AUTO FEATURE WEIGHT FOR INTERACTIVE IMAGE SEGMENTATION USING PARTICLE COMPETITION AND COOPERATION

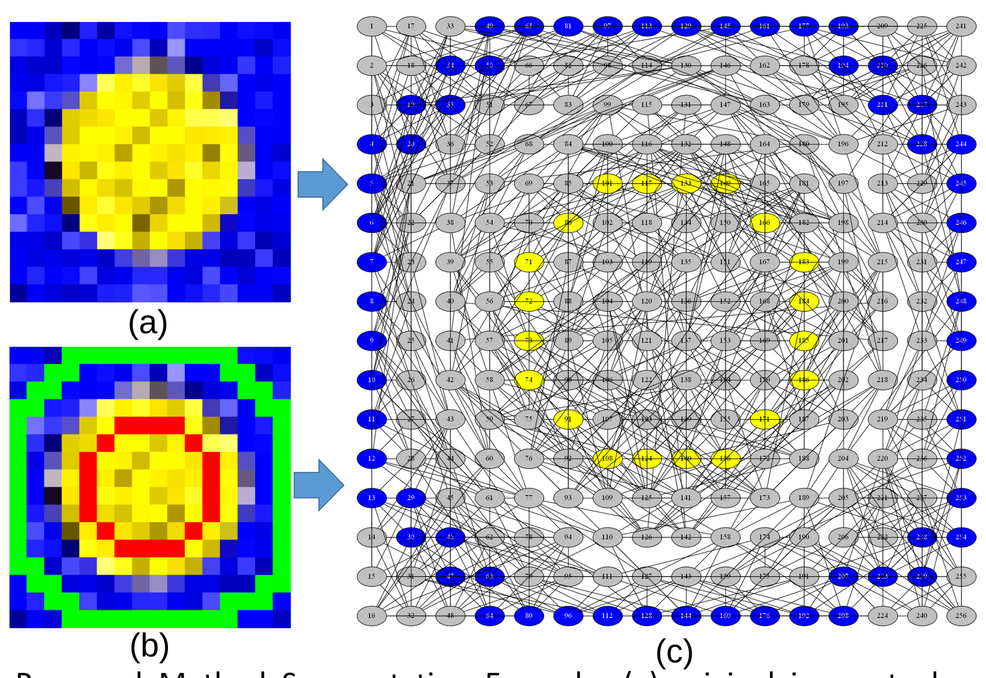
XI Workshop de Visão Computacional
October 05th - 07th 2015
São Carlos (SP) - Brazil

Fabricio Breve
São Paulo State University (UNESP)
fabricio@rc.unesp.br

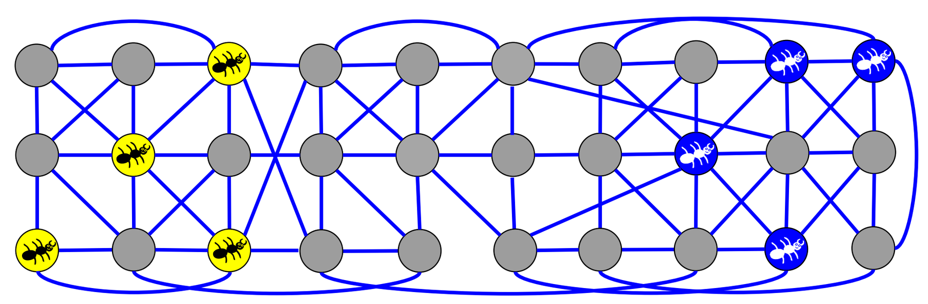
Abstract - Particle competition and cooperation (PCC) is a graph-based semi-supervised learning algorithm. It employs particles walking in a graph to classify data items corresponding to graph nodes. PCC may be applied to the interactive image segmentation problem, in which a human specialist labels some pixels of the image, and the algorithm propagates them to the remaining pixels, splitting the foreground and the background. In this scenario, pixels are converted into graph nodes, and each node is connected to its *k*-nearest neighbors, according to a set of features extracted from the image. However, some features may be more important than others to identify the segments, depending on the particularities of the image. In this paper, four methods to automatically weight the features to the neighbor selecting task are presented. Computer simulations are performed on real-world images from the Microsoft GrabCut dataset, and the segmentation results show the effectiveness of the proposed methods.

