

Técnica de segmentação hierárquica de imagem baseada em grafos com pesos

Thais Lima de Sousa

Orientador: Ronaldo Fumio Hashimoto

IME-USP

2 de Setembro de 2016

1 Tópicos estudados até agora

- Introdução
- Segmentação eficiente de imagem baseada em grafos: o algoritmo de Felzenswalb & Huttenlocher (2004)
- Segmentação Hierárquica baseada em grafos: o algoritmo de Guimarães et al. (2012)

2 Próximos tópicos a serem estudados

O problema de *segmentação de imagem* consiste em agrupar pixels similares em regiões disjuntas de forma que o conjunto de todas as regiões seja uma partição dos conjunto de pixels da imagem. Dessa forma, uma segmentação de imagem nada mais é do que uma partição de seu conjunto de pixels.

Uma *segmentação hierárquica de imagem* é uma coleção de segmentações de imagem sob diferentes níveis de detalhes de forma que segmentações em níveis de detalhes mais grossos são obtidas pela fusão de regiões de segmentações em níveis de detalhes mais finos. De fato, segmentação hierárquica de imagem tem a propriedade de preservar informações de vizinhança entre os pixels e das regiões segmentadas.

Segmentação Baseada em Grafos

Seja $G = (V, E)$ um grafo não dirigido com vértices $v_i \in V$, o conjunto de elementos a serem segmentados, e arestas $(v_i, v_j) \in E$ que correspondem aos pares vizinhos de vértices. Cada aresta $(v_i, v_j) \in E$ possui um peso correspondente $w((v_i, v_j)) \geq 0$ que é a medida de dissimilaridade entre os elementos vizinhos v_i e v_j .

Em segmentação de imagens, os elementos em V são pixels e o peso associado a uma aresta é a medida de dissimilaridade (e.g. diferença de intensidade, cor, localização, etc.) entre os pixels conectados por essa aresta.

Uma segmentação S é uma partição de V em componentes de forma que cada componente (ou região) $C \in S$ corresponde a uma componente conexa em um grafo $G' = (V, E')$, $E' \subseteq E$. Toda segmentação é induzida por um subconjunto de arestas de E .

Qualidade de segmentação: Arestas entre dois vértices na mesma componente devem ter pesos relativamente baixos, e arestas entre vértices de componentes diferentes devem ter pesos maiores.

O Algoritmo de Felzenszwalb & Huttenlocher

A técnica desenvolvida por seleciona arestas de um grafo (como descrito anteriormente), ajustando um critério de segmentação baseado no grau de variabilidade entre regiões vizinhas da imagem. Isso resulta em um método que, enquanto faz decisões gulosas, mostra-se de acordo com certas propriedades globais.

Predicado de Comparação de Regiões Similares:

Um predicado D avalia se existe evidência para um contorno entre duas componentes relativo a medida de dissimilaridade entre os elementos vizinhos dentro de cada uma das duas componentes. O predicado compara as diferenças inter-componentes às diferenças intra-componentes e, portanto, é adaptativa em relação às características locais dos dados.

O Algoritmo de Felzenszwalb & Huttenlocher

Predicado de Comparação de Regiões Similares

Diferença interna

A diferença interna de uma componente $C \subseteq V$ é o maior peso associado a alguma aresta na $MST(C, E)$:

$$Int(C) = \max_{e \in MST(C, E)} w(e)$$

O Algoritmo de Felzenszwalb & Huttenlocher

Predicado de Comparação de Regiões Similares

Diferença entre duas componentes

A diferença entre duas componentes $C_1, C_2 \subseteq V$ é a aresta de peso mínimo conectando duas componentes:

$$Dif(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2, (v_i, v_j) \in E} w((v_i, v_j))$$

O Algoritmo de Felzenszwalb & Huttenlocher

Predicado de Comparação de Regiões Similares

O predicado verifica se $Dif(C_1, C_2)$ grande em relação à diferença interna dentro de pelo menos uma das componentes de $Int(C_1)$ e $Int(C_2)$. Uma função de limiarização é usada para controlar o grau a partir do qual a diferença entre componentes devem ser maiores que a diferença interna mínima.

Predicado de Comparação de Regiões Similares

$$D(C_1, C_2) = \begin{cases} \text{TRUE} & \text{if } Dif(C_1, C_2) > MInt(C_1, C_2) \\ \text{FALSE} & \text{otherwise} \end{cases}$$

onde a diferença interna mínima vale:

$$MInt(C_1, C_2) = \left\{ Int(C_1) + \frac{k}{|C_1|}, Int(C_2) + \frac{k}{|C_2|} \right\}$$

Onde k é um parâmetro que previne a fusão de regiões grandes (valores grandes de k "forçam" a fusão de regiões menores).

