An Adaptive Affinity Graph With Subspace Pursuit For Natural Image Segmentation

南京大學
NANJING UNIVERSITY

Yang Zhang, Huiming Zhang,
Yanwen Guo, Kai Lin, Jingwu He

誠樸雄偉 勵學敦行

CONTENTS

01 Introduction

02 Overview

03 Adaptive affinity graph

04 Experiments

05 Conclusion

1. Introduction

Image segmentation is a process of decomposing an image into independent regions.

Unsupervised methods receive much attention because they require no prior knowledge.

1. Introduction

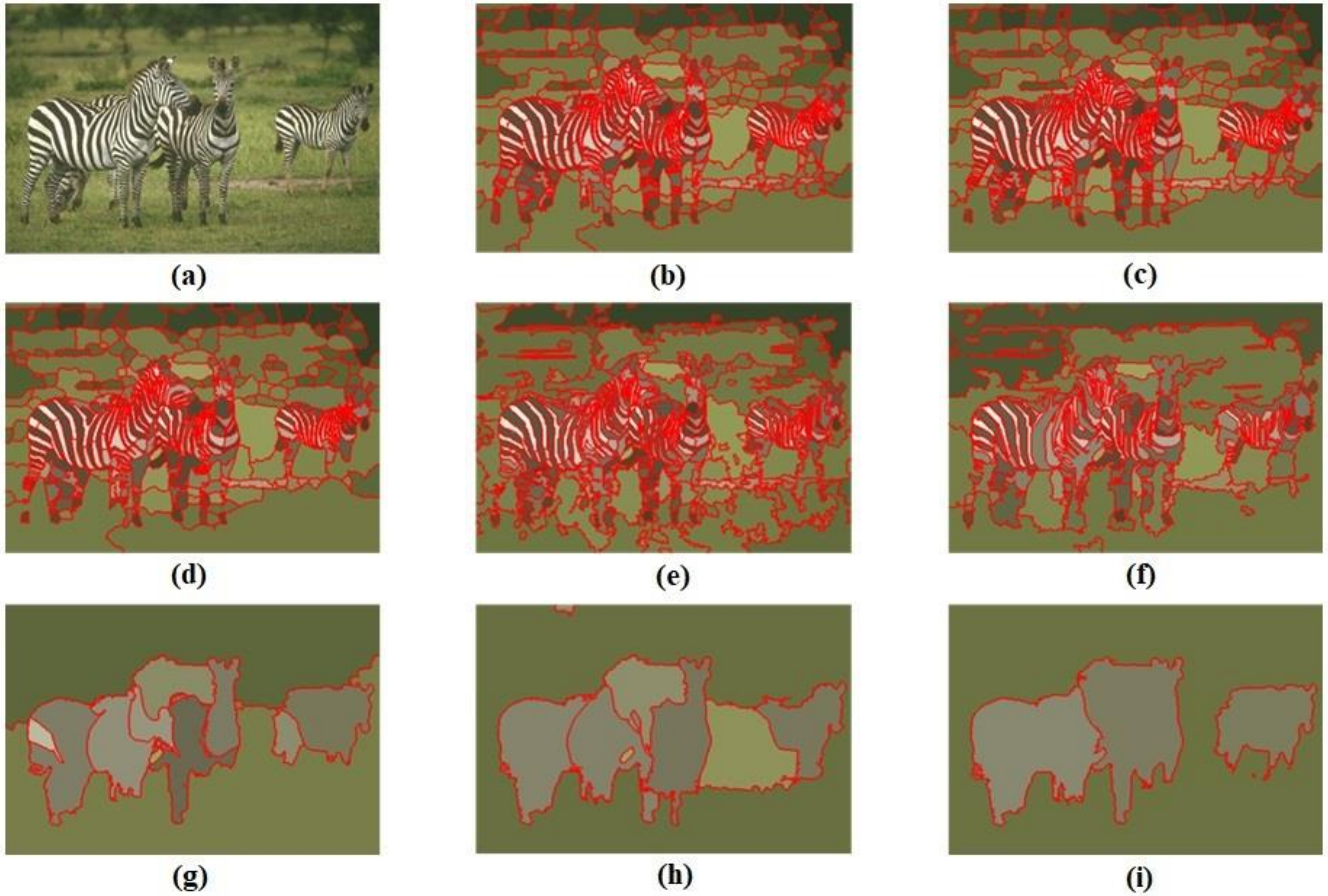


Fig. 1. Segmentation using superpixels. (a) Input image. (b-f) Superpixels generated by over-segmenting the image. (g-i) Segmentation results by adjacency-graph, GL-graph, and the proposed AASP-graph, respectively.

