Evaluation of Scale-Aware Realignments of Hierarchical Image Segmentation

Milena M. Adão, Silvio Jamil Ferzoli Guimarães,
and Zenilton K. G. Patrocínio Jr.[(⊠)]

Computer Science Department, Pontifical Catholic University of Minas Gerais,
Belo Horizonte, MG, Brazil
milena.adao@gmail.com, {sjamil,zenilton}@pucminas.br

Abstract. A hierarchical image segmentation is a set of image segmentations at different detail levels. However, objects may appear at different scales due to their size or to their distance from the camera. One possible solution to cope with that is to realign the hierarchy such that every region containing an object is at the same level (or scale). In this work, we explore the use of regression to predict the best scale value for given region, which is then used to realign the entire hierarchy. Experimental results are presented for two different segmentation methods; along with an analysis of the adoption of different combination of mid-level features to describe regions.

Keywords: Hierarchical image segmentation ·
Alignment of hierarchy · Regression · Random forest · Neural network

1 Introduction

In the last two decades, a huge amount of multimedia data has been stored and made available through the Internet, which has attracted attention from research community to multimedia processing and analysis and, more specifically, to computer vision. Recent research results have pointed out that scale-awareness seems to be helpful in improving final results in many computer vision tasks [6,11–13]. Even though Deep Convolutional Neural Networks (DCNNs) have improved the performance of computer vision systems, they still face some challenges including the existence of objects at multiple scales [5].

The adoption of a hierarchical approach which incorporates information from multiple scales is an alternative to deal with this issue. And, since segmentation is one of the first step involved in almost every computer vision task, the use of a hierarchical segmentation method has helped improving results for different tasks [3,15,16]. Specifically in image context, a hierarchical image segmentation

is a set of image segmentations at different detail levels in which the segmentations at coarser detail levels can be produced from simple merges of regions from segmentations at finer detail levels [10].

Hierarchical segmentation methods [2,10] have been successfully used. These methods create a hierarchy of partitions that can be represented as a tree (Fig. 1). Moreover, the final results can be represented as an Ultrametric Contour Map (UCM) [1], which allows to obtain a particular segmentation (at a given observation scale) through a simple thresholding (Fig. 2). The hierarchies are typically computed by an unsupervised process that is susceptible to under-segmentation at coarse levels and over-segmentation at fine levels. Thus, objects (or even parts of the same object) may appear at different scales due to their size or to their distance from the camera. To cope with that one may explore the use of non-horizontal cuts [8,9]. Another possible solution is to flatten the hierarchy into a single non-trivial (or non-horizontal) segmentation, such as in [17].
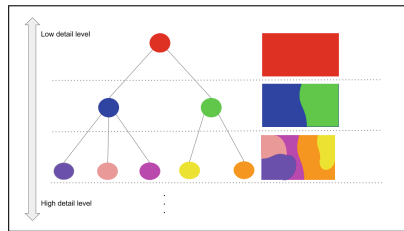

Fig. 1. Example of obtained result by a hierarchical image segmentation method.

In [7], the authors proposed to modify the final result of a hierarchical algorithm by improving its alignment, *i.e.*, by trying to modify the depth of the regions in the tree to better couple depth and scale and, therefore, putting (almost) all objects (and their parts) at the same level (or scale). To do that, they first train a regressor to predict the scale of regions using mid-level features. Then, they create a set of regions that better balance between over and under-segmentation, named *anchor slice*. Finally, the original hierarchy is realigned using the *anchor slice*, *i.e.*, adjusting the hierarchy such that every region in the *anchor slice* is at the same level (or scale) – see Fig. 3.

Fig. 2. Example of obtained result by a hierarchical image segmentation method: (left) original image; and (right) Ultrametric Contour Map (UCM).
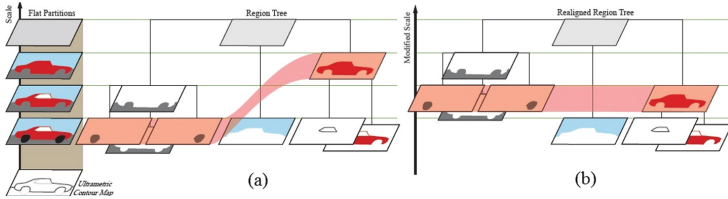
Fig. 3. Example from [7] showing the realignment of a hierarchical image segmentation: (a) original hierarchy; and (b) realigned hierarchy.

