alto, baixo\} \rightarrow \Sigma$

Identificamos, desta forma, quatro tipos de variáveis de entrada possíveis baseando-se no grau de pertinência dos pixels em cada uma das segmentações sob consideração:

- (a) $\Gamma : (alto, baixo) \rightarrow 1$: O pixel apresenta grau de pertinência alto na segmentação baseada em região e baixo na segmentação baseada em contorno. Indica uma situação de intersecção se o pixel pertencer ao interior do contorno, caso contrário indica uma situação de incerteza.
- (b) $\Gamma : (alto, alto) \rightarrow 2$: O pixel apresenta grau de pertinência alto na segmentação baseada em região e alto na segmentação baseada em contorno. Indica uma situação de intersecção.
- (c) $\Gamma : (baixo, alto) \rightarrow 3$: O pixel apresenta grau de pertinência baixo na segmentação baseada em região e alto na segmentação baseada em contorno. Indica uma situação de contorno.
- (d) $\Gamma : (baixo, baixo) \rightarrow 4$: O pixel apresenta grau de pertinência baixo na segmentação baseada em região e baixo na segmentação baseada em contorno. Indica uma situação de incerteza se o pixel pertencer ao interior do contorno, caso contrário indica não pertinência.

3. δ é apresentado no diagrama de estados da figura 4. Entradas não representadas no diagrama de estados são consideradas inválidas.
4. $q_0 = \{a\}$
5. $F = \{c\}$

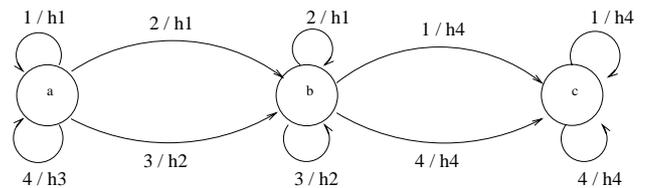


Figura 4: diagrama de transição do autômato que orienta o comportamento do operador de agregação

7 Resultados Experimentais

Nesta seção apresentaremos alguns resultados do operador de agregação definido na seção anterior. A figura 5 apresenta uma imagem obtida por uma ventriculografia radioisotópica (*equilibrium gated blood pool*) cedida pelo Serviço de Radioisótopos da Divisão de Diagnósticos por Imagens do Instituto do Coração em São Paulo. A imagem apresentada é a primeira do ciclo cardíaco e corresponde ao final da diástole. As figuras 6 e 7 são resultados

de segmentações difusas do ventrículo esquerdo (região mais clara à direita da imagem original). A primeira foi obtida por crescimento de região e a outra foi obtida por detecção de arestas. O operador de agregação proposto suporta apenas segmentações baseadas em descontinuidade com contornos fechados. Observe que o algoritmo de segmentação baseado em homogeneidade, cujo resultado é mostrado na figura 6, propositalmente, não foi capaz de identificar o ventrículo esquerdo, com o objetivo de avaliar o comportamento do operador de agregação com relação à confiabilidade das fontes. A combinação, baseada no grau de confiabilidade de cada fonte, deve identificar as evidências de pertinência de cada pixel à região final. Em particular, graus de pertinência iguais a 0.5 indicam incerteza máxima, ou seja nenhuma das fontes foi capaz de indicar evidências quanto a pertinência ou não de um pixel à região segmentada. A incerteza máxima acontece quando ambas as fontes possuem a mesma confiabilidade e discordam quanto à pertinência do pixel à região de interesse, veja figura 8. Na medida em que uma fonte torna-se mais confiável que a outra, o resultado da agregação se altera, veja figuras 9 e 10. No estudo apresentado, a confiabilidade das fontes é obtida de forma subjetiva.