2. Overview

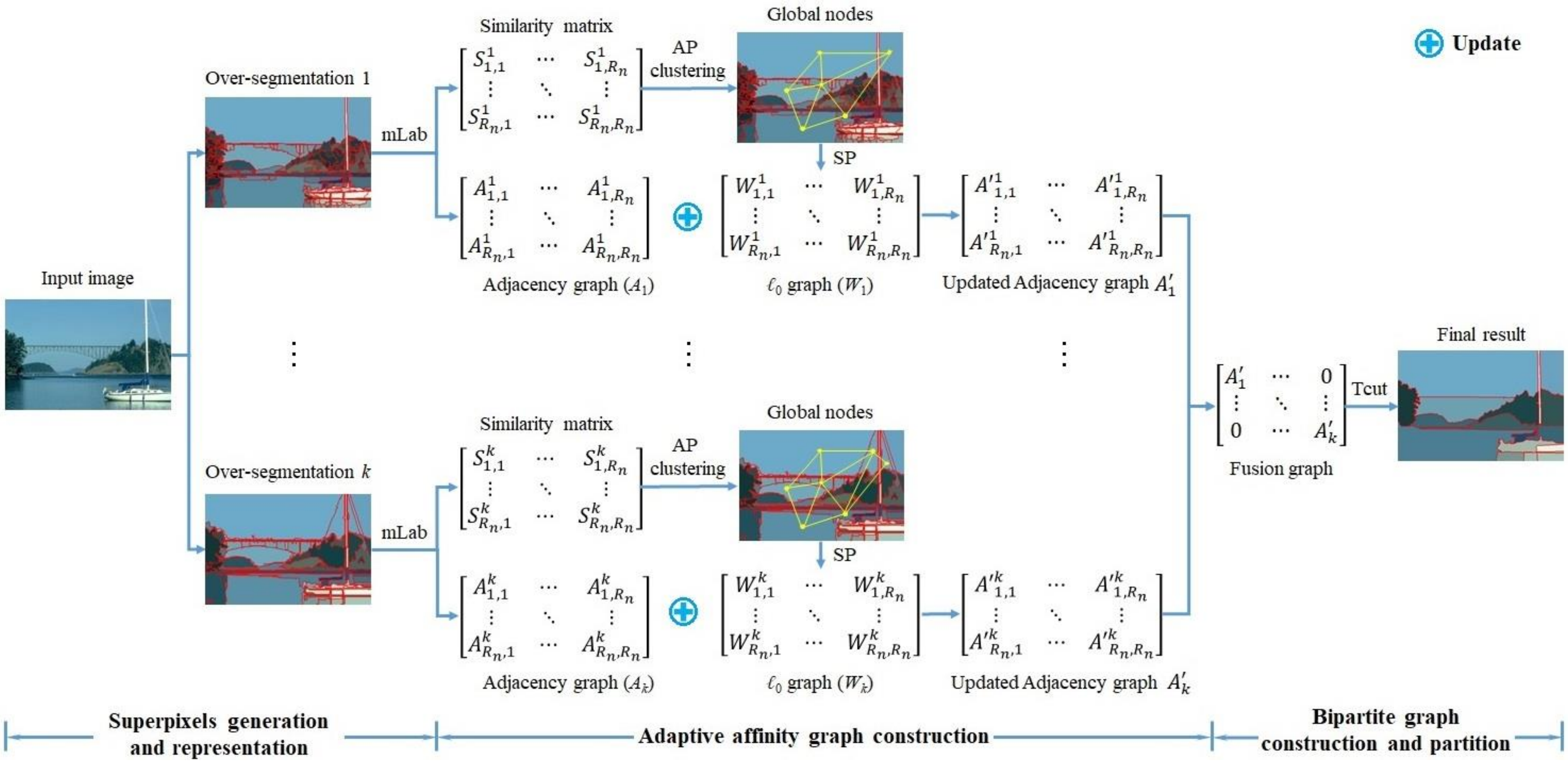