In this work, we explore the use of regression to predict the best scale value for given region, which is then used to realign the entire hierarchy. In our assessment, we used two different hierarchical image segmentation methods: **gPb-owt-ucm** [2] and **hGB** [10]. The main contributions of this work are: (i) impact analysis of a learning strategy on prediction of scale values for distinct hierarchical methods; and (ii) evaluation of the use of different combination of mid-level features to describe regions.

The paper is organized as follows. Section 2 describes the hierarchical segmentation methods used. In Sect. 3, we present the realignment approach for hierarchies. Section 4 presents experimental results. Finally, we draw some conclusions in Sect. 5.

2 Hierarchical Image Segmentation

There is a rich literature of hierarchical image segmentation. But here, in the following, we only describe the hierarchical methods used in this work: **gPb-owt-ucm** [2] and **hGB** [10].

Method gPb-owt-ucm. A widely-used state-of-the-art hierarchical segmentation method proposed in [2]. Discriminative features are learned for local boundary detection and spectral clustering is applied to it for boundary globalization. Afterwards, a hierarchical segmentation is built by exploring the information on the contour signal. The authors of [2] proposed a variant of the watershed transform, named Oriented Watershed Transform (OWT), for producing a set of initial regions from contour detection output. Then, an UCM is generated from the boundaries of these initial regions.

Method hGB. According to [10], a hierarchical segmentation should be able to maintain spatial and neighborhood information between segments, even when changing the scale. Thanks to that, one can compute the hierarchical observation scales for any graph, in which the adjacent graph regions are evaluated depending on the order of their merging in the fusion tree. The core of **hGB** [10] is the identification of the smallest scale value that can be used to merge the largest region to another one while guaranteeing that the internal differences of these merged regions are greater than the value calculated for smaller scales.
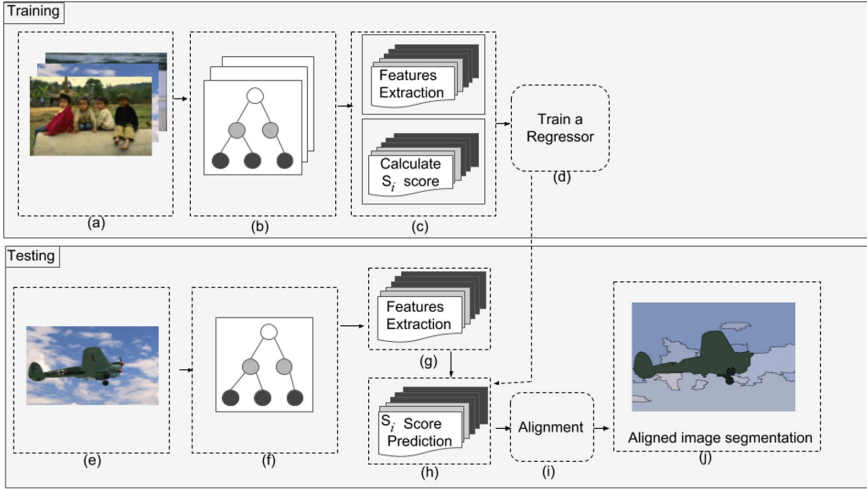
Fig. 4. Realign approach.

Starting with simple regions representing single image pixels, **hGB** is able to produce a hierarchy of partitions for the entire image. It has been successfully applied not only to image segmentation [10], but also to several other tasks, such as: video segmentation [16]; video summarization [3]; and video cosegmentation [15].

3   Realignment Approach

The realignment approach used in this work is illustrated in Fig. 4. First, a set of training images (Fig. 4a) is used to produce the corresponding set of hierarchies (Fig. 4b) using **gPb-owt-ucm** [2] or **hGB** [10]. Then, all regions belonging to training hierarchies are described with a set of features (see Sect. 3.1) and have their S_i scores (Eq. 1) calculated (Fig. 4c). These data is used to train a regression method (Fig. 4d). During testing, for each test image (Fig. 4e) the corresponding hierarchy is produced (Fig. 4f); and features are computed for every region (Fig. 4g). These features are used with the trained regressor to predict the best scale value (using S_i scores) for each region (Fig. 4h). Finally, the predicted scores/scales are used to realign (see Sect. 3.3) the original hierarchy (Fig. 4i), which is used to produce a final segmentation (Fig. 4j).

3.1   Features Extraction

The mid-level features were extracted from all regions of each hierarchy. Similar to [7], the chosen features were the following:

– **Graph partition properties:** cut, ratio cut, normalized cut, unbalanced normalized cut;

– **Region properties:** area, perimeter, bounding box size, major and minor axis lengths of the equivalent ellipse, eccentricity, orientation, convex area, Euler number;
– **Gestalt properties:** inter- and intra-region texton similarity, inter- and intra-region brightness similarity, inter- and intra-region contour energy, curvilinear continuity, convexity.