O Algoritmo de Felzenszwalb & Huttenlocher I

Entrada: grafo $G = (V, E)$ com n vértices e m arestas

Saída: uma segmentação de V em componentes $S = C_1, \dots, C_n$

- 1 Ordena E em $\pi = (o_1, \dots, o_m)$ por ordem não decrescente do peso das arestas;
- 2 Comece com uma segmentação S^0 , onde cada vértice v_i está em sua própria componente
- 3 Repita o passo 4 para $q = 1, \dots, m$;

O Algoritmo de Felzenszwalb & Huttenlocher II

- 4 Construa S^q , dado S^{q-1} da seguinte forma:
Considere v_i e v_j , os vértices conectados pela q -ésima aresta na ordenação, i.e. $o_q = (v_i, v_j)$. Se v_i e v_j estão em componentes disjuntas de S^{q-1} e $w(o_q)$ é pequeno comparado à diferença interna de ambas as componentes, então fundimos as duas componentes, caso contrário não fazemos nada.
Em outras palavras (em relação ao passo 4): seja C_i^{q-1} a componente de S^{q-1} contendo v_i e C_j^{q-1} a componente contendo v_j . Se $C_i^{q-1} \neq C_j^{q-1}$ e $w(o_q) \leq \text{MInt}(C_i^{q-1}, C_j^{q-1})$, então S^q é obtida a partir de S^{q-1} a partir da fusão entre C_i^{q-1} e C_j^{q-1} . Caso contrário, $S^q = S^{q-1}$.
- 5 Retorne $S = S^m$.

O Algoritmo de Felzenszwalb & Huttenlocher

Problemas do método:



(a) Original



(b) $k = 7500$ (8)



(c) $k = 9000$ (14)

Fig. 1. A real example illustrating the violation of the causality principle by [3]: the number of regions (in parentheses) increases from 8 to 14, instead of decreasing when the so-called “scale of observation” increases

Problemas do método:

- O número de regiões pode aumentar quando o parâmetro k aumenta. Isso não deveria ocorrer se k for uma verdadeira "escala de observação". Isso viola o *Princípio de Causalidade* - um contorno presente em uma escala k_1 deveria estar presente em toda escala $k_2 < k_1$ (o número de regiões deveria diminuir quando a "escala de observação" aumenta).
- Até mesmo quando o número de regiões decresce, os contornos não são estáveis: eles podem se mover quando o parâmetro k varia, violando o *Princípio de Localização*.

Segmentação Hierárquica: o algoritmo de Guimarães et al.

Métodos hierárquicos têm como propriedade a preservação de informações de espaço e vizinhança entre as regiões segmentadas. No método de Guimarães et al., é usada a mesma representação em grafos descrita anteriormente, e cada hierarquia pode ser representada por uma árvore geradora mínima (MST).

O método consiste em calcular uma hierarquia de partições baseada em escalas de observação.

Segmentação Hierárquica baseada em grafos com pesos

Considerando as definições de $Dif(C)$ e C_1, C_2 apresentadas anteriormente, definimos ocorre fusão entre duas regiões X e Y quando:

$$Diff(X, Y) \leq \min \left\{ Int(X) + \frac{k}{|X|}, Int(Y) + \frac{k}{|Y|} \right\}$$

E essa desigualdade depende da escala k na qual as regiões X e Y são observadas.

Segmentação Hierárquica baseada em grafos com pesos

Considere a *escala de observação* $S_Y(X)$ de X relativa a Y :

$$S_Y(X) = (Diff(X, Y) - Int(X)) * |X|$$

Então:

$$S(X, Y) = \max \{S_Y(X), S_X(Y)\}$$

Portanto podemos escrever:

$$k \geq S(X, Y)$$

As regiões vizinhas X e Y sofrem fusão quando a escala delas é menor que o parâmetro de limiarização k .

Noções de hierarquia

Considere w um mapa $E \rightarrow \mathbb{N}$ que associa pesos a arestas e um parâmetro $\lambda \in \mathbb{N}$ para limiarização.

Para toda árvore T geradora do conjunto V de pixels de imagem, todo mapa w e toda limiarização λ , podemos associar a partição P_λ^w de V induzida pelas componentes conexas do grafo criado por V e as arestas associadas a pesos $w < \lambda$.

Para dois valores quaisquer de λ_1 e λ_2 de forma que $\lambda_1 \geq \lambda_2$, as partições $P_{\lambda_1}^w$ e $P_{\lambda_2}^w$ são *aninhadas* e $P_{\lambda_1}^w$ é *mais grossa* que $P_{\lambda_2}^w$. Então o conjunto $\mathcal{H} = \{P_\lambda^w | \lambda \in \mathbb{N}\}$ é uma *hierarquia*.

Futuros estudos:

- Segmentação hierárquica e o algoritmo de Guimarães et al.
- Tarjan's Union Find
- Fredman and Tarjan's Fibonacci Heap
- Implementação

- Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. IJCV 59, 167–181 (2004).
- Guimarães, S.J.F., Cousty, J., Kenmochi, Y., Najman, L.: A hierarchical image segmentation algorithm based on an observation scale. In: Gimel'farb, G., Hancock, E., Imiya, A., Kuijper, A., Kudo, M., Omachi, S., Windeatt, T., Yamada, K. (eds.) SSPR & SPR 2012. LNCS, vol. 7626, pp. 116–125. Springer, Heidelberg (2012).

Fim da apresentação.