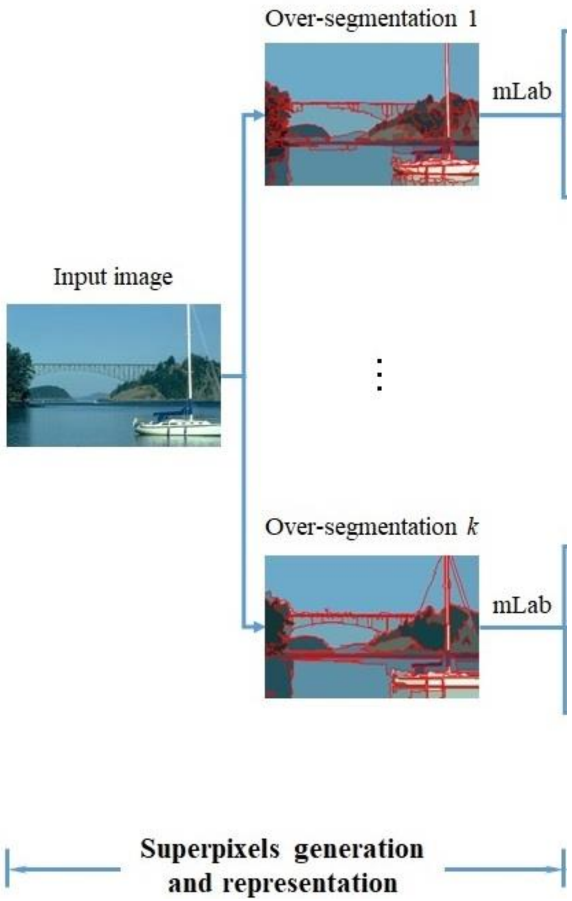
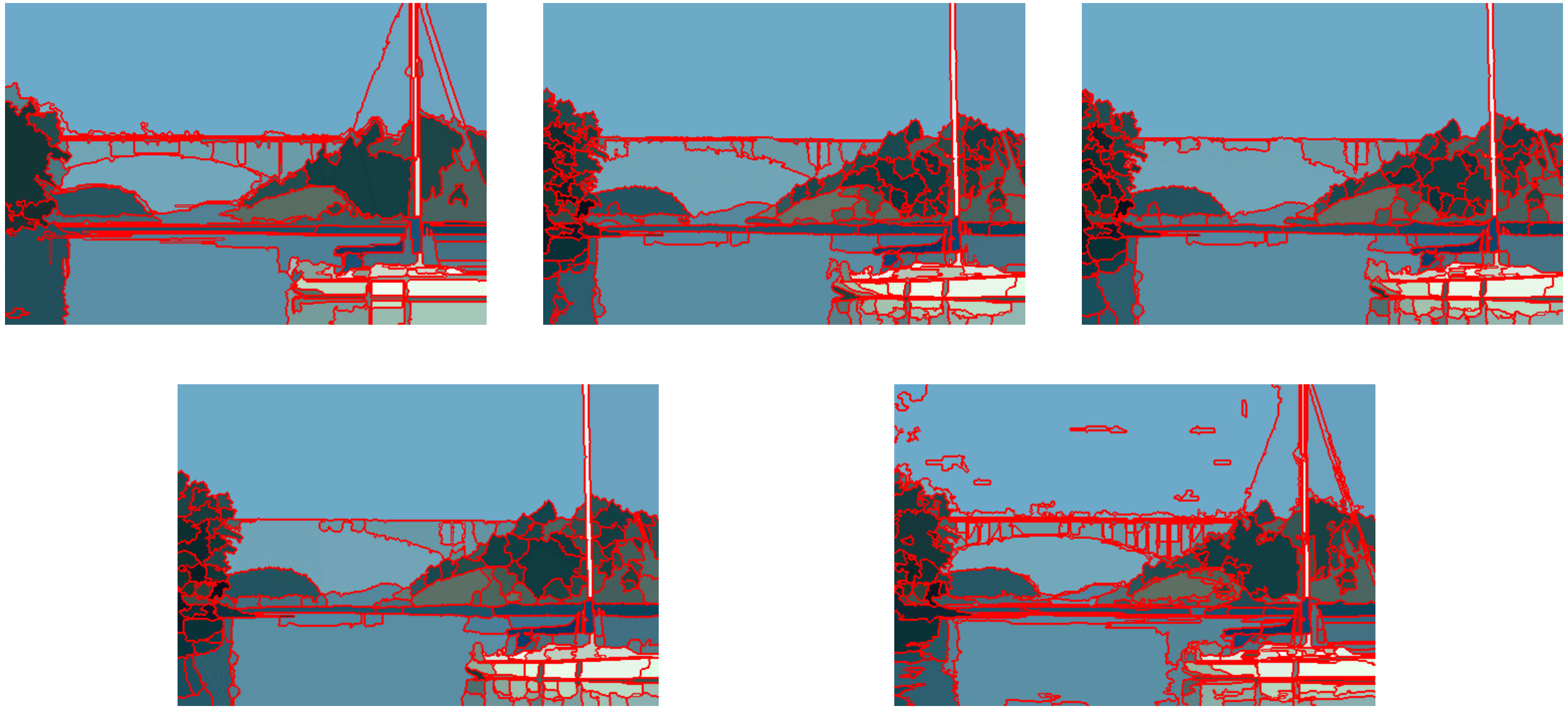