We have also explored features to encode **color properties**, such as color mean, and color histogram. Color-related features are calculated for each channel (in RGB color space) and histograms are generated with 04 bins per channel (in RGB color space). More details about these features could be found in [4].

3.2 Training and Predicting

A regression forest (with 100 trees) was trained using all regions R_i from each hierarchy. For each region R_i, its corresponding *ground-truth* region G_i is identified and used to calculate S_i score by Eq. 1.

$$S_i = \frac{|G_i| - |R_i|}{\max(|G_i|, |R_i|)} \qquad (1)$$

in which $|R_i|$ and $|G_i|$ represent the size of region R_i and of its corresponding *ground-truth* region G_i, respectively. Similar to [7], the most-overlapping human-annotated segment is taken as the corresponding *ground-truth*. When S_i score is a negative value, it indicates that the region R_i is under-segmented, while a positive value stands for over-segmented (and, 0 for properly segmented). In order to describe the region properties, all features described before were extracted. In this step, regions whose area is less than 50 pixels were excluded.

After training the regression approach, they are used to make prediction. For that, the same set of features were extracted from all regions belonging to each hierarchy generated for the test subset and used to predict the best scale value.

3.3 Alignment of Hierarchical Image Segmentation

Following [7], each node belonging to a hierarchy should be labeled as: -1, 0, and $+1$ indicating under-, properly- and over-segmented, respectively. This could be done solving a problem that penalizes two cases: (i) segments in the group of under-segmented with positive scores; and (ii) segments in the group of over-segmented with negative scores. The resulting problem can be solved via dynamic programming. The *anchor slice* consists of regions labeled as 0.

After that, a local linear transform (the same used in [7]) is performed on the UCM corresponding to the hierarchy, and the *anchor slice* is aligned to scale value of 0.5 (for the convenience and later use).

Table 1. Average results for regression with random forest.

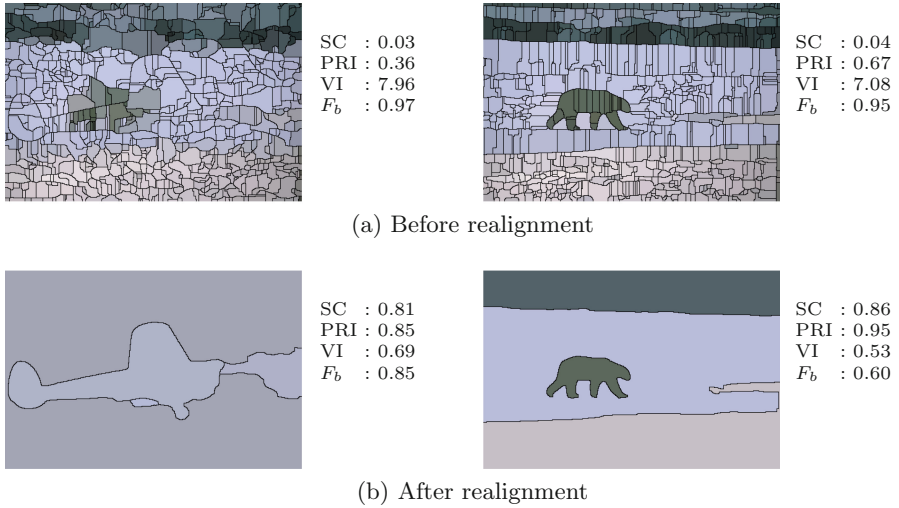|	SC ↑		PRI ↑		VI ↓		F_b ↑	
	ODS	OIS	ODS	OIS	ODS	OIS	ODS	OIS
gPb-owt-ucm w/no-alignment	0.59	0.65	0.83	0.86	1.69	1.48	0.73	0.76
gPb-owt-ucm w/alignment (c+gr+s)	0.58	0.64	0.82	0.85	1.75	1.49	0.69	0.76
gPb-owt-ucm w/alignment (c+s)	0.55	0.61	0.79	0.83	1.81	1.58	0.69	0.74
hGB w/no-alignment	0.43	0.62	0.74	0.80	2.34	1.88	0.49	0.50
hGB w/alignment (c+ge+gr+s)	0.38	0.53	0.75	0.81	2.45	1.89	0.50	0.50
hGB w/alignment (ge+gr+s)	0.39	0.53	0.74	0.81	2.45	1.89	0.50	0.50
hGB w/alignment (c+s)	0.43	0.62	0.74	0.80	2.34	1.88	0.49	0.50

