Pattern Recognition Letters 43 (2014) 71–80

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

An evaluation of the compactness of superpixels

Alexander Schick [a,*], Mika Fischer [b], Rainer Stiefelhagen [b]

[a] Fraunhofer Institute of Optronics, System Technologies and Image Exploitation, Fraunhoferstr. 1, 76131 Karlsruhe, Germany
[b] Karlsruhe Institute of Technology, Institute of Anthropomatics, Vincenz-Priessnitz-Str. 3, 76131 Karlsruhe, Germany

A R T I C L E I N F O

Article history:
Available online 14 October 2013

Communicated by Kim Boyer

Keywords:
Superpixel segmentation
Compactness
Superpixel lattices

A B S T R A C T

Superpixel segmentation is the oversegmentation of an image into a connected set of homogeneous regions. Depending on the algorithm, superpixels have specific properties. One property that almost all authors claim for their superpixels is compactness. However, the compactness of superpixels has not yet been measured and the implications of compactness have not been investigated for superpixels. As our first contribution, we propose a metric to measure the compactness of superpixels. We further discuss implications of compactness and demonstrate the benefits of compact superpixels with an example application. Most importantly, we show that there is a negative correlation between compactness and boundary recall. A second desirable property for superpixel segmentations is conforming to a lattice. This regular structure, similar to the pixel grid of an image, can then be used for more efficient algorithms. As our second contribution, we propose an algorithm that offers both a transparent and easy-to-use compactness control with an optional lattice guarantee. We show in our evaluation with six benchmark algorithms, that the proposed algorithm outperforms the state-of-the-art.

1. Introduction

Image segmentation is a fundamental task in computer vision and many applications rely on it as a preprocessing step. Superpixel segmentation belongs to the class of oversegmentation algorithms and the term superpixel was introduced by Ren and Malik (2003).

A superpixel is defined as a homogeneous image region that aligns well with object boundaries. This allows to represent an image with only a couple of hundred segments instead of tens of thousands of pixels. This reduction of input complexity makes superpixels particularly useful for a wide range of application domains, for example image segmentation (Achanta et al., 2010; Schick and Stiefelhagen, 2011; Veksler et al., 2010), object recognition (Achanta et al., 2010), object localization (Fulkerson et al., 2009), labeling tasks (Kohli et al., 2009), motion segmentation (Ayvaci and Soatto, 2009), foreground segmentation (Schick et al., 2012), tracking (Wang et al., 2011), and pose estimation (Mori, 2005; Mori et al., 2004), to name just a few.

A *compact* superpixel has a regular shape with smooth boundaries and many authors agree that compactness is desirable for superpixels (Achanta et al., 2010; Levinshtein et al., 2009; Liu et al., 2011; Moore et al., 2010; Veksler et al., 2010; Zeng et al., 2011; Zhang et al., 2011; Perbet and Maki, 2011). However, the compactness of superpixel segmentations has not yet been systematically measured and evaluated. We are the first to measure the compactness of superpixels and investigate its implications.

This work is an extended version of Schick et al. (2012). The main additional contribution is an extension of the superpixel segmentation in Schick et al. (2012) that guarantees that it conforms to a lattice. Further, we extended the experimental section with more discussions about compactness including a correlation, convergence, and lattice stability analysis.

The remainder of this paper is organized as follows. Related work is discussed in Section 2. The compactness metric is presented in Section 3. Section 4 introduces the segmentation algorithm followed by the proposed lattice constraints in Section 5. The evaluation is presented in Section 6 with results in Section 7. Section 8 demonstrates the benefits of compactness with an example application before the conclusion in Section 9.

2. Related work

Superpixels have received increasing attention in the last years and there is a wide range of superpixel segmentation algorithms. These algorithms differ in how they solve the segmentation task which results in different properties regarding runtime, segmentation quality, and superpixel shape. Graph-based segmentation algorithms were proposed by Shi and Malik (2000), a normalized cut approach by Malik et al. (2001), and a graph cut approach by Veksler et al. (2010). Zhang et al. (2011) proposed a superpixel

* Corresponding author. Address: Fraunhofer IOSB, Fraunhoferstr. 1, 76131 Karlsruhe, Germany. Tel.: +49 721 6091 620.
E-mail address: alexander.schick@iosb.fraunhofer.de (A. Schick).
URL: http://iosb.fraunhofer.de (A. Schick).