Fig. 2. The overall framework of image segmentation based on the proposed AASP-graph.

3. Adaptive affinity graph

3.1 Superpixels generation and representation

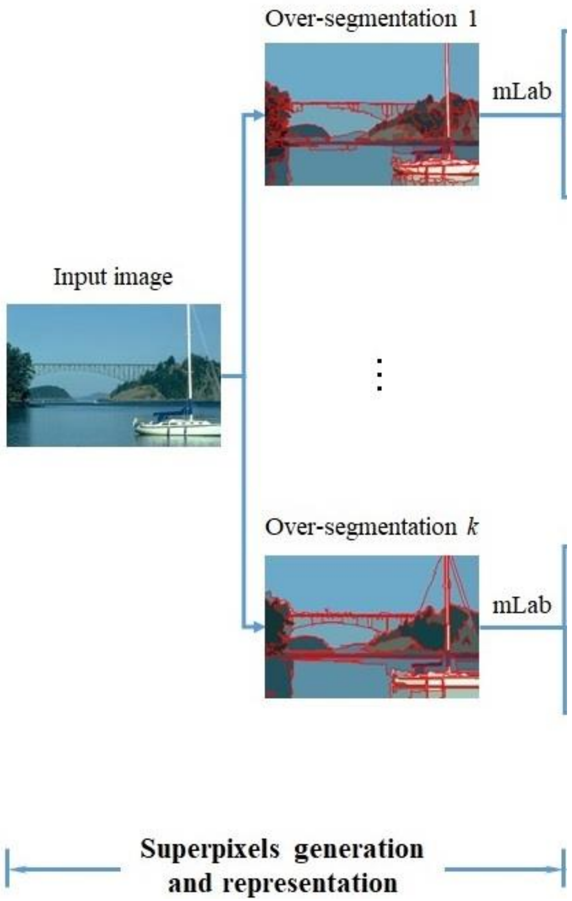




An input image is over-segmented into superpixels at *k* different scales (e.g., 5 scales) following the method in the SAS

3. Adaptive affinity graph

3.1 Superpixels generation and representation



Over-segmentation 1

mLab

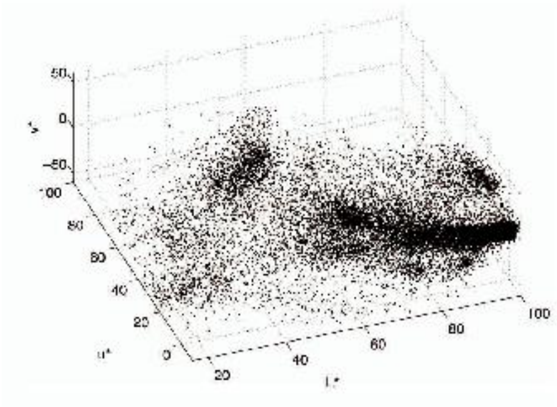
Input image

⋮

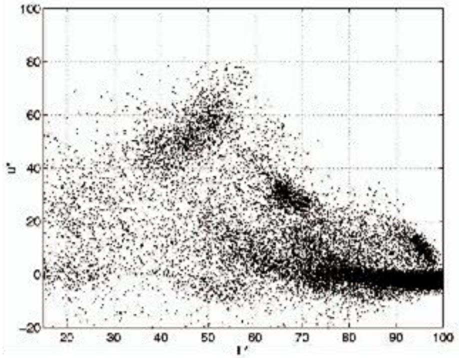
Over-segmentation *k*

mLab

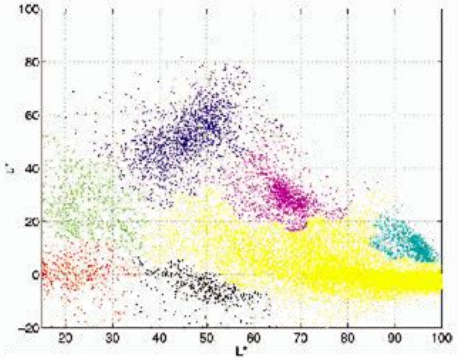
Superpixels generation and representation



CIE L*a*b* space representation



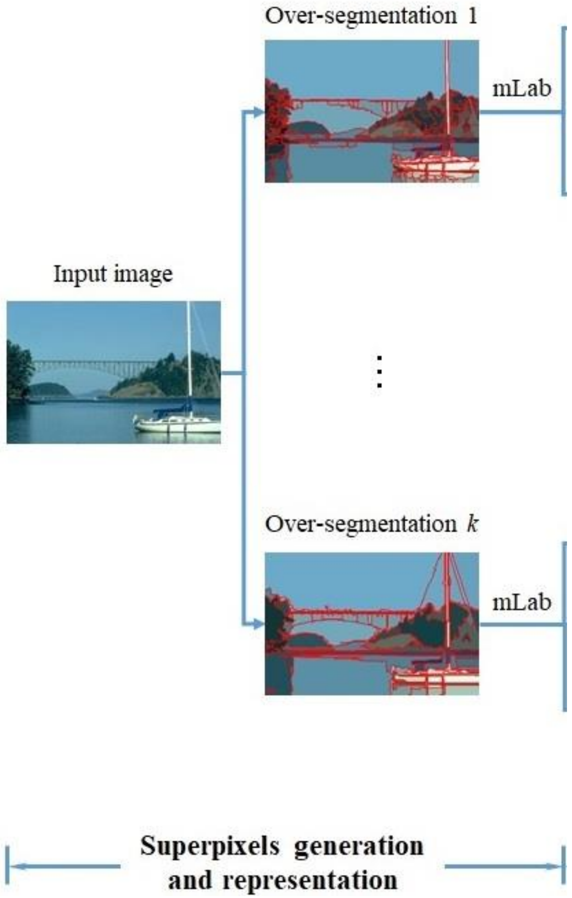
2D (L*a) representation



Final clustering result

3. Adaptive affinity graph

3.1 Superpixels generation and representation



Let $SI_k=\{si_i\}_{i=1}^{R_n}$ be the superpixels of an input image *I* at scale *k*, where *Rn* is the number of superpixels. Formally, it can be written as:

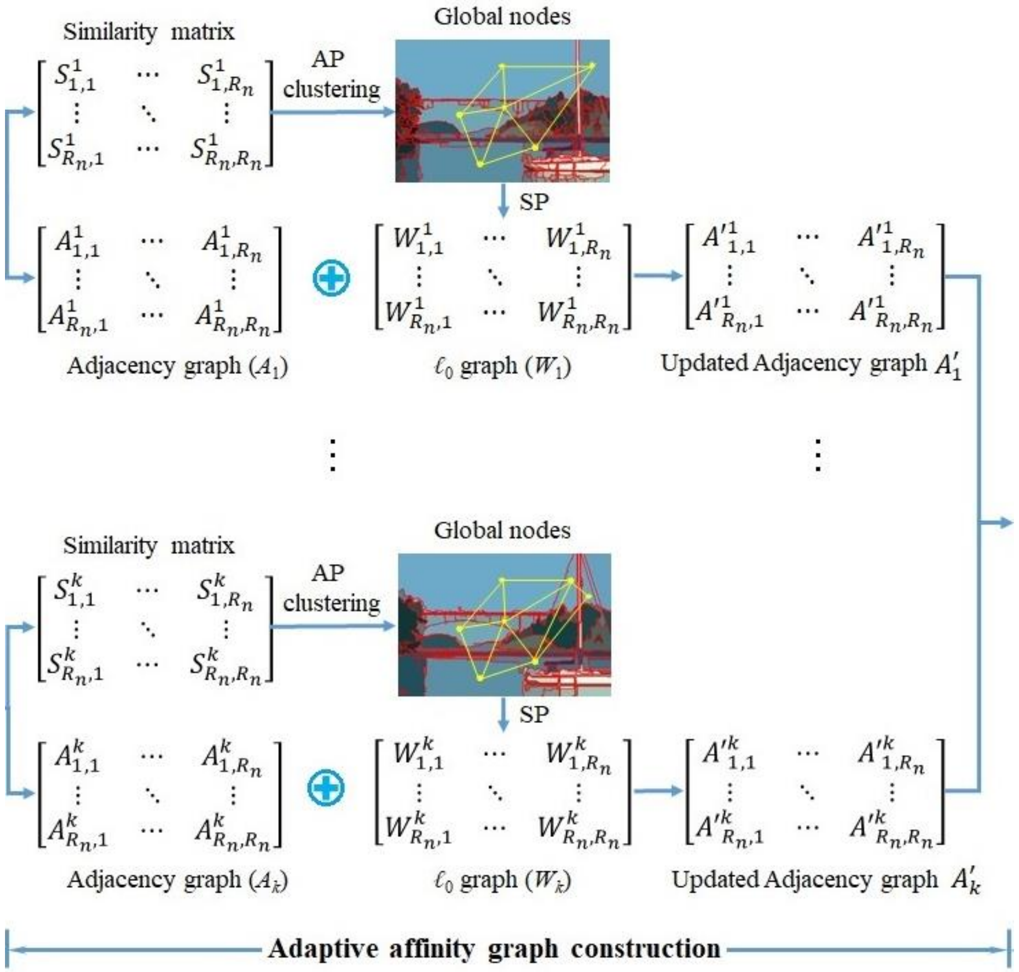
$$d_i = Dc_i, \quad c_{ii} = 0 \qquad (1)$$

where $c_i \in R^n$ is the sparse representation of superpixels, and $d_i \in R^m$ over the dictionary *D* which is a matrix representation of superpixels. The constraint $c_{ii} = 0$ prevents the self-representation of d_i.

3. Adaptive affinity graph

3.2 Graph construction

- ◆ Selecting global nodes
- ◆ Building a l_0-graph



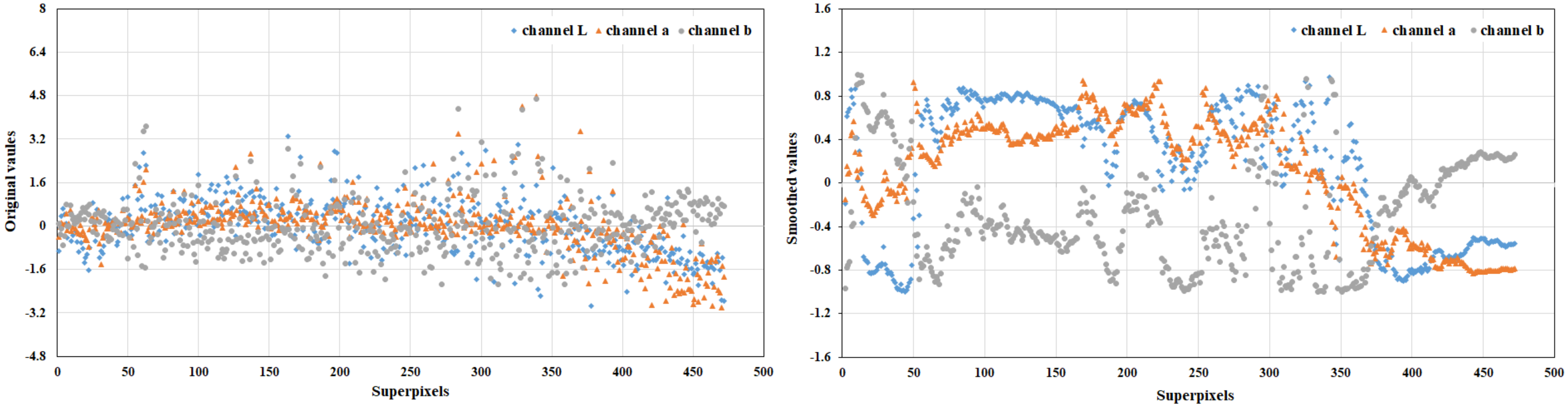
- ◆ Building an Adjacency-graph
- ◆ Building a fusion graph