4   Experimental Results

In order to evaluate the realignment of hierarchies generated by **gPb-owt-ucm** [2] and **hGB** [10], we use the BSDS500 dataset [2], which includes 500 images (200 for training, 100 for validation, and 200 for testing). As segmentation evaluation measures, we adopted: (i) Segmentation Covering (SC); (ii) Probabilistic Rand Index (PRI); (iii) Variation of Information (VI); and (iv) F-measure for boundary (F_b); all four computed at Optimal Dataset Scale (ODS) and Optimal Image Scale (OIS) – see [14] for a review of these measures and scales. Note that for all measures a large value is better, except for VI.
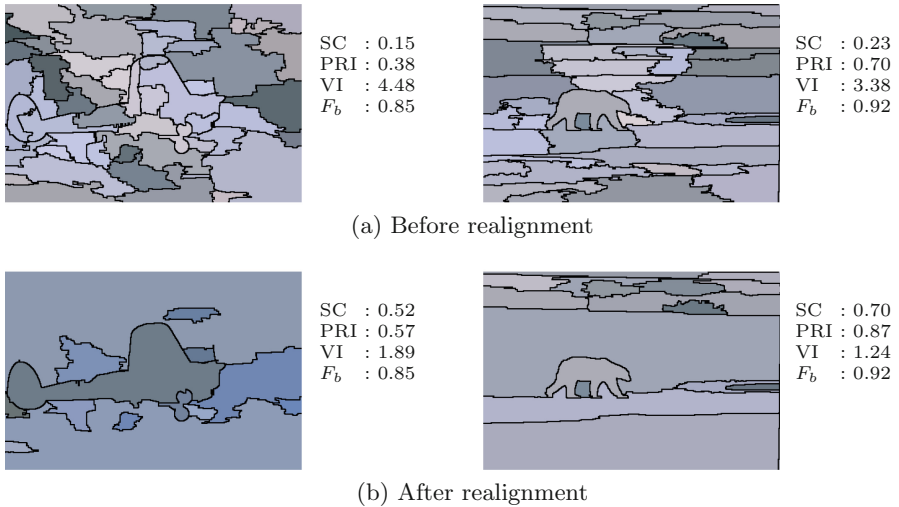
Average results obtained for regression made with random forest are shown in Table 1. In that table, 'c' stands form color based features (such as color mean and color histogram), 's' is used to represent region shape features (such as area, perimeters, etc.), 'gr' stands for graph features (such as cut, ratio cut, etc.), and 'ge' is used to represent gestalt features (such as texton similarities, brightness similarities, etc.). For **gPb-owt-ucm**, the realignment exhibits a improvement in average VI score when a set of color, graph and shape features is used, while for **hGB**, there is no difference in any metric (on average). But a closer look in some specific final results may help us understanding better those results.

In Figs. 5 and 6, we illustrate some examples in which the realignment of original hierarchies produces quite interesting results when the scale is set to 0.5 (which corresponds to the *anchor slice*). For **gPb-owt-ucm**, in Fig. 5(a) the results are showed without the realignment, while Fig. 5(b) illustrates the realigned results obtained. One can easily see the improvements related to SC, PRI, and VI measures. But this has some negative impact on F_b scores obtained by **gPb-owt-ucm**, specially for second example – see Fig. 5(b).

Similarly, for **hGB**, in Fig. 6(a) the results are showed without the realignment, while Fig. 6(b) illustrates the realigned results obtained. Again, it is easy to verify the improvements related to SC, PRI, and VI measures. But the main difference is that F_b scores obtained by **hGB** are not affected in this case. That could explain the decrease in average F_b scores in the realigned results for **gPb-owt-ucm** shown at Table 1.

SC : 0.03
PRI : 0.36
VI   : 7.96
F_b : 0.97

SC : 0.04
PRI : 0.67
VI   : 7.08
F_b : 0.95

(a) Before realignment

SC : 0.81
PRI : 0.85
VI   : 0.69
F_b : 0.85

SC : 0.86
PRI : 0.95
VI   : 0.53
F_b : 0.60

(b) After realignment

Fig. 5. Examples of segmentations results before and after the realignments by using **gPb-owt-ucm** for computing the hierarchy.