Particle Competition and Cooperation

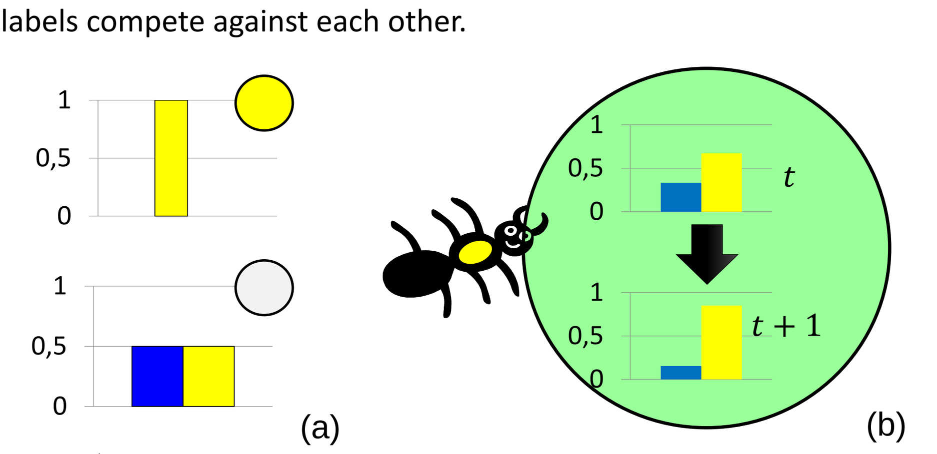
An undirected and unweight graph is generated from the image. Each pixel becomes a graph node. Each node is connected to its k-nearest neighbors according to some pixel features.



Proposed Method Segmentation Example: (a) original image to be segmented (16x16 pixels); (b) original image with user labeling (green and red traces); and (c) graph generated after the original image, where each image pixel corresponds to a graph node. Labeled nodes are colored blue and yellow, and unlabeled nodes are colored grey. Each labeled node will have a particle assigned to it.



A particle is generated for each labeled node. Particles initial position are set to their corresponding nodes. Particles with same label play for the same team and cooperate with each other. Particles with different labels compete against each other.

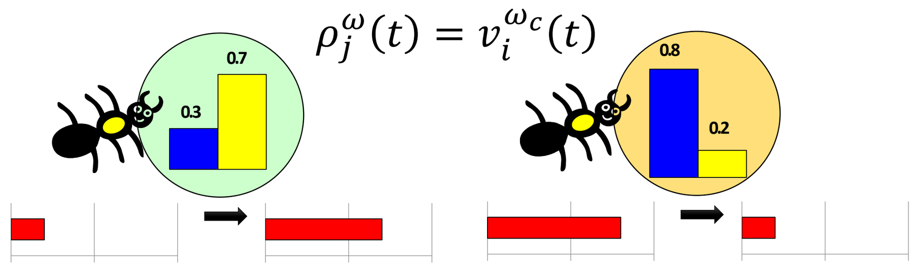


$$v_i^{\omega_c} = \begin{cases} 1 & \text{if } x_i \text{ is labeled } y(x_i) = c \\ 0 & \text{if } x_i \text{ is labeled } y(x_i) \neq c \\ 1/c & \text{if } x_i \text{ is unlabeled} \end{cases}$$