3. Adaptive affinity graph

3.2 Graph construction

- Selecting global nodes

Kernel density estimation (KDE) to estimate color features (mLab) of natural images

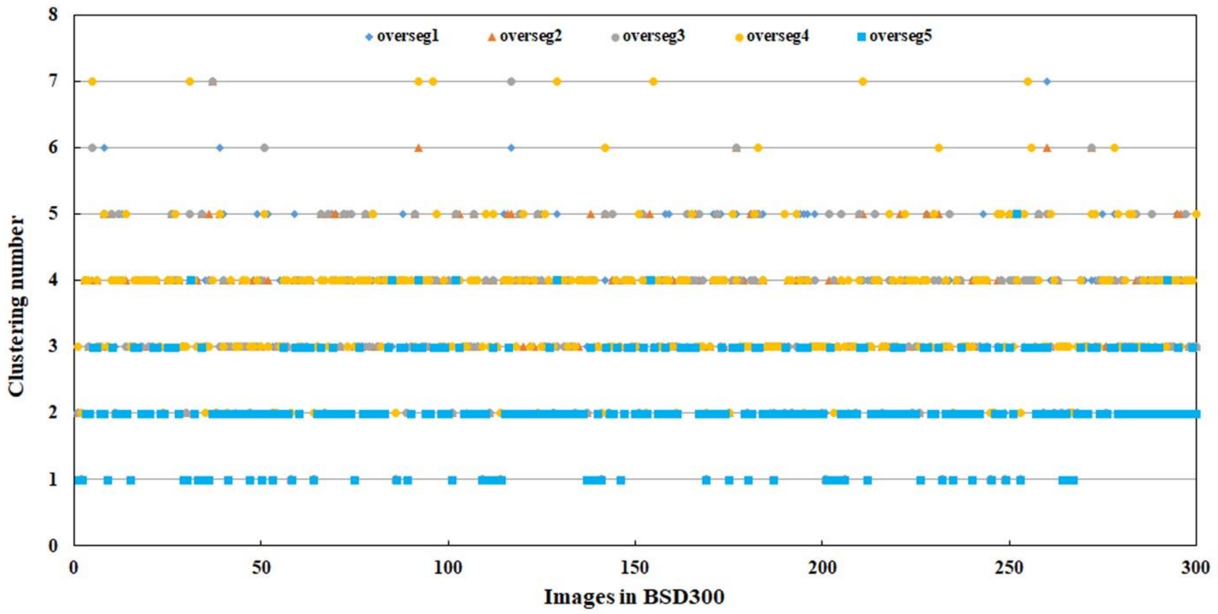


3. Adaptive affinity graph

3.2 Graph construction

- Selecting global nodes

Affinity propagation clustering (APC) with the mLab features of superpixels to find global nodes.



3. Adaptive affinity graph

3.2 Graph construction

- Building a l_0-graph

It is proved that the sparsest solution of Eq. (1) measured in the sense of l_0-norm is unique and conveys the most meaningful information of a signal. So, the sparse solution can be regarded as followings:

$$min||c_i||_0 \quad s.t. \quad d_i = Dc_i, \quad c_{ii} = 0 \tag{2}$$

where $||\cdot||_0$ represents the l_0-norm, which calculates the number of nonzero values in the vector.

3. Adaptive affinity graph

3.2 Graph construction

- Building a l_0-graph

Subspace pursuit (SP) is a simple and fast greedy method to seek an approximation of the sparsest solution.

$$\tilde{c}_i = argmin_{c_i} \{||f_i - M_W c_i||_2^2, ||c_i||_0 \leq LC, c_{ii} = 0\} \quad (3)$$

where the parameter LC is the maximal number of coefficients for each input mLab feature vector f_i, which controls the sparsity of the matrix-representation M_W.

