Complexity-Adaptive Distance Metric for Object Proposals Generation

Yao Xiao, Cewu Lu, Efstratios Tsougenis, Yongyi Lu, Chi-Keung Tang
The Hong Kong University of Science and Technology

Object Proposals have become indispensable for object detection, providing the latter with a set of image regions where objects are likely to occur. Currently, the mainstream methods [1, 4] partition the image into hundreds of superpixels, and then group them under certain criteria to form object proposals. Typically, the distance metric computes the difference between two superpixels in terms of an aggregate measure. While well suited to low-complexity superpixels grouping, it becomes less effective in high-complexity scenarios, Figure 1. In this paper, we propose a novel distance metric for grouping two superpixel sets that is adaptive to their complexity. Our distance metric combines a “low-complexity distance” and a “high-complexity distance” making it adaptive to different complexities.

Our system adopts the grouping scheme of [4]. Initially, a number of superpixels are generated and in each iteration, two superpixel sets with the smallest distance are merged. However, different from [4], low-level features (histogram) are not propagated during superpixel merging. Our complexity-adaptive distance metric is composed of several basic distance:

Color and texture feature distance is measured using L1 distance of color and texture histogram h_c and h_t then summed together:

$$d_{ct}(i,j) = \|h_c(i) - h_c(j)\| + \|h_t(i) - h_t(j)\| \tag{1}$$

Graph Distance. Our algorithm does not restrict grouping exclusively local neighboring superpixels. Instead we use graph distance to regularize the grouping process to prefer spatially close superpixels:

$$D_g(m,n) = \min\{d_g(i,j) | i \in S_m, j \in S_n\} \tag{2}$$

where S_m and S_n denote two superpixel sets.

Edge cost measures the edge responses along the common border of the segments. The edge cost for each neighboring segments is calculated by summing up the edge responses within the common border pixels and then normalized by the length of the common border. Denote the common border pixels set as $l_{i,j}$, then

$$D_e(m,n) = \begin{cases} \frac{\sum_{i\in S_m, j\in S_n} |l_{i,j}| d_e(i,j)}{\sum_{i\in S_m, j\in} |l_{i,j}|} & \text{if } \sum_{i\in S_m, j\in} l_{i,j} \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Consider two super-pixel set S_m and S_n, we define the nearest and farthest pairwise distance as,

$$D_{\min}(m,n) = \min\{d_{ct}(i,j) | i \in S_m, j \in S_n\} \tag{4}$$

$$D_{\max}(m,n) = \max\{d_{ct}(i,j) | i \in S_m, j \in S_n\} \tag{5}$$

$D_{\max}$ and $D_{\min}$ can be used to indicate respectively the low and high complexity distance of two given superpixels. A small $D_{\max}$ indicates that all the elements in the two sets are similar, meaning that they are of low complexity. Thus $D_{\max}$ suits for low-complexity region merging. In contrast, a small $D_{\min}$ means that the two sets are connected by at least two elements from the respective two superpixel sets. Therefore, $D_{\min}$ is a reasonable indicator for merging in high-complexity scenarios. Our low-complexity distance D_L and high-complexity distance D_H are respectively given by

$$D_L(m,n) = D_{\max}(m,n) + D_e(m,n) + D_g(m,n) \tag{6}$$

$$D_H(m,n) = D_{\min}(m,n) + bD_g(m,n) \tag{7}$$

where $0 < b < 1$ serves as a lower bound of spatial constraint. By combining the low and high complexity distance and complexity level factor $\rho_{n,m}$, we define the complexity-adaptive distance as

$$D_{total} = \rho_{m,n} D_L + (1 - \rho_{m,n}) D_H + \eta D_s \tag{8}$$

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.

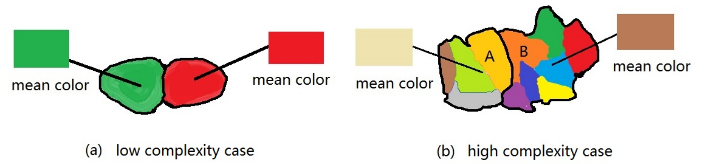

Figure 1: Color distance metrics. (a) The mean color is well behaved to delineate low-complexity superpixels, but (b) such aggregate measure fails to reflect the distance between two high-complexity superpixel sets.

	500 Candidates		1000 Candidates		2000 Candidates		
Methods	MABO	AUC	MABO	AUC	MABO	AUC	time
Selective Search [4]	0.771	0.517	0.799	0.562	0.812	0.585	5.4
MCG [2]	0.757	0.510	0.782	0.547	0.802	0.578	33.4
EdgeBox [5]	0.755	0.520	0.782	0.559	0.798	0.585	0.3
SPA [3]	0.736	0.487	0.776	0.545	0.800	0.583	16.7
CA1	0.768	0.517	0.809	0.585	0.836	0.631	6.3
CA2	**0.775**	**0.536**	**0.812**	**0.597**	**0.840**	**0.647**	22.6

Table 1: Comparison results of MABO and AUC using 500, 1000, and 2000 candidates; CA1 and CA2 respectively are the two settings used in our method.

Here D_s is defined as

$$D_s(m,n) = r_m + r_n \tag{9}$$

where r_m and r_n are the respective sizes of super-pixels m and n.

The function $\rho(m,n)$ indicates the complexity level of two sets m and n. Denote the element number of two sets respectively as T_m and T_n, and the total number of superpixels as T. We define

$$\alpha = -\log_2 \frac{T_m + T_n}{T}, \qquad \rho_{m,n} = (1 + \exp(-\frac{\alpha - \lambda}{\sigma}))^{-1} \tag{10}$$

where α represents the complexity level and λ controls the boundary of different complexity levels.

We compare our complexity-adaptive distance algorithm with state-of-the-art methods [2, 3, 4, 5]. For comparison, we use two different settings in our method. The first setting CA1 applies setting of 4 branches. The second setting CA2 adopts 12 branches to expand diversity. Two measurement criteria are used for overall performance evaluation: the Mean Average Best Overlap (MABO), introduced in [4], corresponds to the mean value of average best overlap considering all object categories, while the Area Under Curve (AUC) is the total area under the “recall versus IoU threshold" curve [5]. Table 1 tabulates the MABO and AUC results of all the tested methods using 500, 1000 and 2000 candidates where our algorithm’s efficiency is also compared to the-state-of-the-art methods. Although not equally efficient as [5], our method still produces high-quality results with acceptable execution time comparing to [2, 3].

Check http://www.cse.ust.hk/∼yxiaoab/cvpr2015/CADM.html for executables and full paper.

[1] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(11):2189–2202, 2012.

[2] Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping. CVPR, 2014.

[3] Pekka Rantalankila, Juho Kannala, and Esa Rahtu. Generating object segmentation proposals using global and local search. CVPR, 2014.

[4] Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

[5] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *Computer Vision–ECCV 2014*, pages 391–405. Springer, 2014.