**Multilevel Object Closure Detection by Superpixels**

CSC2523 Research Project, Winter 2018

Han Meng, ID: 1003748171; Hanwen Liang, ID: 1003868291

**1. Abstract (Hanwen)**

The closure of any object in an image sets the scope for the pixels for the object, and those pixels within the closure can be used for further application, for example neural network models for recognition tasks. Finding a good closure for an object thus can be thought as a preprocessing to focus the recognition on a specific object instead of finding a main object among a few possible objects. While the definition of objects can be vague and abstract, if we treat objects in different level, we will be able to have priority in detecting objects in an image. For example, an image may contain a car, a tree, and a person. This might be the most outer level of objects. Then, in the car, there are the front windshield, wheels, front hood, headlights, and so on. These are the objects in the next level constrained by the closure of the car, and they can also be detected by their closures. In our project, we develop a hierarchical structure that can separate the image depending on the need of the user, for instance how specific for objects and in what range.

However, there are many challenges in obtaining contour closure, linking together a set of fragmented contours into a cycle that separates an object from its background. Early perceptual grouping researchers [61] identified a set of non-accidental contour relations, such as symmetry, parallelism, collinearity, co-curvilinearity, etc., that can be used to link together causally related contours. However, the space of possible closures is still overwhelming, particularly when one allows larger and larger boundary gaps in a closure. Finding an optimal solution is intractable without somehow reducing the complexity of the problem.

In our project, we used Alex’s method that reformulates the problem of searching for cycles of contours as the problem of searching for a subset of superpixels whose border has strong contour support in the contour image; the assumption we make here is that those salient contours that define the boundary of the object (our target closure) will align well with superpixel boundaries. In this method, parametric maxflow is used to yield the top k solutions. In every layer of the hierarchy, the user can choose one of the result that best fits his demand and do the deeper layer.

**2. Introduction**

**2.1 Background and Literature Review (Hanwen)**

Image segmentation is the process of decomposing an image into multiple segments based on pixel similarity, edges or other prior information. It is essential for many computer vision tasks, including object recognition, detection and tracking. 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. Therefore, the segmentations at finer levels are nested with respect to those at coarser levels.

Over the past few decades, hundreds or even thousands of segmentation algorithms have been proposed, trying to produce segmentation results that are similar to what human eyes perceive. However, image segmentation is still considered as a challenging problem due to the high dimensionality of visual data and complex nature of its content. Computing contour closures of an image provides a meaningful approach for image segmentation, which links together a set of fragmented contours into a cycle that separates an object from its background. It also leads to another complex problem: How to find the proper circles among intractable number of candidates existing in the contours.

Early researches identified a set of nonaccidental contour relations, such as classical Gestalt cues of parallelism and symmetry, that can be used to link together causally related contours. Yet the space of possible closures is still overwhelming, particularly when one allows larger boundary gaps in a closure. One possible taxonomy for categorizing related work is based on the nature of the prior information used to constrain the grouping process. Since it’s unclear how to make methods based on prior information scale up to large databases, our project focus on methods that make no assumptions about scene content and incorporate low-, mid-, high-level shape priors, as exemplified by Ren et al. [7]. One challenge faced by these merging shape priors methods is the complexity of pairwise contour grouping to detect symmetry-related contour pairs. At this point, Levinshtein et al. [16] proposed to overcome this computational complexity limitation by constraining the symmetric parts to be collections of superpixels.

Superpixels provide a convenient primitive from which to compute local image features. They capture redundancy in the image and greatly reduce the complexity of subsequent image processing tasks. They have proved increasingly useful for applications such as depth estimation, image segmentation, skeletonization, body model estimation and object location. For superpixels to be useful they must be fast, easy to use and produce high quality segmentations. Some of the outstanding superpixel-generation methods involve SLIC, Ncuts and Turbo.

Our project draws on this idea of grouping superpixels by three methods above but focuses on the more generic cue of closure. Further down the spectrum of prior knowledge are methods based on weaker shape priors than parallelism and symmetry. For example, Jacobs [17] uses convexity as well as gap to extract closed contours by grouping straight line segments. A less restrictive measure is that of compactness, which can be attained by normalizing the gap by area. Finally, our project uses the most general methods proposed by Alex et al. [core] that compute closure using only weak shape priors, such as continuity and proximity. This method uses a notion of boundary gap, which is a measure of missing image edges along the closed contour.

Elder and Zucker [18] model the probability of a connection between two adjacent contour fragments and find contour cycles using a shortest path algorithm. Wang et al. [19] optimize a measure of average gap using the ratio cut approach.

All the above methods suffer from the high complexity of choosing the right closure from a sea of contour fragments. To cope with this complexity, Alex’s method tried to minimize closure cost using ratio cuts. His manipulation of superpixels provides greater scope not only for gap computation, but also for incorporation of internal appearance-based affinity. His methods also provide a set of optimal solutions that capture closures at multiple scale, also give users an opportunity to choose the ideal one for building next layer of hierarchical structure.

**2.2 Project Overview and Structure (Han)**

This research project was done based on the work and code from [Core ref.]. Basically, there are a few critical steps involved, and they are pB edges, superpixels, parametric maxflow grouping, and recursive object closure detection. The following is their brief descriptions, and their details will be in the next sections.

The first step is to compute the pB boundaries for the image. The pB boundary detector was first proposed by Martin et. al. [ref] to combine brightness, color, and texture as local features to detect boundaries for natural images. The result of the pB boundary detector is used to help use determine if the contour as enough edge support. The more edge support we have, the more confident we are about the selected closure.

The second step is to compute superpixels, which are a higher level than the fundamental pixels in an image. Since we intend to use a group of adjacent superpixels to represent an object and use its boundary to approximate its closure, generating superpixels and labelling them for an image are necessary for this project. There are a few different algorithms to compute superpixels, and we chose SLIC, NCuts, and turbo pixels as the main candidates. We also intend to compare the results by using these three different superpixel algorithms.

The third step is to group multiple adjacent superpixels. According to [Core Ref], the grouping of multiple superpixels together to get their closure as an approximation of the boundary of an object can be formulated as an optimization problem. This problem can then be reformulated as an energy flow problem and can be solved by parametric maxflow.

The last step is to wrap the above three steps as a function and use it to detect the objects on the same level and the next level. For the same level object detection, we intend to search all valid objects with the criteria of reaching a certain size. For the next level objects, we intend to search all objects within any object in the parent level. By recursively repeating these two steps, we can find a tree-structure, which can represent the relationships among all possible objects in an image.

**3. Methodology**

**3.1 pB Edge Detector (Han)**

**3.2 Superpixels (Hanwen)**

**3.3 Parametric Maxflow (Hanwen)**

**3.4 Recursive Extract-Objects Algorithm (Han)**

**4. Results**

**4.1 Pictorial results (Han)**

**4.2 Analysis of results (Han)**

**4.3 Comparison Among Different Superpixel Algorithms (Hanwen)**

**5. Limit and Future Work (Han)**

**6. Conclusion (Han)**

**7. Acknowledge (Han)**

**8. References (Hanwen)**

**9. Appendix (Han)**