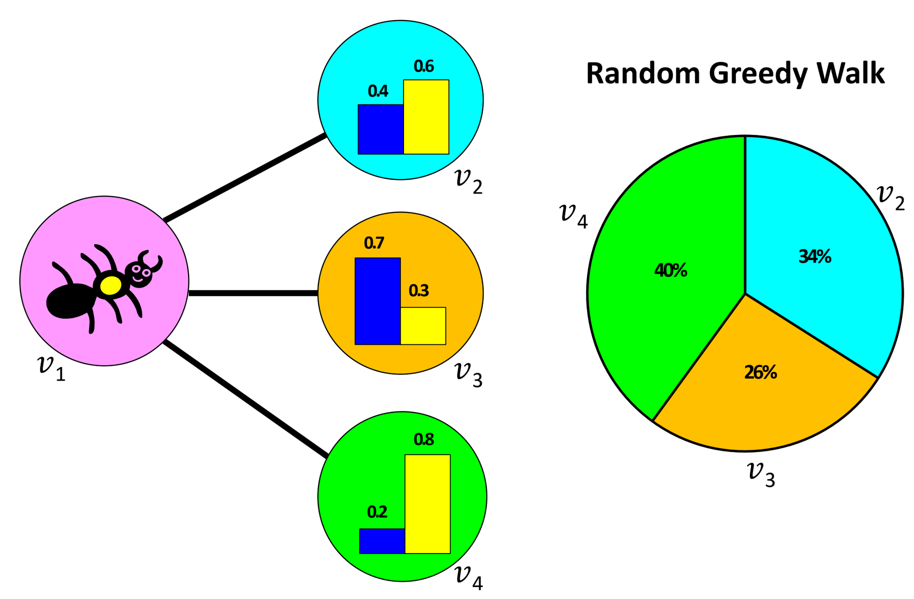
$$v_i^{\omega_c}(t+1) = \begin{cases} \max\left\{0, v_i^{\omega_c}(t) - \frac{0.1\,\rho_j^{\omega}(t)}{C-1}\right\} & \text{if } c \neq \rho_j^c \\ v_i^{\omega_c}(t) + \sum_{r \neq c} v_i^{\omega_r}(t) - v_i^{\omega_r}(t+1) & \text{if } c = \rho_j^c \end{cases}$$

(a) Nodes have a domination vector. Labeled nodes have ownership set to their respective teams (classes). Unlabeled nodes have ownership levels set equally for each team. (b) When a particle selects a neighbor to visit: it decreases the domination level of the other teams while it increases the domination level of its own team. Exception: labeled nodes domination levels are fixed.

$$\rho_j^{\omega}(t) = v_i^{\omega_c}(t)$$



A particle gets strong when it selects a node being dominated by its own team, but it gets weak when it selects a node being dominated by another team.



$$p(v_i|\rho_j) = \frac{W_{qi}}{2\sum_{\mu=1}^{N} W_{q\mu}} + \frac{W_{qi} v_i^{\omega_c} \left(1 + \rho_j^{d_i}\right)^{-2}}{2\sum_{\mu=1}^{N} W_{q\mu} v_\mu^{\omega_c} \left(1 + \rho_j^{d_\mu}\right)^{-2}}$$

Each particles randomly chooses a neighbor to visit at each iteration. Nodes which are already dominated by the particle team and closer to particle initial node have higher probabilities of being chosen.

Auto Feature Weight

From each pixel, 23 features are extracted: row, column, RGB and HSV components, mean and standard deviation of each RGB and HSV components including their adjacency, and ExR, ExG, and ExB components. For each feature, a corresponding weight is defined. The Euclidean distance between each pair of pixels, considering the weighted features, is used to define each pixel k-nearest neighbors to compose the graph.

Four methods are proposed to calculate the weights using only the information available (the subset of labeled pixels):

Intra-class mean difference (DMI): The mean of a given feature f among labeled pixels of the same class is calculated for each class. Then the difference between these means is used to define the weight:

$$\omega_f = \sum_{i|y_i=1} x_{if} - \sum_{i|y_i=2} x_{if}$$

where x_{if} is the value of feature f in pixel x_i, and y_i is x_i label.

Difference between Intra-Class Means divided by Intra-Class Standard Deviation Sum (DMI/SDPI): Similar to DMI, but the difference is then divided by the sum of the standard deviation of the feature in each class:

$$\omega_f = \frac{\sum_{i|y_i=1} x_{if} - \sum_{i|y_i=2} x_{if}}{\sigma_{f1} + \sigma_{f2}}$$

where σ_{f1} and σ_{f2} are the standard deviation of feature f in classes 1 and 2 , respectively.