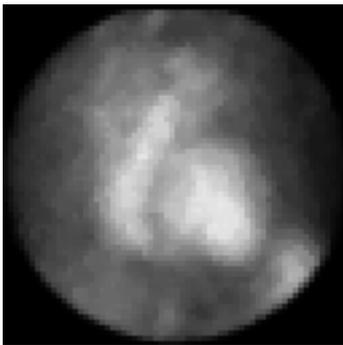


Figura 5: Imagem Original 64×64



Figura 6: segmentação baseada em região



Figura 7: segmentação baseada em contorno

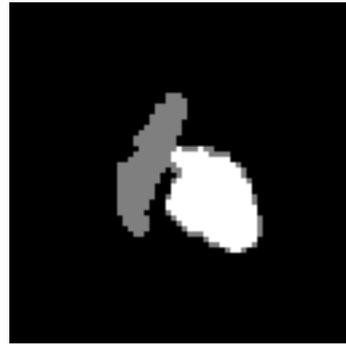


Figura 8: resultado da operação de agregação com $FC=1$

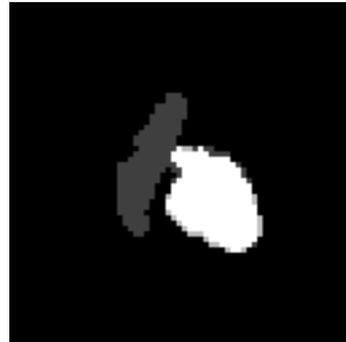


Figura 9: resultado da operação de agregação com $FC=1.5$



Figura 10: resultado da operação de agregação com $FC=2$

8 Considerações Finais

O operador orientado por autômatos finitos proposto neste trabalho é uma abstratação de operadores de agregação, tornando a maioria dos outros operadores apresentados na literatura como u casos particulares deste. A definição do operador proposto é bastante abrangente, e a exploração adequada do conceito apresentado pode conduzir à solução de vários problemas interessantes. Em particular, propusemos um operador de agregação capaz de integrar o resultado de duas segmentações complementares e demonstramos, através de exemplos, o interesse da definição. Ressaltamos que, como a definição do operador de agregação é bastante geral, a solução que encontramos para integrar as duas segmentações não é única.

9 Agradecimentos

Este trabalho contou com o apoio financeiro da CAPES, programa PICDT.

Referências

- [1] H. ASAR; N. NANDHAKUMAR ; J.K. AGGARWAL. Pyramid-based image segmentation using multisensory data. *Pattern Recognition*, 23(6):583–593, 1990.
- [2] F. MEYER; S. BEUCHER. Morphological segmentation. *Journal of Visual Communication and Image Representation*, 1(1):21–46, September 1990.
- [3] I. BLOCH. Information combination operators for data fusion: a comparative review with classification. *IEEE Trans. on systems, man, cybernetics - part a: systems and humans*, 26(1), january 1996.
- [4] J.F. HADDON; J.F. BOYCE. Image segmentation by unifying region growing and boundary information. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(10), october 1990.
- [5] C.C. CHU. The integration of image segmentation maps using region and edge information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(12):1241–1252, december 1993.
- [6] G.J. KLIR; T.A. FOGER. *Fuzzy Sets, Uncertainty and Information*. Prentice Hall, 1988.
- [7] A. ROSENFELD; A.C. KAK. *Digital Picture Processing*. Academic Press, Inc, second edition, 1982.
- [8] T.PAVLIDIS; Y-T. LIOW. Integrating region growing and edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(3), march 1990.
- [9] L. VINCENT; P. SOILLE. Watershed in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, June 1991.
- [10] A.L. ZOBRIST;W.B. THONPSON. Building a distance function for gestalt grouping. *IEEE Trans. on Computer*, 7(c-4):718–728, july 1975.
- [11] J.E. HOPCROFT; J.D. ULLMAN. *Introduction to Automata Theory Languages and Computaton*. Addison-Wesley Publishing Company, 1979.
- [12] R.R. YAGER. Connectives and quantifiers in fuzzy sets. *Fuzzy Sets and Systems*, 40:39–75, 1991.
- [13] R.R. YAGER. An introduction to aggregation operations on fuzzy sets. In *Sixth International Fuzzy Systems Associations World Congress*, pages 158–180, 1995.
- [14] Y. XIAOHAN; J. YLA-JAASKI. Direct segmentation in 3d and its application to medical images. In *SPIE image processing*, volume 1898, pages 187–192, 1993.
- [15] G.j. KLIR; B. YUON. *Fuzzy Sets and Fuzzy Logic*. Printice Hall PTR, 1995.
- [16] L.A. ZADEH. Fuzzy sets. *Information and Control*, 8:338–353, 1965.