SC : 0.15
PRI : 0.38
VI   : 4.48
F_b : 0.85

SC : 0.23
PRI : 0.70
VI   : 3.38
F_b : 0.92

(a) Before realignment

SC : 0.52
PRI : 0.57
VI   : 1.89
F_b : 0.85

SC : 0.70
PRI : 0.87
VI   : 1.24
F_b : 0.92

(b) After realignment

Fig. 6. Examples of segmentations results before and after the realignments by using **hGB** for computing the hierarchy.

5   Conclusion

In this work, we explored the use of regression to predict the best scale value for given region, which is then used to realign the entire hierarchy.

Experimental results are presented for two different segmentation methods; along with an analysis of the adoption of different combination of mid-level features to describe regions.

For **gPb-owt-ucm**, the realignment exhibits a improvement in average VI score when a set of color, graph and shape features is used, while for **hGB**, there is no difference in any metric (on average). But a closer look in some specific final results seems to indicate that the realignment of hierarchies generated by **gPb-owt-ucm** has some negative impact on F_b scores, while this is not observed for hierarchies produced by **hGB**.

In order to improve and better understand our results, further works involve training of different regression approaches and adoption of other segmentation methods; and also the application of our proposal to another datasets.

References

1. Arbelaez, P.: Boundary extraction in natural images using ultrametric contour maps. In: Conference on Computer Vision and Pattern Recognition Workshop, CVPRW 2006, pp. 182–182. IEEE (2006)
2. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE TPAMI **33**(5), 898–916 (2011)
3. Belo, L.S., Caetano, C.A., Patrocínio Jr., Z.K.G., Guimaães, S.J.F.: Summarizing video sequence using a graph-based hierarchical approach. Neurocomputing **173**, 1001–1016 (2016)
4. Carreira, J., Sminchisescu, C.: Constrained parametric min-cuts for automatic object segmentation. In: IEEE CVPR 2010, pp. 3241–3248. IEEE (2010)
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE TPAMI **40**(4), 834–848 (2018)
6. Chen, L.C., Yang, Y., Wang, J., Xu, W., Yuille, A.L.: Attention to scale: scale-aware semantic image segmentation. In: IEEE CVPR 2016, pp. 3640–3649 (2016)
7. Chen, Y., Dai, D., Pont-Tuset, J., Van Gool, L.: Scale-aware alignment of hierarchical image segmentation. In: IEEE CVPR 2016, pp. 364–372 (2016)
8. Cousty, J., Najman, L.: Morphological floodings and optimal cuts in hierarchies. In: IEEE ICIP 2014, pp. 4462–4466 (2014)
9. Guiges, L., Cocquerez, J., Men, H.L.: Scale-sets image analysis. IJCV **68**(3), 289–317 (2006)
10. Guimarães, S.J.F., Kenmochi, Y., Cousty Jr., J., Z.K.G.P., Najman, L.: Hierarchizing graph-based image segmentation algorithms relying on region dissimilarity: the case of the Felzenszwalb-Huttenlocher method. Math. Morphol. Theory Appl. **2**, 55–75 (2017)
11. Hao, Z., Liu, Y., Qin, H., Yan, J., Li, X., Hu, X.: Scale-aware face detection. In: IEEE CVPR 2017, pp. 1913–1922 (2017)
12. Jie, Z., Liang, X., Feng, J., Lu, W.F., Tay, E.H.F., Yan, S.: Scale-aware pixelwise object proposal networks. IEEE TIP **25**(10), 4525–4539 (2016)
13. Li, J., Liang, X., Shen, S., Xu, T., Feng, J., Yan, S.: Scale-aware fast R-CNN for pedestrian detection. IEEE Trans. Multimed. **20**(4), 985–996 (2018)
14. Pont-Tuset, J., Marques, F.: Supervised evaluation of image segmentation and object proposal techniques. IEEE TPAMI **38**(7), 1465–1478 (2016)
15. Rodrigues, F., et al.: Graph-based hierarchical video cosegmentation. In: Battiato, S., Gallo, G., Schettini, R., Stanco, F. (eds.) ICIAP 2017. LNCS, vol. 10484, pp. 15–26. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68560-1_2

16. Souza, K.J.F., Araújo, A.A., Patrocínio Jr., Z.K.G., Guimarães, S.J.F.: Graph-based hierarchical video segmentation based on a simple dissimilarity measure. PRL **47**, 85–92 (2014)
17. Xu, C., Whitt, S., Corso, J.J.: Flattening supervoxel hierarchies by the uniform entropy slice. In: IEEE ICCV 2013, pp. 2240–2247 (2013)