Difference between Histograms (DH): Given a feature f, for each subset of pixels from the same class, a 10 bar histogram is generated. Both histograms have their values normalized by dividing each bar by the total amount of labeled pixels of the class, so the 10 bar sum is 1. Then the weights are defined as the sum of the absolute differences between the corresponding pair of bars in each histogram:

$$\omega_f = \sum_{b=1}^{10} |h_{b1} - h_{b2}|$$

Where h_{b1} and h_{b2} are the bars b from the normalized histograms of feature f, in classes 1 e 2, respectively.

Difference between Accumulated Histograms (DHA): Similar to DH, but it uses the accumulated histogram, given by:

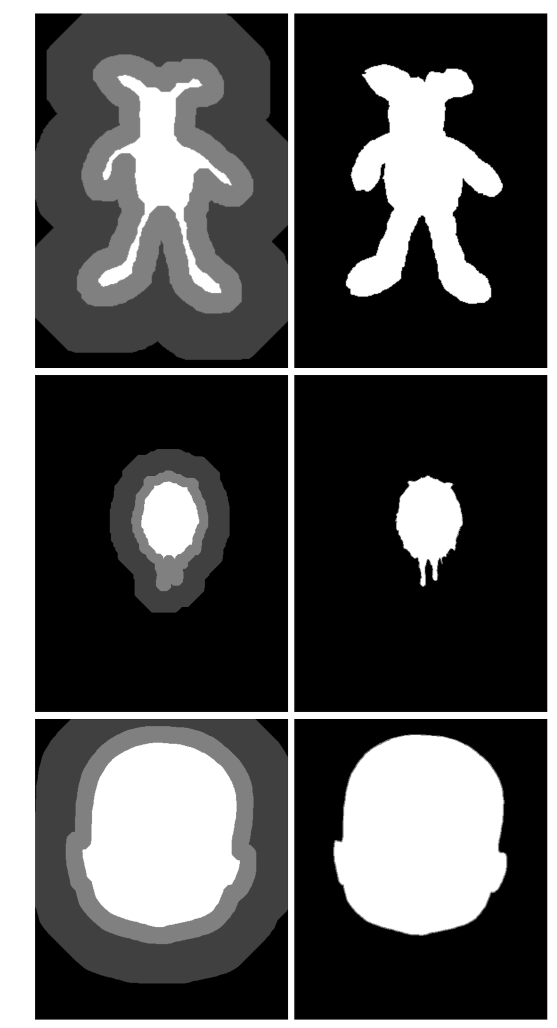
$$a_{bc} = \sum_{1}^{b} h_{bc}$$

where a_{bc} is the b bar of the accumulated histogram of class c, and h_{bc} is the b bar of the original histogram of class c. And then:

$$\omega_f = \sum_{b=1}^{10} |a_{b1} - a_{b2}|$$

Computer Simulations

Original images from the GrabCut dataset



Trimaps with seed regions (left), and ground truth images (right)

Conclusions

In this article, four methods to automatically define feature weights are proposed. They are used to define a graph in which the semi-supervised learning method known as particle competition and cooperation is applied. In that method, particles walk through a graph generated from the image to be segmented. The computer simulations with real-world images shown that the proposed methods raises the classification accuracy in the evaluated images, therefore it is a promising approach.

As future work, simulations with a bigger amount of images and features will be performed aiming to achieve more conclusive results. Moreover, the best methods may be improved to increase their efficacy.

BASE - Error: 1.98% | DMI - Error: 2.00% | DMI/SDPI - Error: 2.05% | DH - Error: 2.06% | **DHA - Error: 1.80%**

BASE - Error: 2.55% | DMI - Error: 2.64% | DMI/SDPI - Error: 2.06% | DH - Error: 2.09% | **DHA - Error: 1.90%**

BASE - Error: 2.18% | DMI - Error: 1.66% | **DMI/SDPI - Error: 1.27%** | DH - Error: 2.90% | DHA - Error: 2.68%

References

F. Breve, L. Zhao, M. Quiles, W. Pedrycz, and J. Liu, “Particle competition and cooperation in networks for semi-supervised learning,” *Knowledge and Data Engineering, IEEE Transactions on*, vol. 24, no. 9, pp. 1686 –1698, sept. 2012.

Y. Boykov and M.-P. Jolly, “Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images,” in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1, 2001, pp. 105–112 vol.1.