3. Adaptive affinity graph

3.2 Graph construction

- Building a l_0-graph

The affinities coefficient $W_{i,j}$ of their l_0-graph W between superpixels s_{ii} and s_{ij} $(i, j = 1, 2, \ldots, R_n)$ can be computed as follows:

$$W_{i,j} = \begin{cases} 1 & if \quad i = j \\ 1 - (r_{i,j} + r_{j,i})/2 & if \quad i \neq j \end{cases} \tag{4}$$

where

$$r_{i,j} = ||f_i - c_{i,j} f_j||_2^2 \tag{5}$$

3. Adaptive affinity graph

3.2 Graph construction

- Building an Adjacency-graph

As for all superpixels at each scale, every superpixel is connected to its adjacent superpixels, denoted as **Adjacency-graph**. Let M_A be the matrix representation of all its adjacent neighbors, we attempt to represent f_i as a linear combination of elements in M_A. In practice, we solve the following optimization problem:

$$\tilde{c}_i^* = argmin_{c_i^*} ||f_i - M_A c_i^*||_2 \tag{6}$$

If a minimizer $\tilde{c}_i^*$ has been obtained, the affinities coefficient $A_{i,j}$ of the affinities A between a superpixel and its graph are calculated as in Eqs. (4) and (5).

3. Adaptive affinity graph

3.2 Graph construction

- Building a fusion graph

The l_0-graph W is used to replace the adjacency-graph A at global nodes for combining local graph and global graph to obtain the updated adjacency graph at a certain scale.

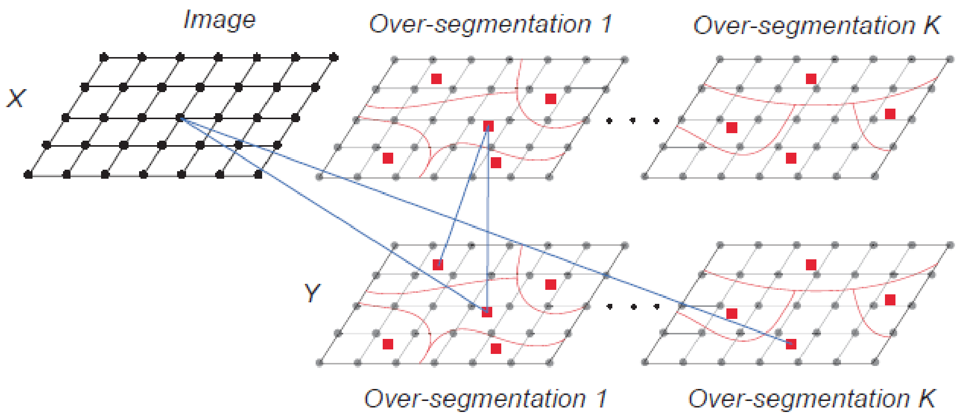
To fuse all scales of superpixels, we plug each scale affinity matrix A'_k corresponding to its AASP-graph into a block diagonal multiscale affinity matrix A_{SS} as follows:

$$A_{SS} = \begin{pmatrix} A'_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & A'_k \end{pmatrix} \tag{7}$$

3. Adaptive affinity graph

3.3 Bipartite graph construction and partition

To map the relationships between pixels and superpixels and enable propagation of grouping cues across superpixels at different scales, a **bipartite graph** [4] is built to describe the relationships of pixels to superpixels and superpixels to superpixels. The transfer cuts (Tcut) algorithm can be applied to partition a **bipartite graph** into different groups.



4. Experiments

4.1 Experimental setup

Database All experiments are carried out on the Berkeley Segmentation Database (BSD), which includes 300 images and the ground truth data.

Metrics To evaluate our method, there are four standard measurements:

- probabilistic rand index (PRI)
- variation of information (VoI)
- global consistency error (GCE)
- boundary displacement error (BDE)

4. Experiments

4.2 Results of different modules

Table 1 Quantitative comparison for different modules using in AASP-graph.

APC	IKDE	LKM	PRI ↑	VOI ↓	GCE ↓	BDE ↓
		✓	0.8442	1.6530	0.1741	14.6827
	✓	✓	0.8442	1.6532	0.1740	14.6807
✓		✓	0.8442	1.6529	0.1739	14.6761
✓	✓	✓	0.8446	1.6485	0.1737	14.6416

If the baseline does not select the APC, we use the *k*-means with 2 clustering number for fair comparison. The Lite *k*-means (LKM) is used in Tcut method, respectively.

4. Experiments

4.3 Comparison with state-of-the-art methods

Table 2 Quantitative results of the AASP-Graph with state-of-the-art approaches on BSDS300 dataset.

Methods	PRI ↑	VOI ↓	GCE ↓	BDE ↓	Methods	PRI ↑	VOI ↓	GCE ↓	BDE ↓
Ncut [7]	0.7242	2.9061	0.2232	17.15	Co-transduction[24]	0.8083	2.3644	0.2681	14.1972
MNCut [9]	0.7559	2.4701	0.1925	15.1	HO-CC[25]	0.8140	1.743	N/A	10.377
JSEG[20]	0.7756	2.3217	0.1989	14.4	LFPA[26]	0.8146	1.8545	0.1809	12.21
HIS-FEM[21]	0.7769	2.3067	0.2215	10.66	SAS[3]	0.8319	1.6849	0.1779	11.29
FusionTPG[22]	0.7771	3.3089	0.3654	13.2428	l_0-Graph[8]	0.8355	1.9935	0.2297	11.19
NTP[23]	0.7984	2.113	0.2171	13.58	GL-Graph[4]	0.8384	1.8012	0.1934	10.6633
H +R_Better[2]	0.8073	1.826	0.2079	12.16	**AASP-Graph**	**0.8446**	**1.6485**	**0.1737**	14.6416

4. Experiments

4.3 Comparison with state-of-the-art methods

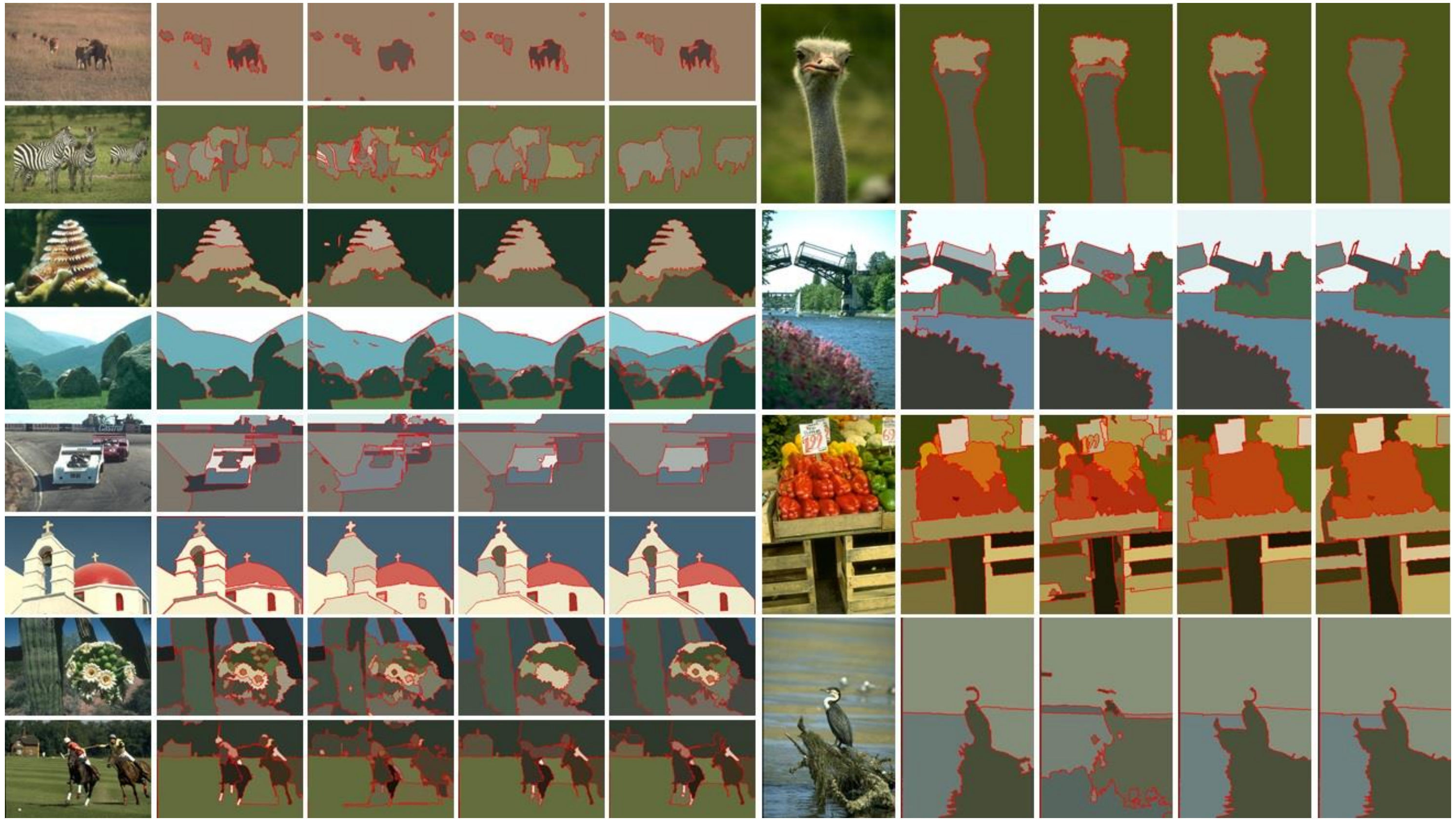


Fig. 3. Visual comparison obtained with the SAS, l_0-graph, GL-graph, and AASP-graph.

5. Conclusion

- An AASP-graph is proposed to obtain good results for natural image segmentation. The method uses superpixels of different scales as segmentation primitive.

- The improved APC is applied to adaptively select global sets, which are used to build l_0-graph with an SP method to update the adjacency-graph of all superpixels.

- Experimental results on a large number of images show the good performance and high efficiency of the proposed method. AASP-graph achieves competitive results, such as ranking the first in PRI, VoI, and GCE.

THANKS FOR YOUR TIME !

南京大學
NANJING UNIVERSITY

誠樸雄偉 勵學敦行