0167-8655/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2013.09.013

segmentation based on pseudo-boolean optimization as an improvement to the method proposed by Veksler et al. (2010). Levinshtein et al. (2009) grow superpixels based on geometric flows. Variations of Levinshtein et al. (2009) were presented by Zeng et al. (2011) where they proposed a structure-sensitive superpixel segmentation based on the geodesic distance and by Xiang et al. (2010) where they incorporate eigen-images. Superpixel segmentation based on random walks was presented by Perbet and Maki (2011) with discussions of superpixel shape; an extended version (Perbet et al., 2012) also discussed compactness. However, they focus more on spatial homogeneity while we are concerned with the compactness aspects and their implications. Superpixels are not limited to color images, but can also be applied to depth images (Weikersdorfer et al., 2012). Liu et al. (2011) proposed a novel objective function based on entropy rate and a balancing term to compute superpixels in a graph-based framework. Up until now, they achieved the best results on established metrics. However, the superpixels are often quite irregular even though they claim compactness. We propose an algorithm outperforming the entropy rate superpixels even while maintaining higher compactness.

Most of the currently existing superpixel algorithms destroy the regular pixel lattice. Two algorithms that compute a superpixel lattice were proposed by Moore et al. (2010) and Moore et al. (2008). The first one is a greedy algorithm with added topological constraints (Moore et al., 2008). The second one applies graph cuts to graphs that have additional edges to enforce the lattice structure (Moore et al., 2010).

SLIC, proposed by Achanta et al. (2010), iteratively clusters superpixels based on k-means. SLIC is accurate and quite fast with extensions for supervoxels (Lucchi et al., 2012) and GPU implementations (Ren and Reid, 2011). We propose a modified version of SLIC that achieves better results while additionally maintaining a lattice structure.

Our contributions to the state-of-the-art are twofold. First, compactness is considered important (Achanta et al., 2010; Levinshtein et al., 2009; Liu et al., 2011; Moore et al., 2010; Veksler et al., 2010; Zeng et al., 2011; Zhang et al., 2011; Perbet and Maki, 2011) but has not yet been thoroughly investigated. The proposed compactness metric will therefore help researchers to better evaluate their algorithms and to investigate the effects of compactness on their specific applications. Second, the proposed algorithm offers a transparent compactness control thus making it a good choice for systematic evaluations. The additional, but optional, lattice guarantee further strengthens its usefulness.

3. Superpixel compactness

The larger the area of a shape for a given boundary length, the higher is its compactness. The same holds for superpixels and there seems to be an intuitive understanding that compactness is indeed a desirable property (Achanta et al., 2010; Levinshtein et al., 2009; Liu et al., 2011; Moore et al., 2010; Veksler et al., 2010; Zeng et al., 2011; Zhang et al., 2011; Perbet and Maki, 2011). We will now discuss why compactness is indeed advantageous before explaining the compactness metric.

3.1. Why compactness?

Besides a more appealing visual appearance, there are various reasons why compact superpixels are generally advantageous: (a) Compact superpixels better capture spatially coherent information, (b) due to their regular shape it is more likely that their neighborhood relationships are less complex, (c) their representation size is smaller due to more regular and shorter boundaries, and (d) algorithms operating directly on the boundaries have a decreased input complexity.

Superpixels are meant as building blocks and the scale of interest is given by their size. It can be seen as overfitting when boundaries of non-compact superpixels become highly irregular to capture every minor detail in the image.

3.2. Compactness metric

In mathematics, measuring the compactness of a two-dimensional shape is a well-known task. Related to this task is the isoperimetric problem: for a given boundary length, find the two-dimensional shape with the maximal area (Plya, 1990). The solution to this problem and the most compact shape is the circle.

The isoperimetric quotient is related to this problem and is a measure for compactness. It relates the area of a shape to that of a circle with the same boundary length. The isoperimetric quotient for the circle is 1 and decreases for less compact shapes.

For a shape, e.g. superpixel S, let A_S be its area and L_S its perimeter. It follows that the radius of a circle with the same perimeter as the superpixel is $r = \frac{L_S}{2\pi}$ and the area of this circle is $A_C = \pi\left(\frac{L_S}{2\pi}\right)^2$. The isoperimetric quotient then is

$$Q_S = \frac{A_S}{A_C} = \frac{4\pi A_S}{L_S^2}. \tag{1}$$

We propose a metric based on the isoperimetric quotient to measure the compactness (CO) of a superpixel segmentation. For a given superpixel segmentation $\mathfrak{S}$ of image I, we compute the isoperimetric quotient for each superpixel $S \in \mathfrak{S}$ and normalize it by its size $\frac{|S|}{|I|}$. Then, the compactness of the segmentation is

$$CO = \sum_{S \in \mathfrak{S}} Q_S \cdot \frac{|S|}{|I|}. \tag{2}$$

The normalization is important because there could be segmentations with very small, but perfectly compact superpixels and only few large segments. When normalizing with the number of superpixels, the large number of small, but compact superpixels would dominate the result. Due to the normalization, each superpixel contributes to the compactness metric according to its size.

Orientation changes in the boundaries increase the overall boundary length while contributing very little to the area. This negatively affects the compactness which is in accordance with the discussion in Section 3.1.

4. Superpixel segmentation

In this section, we propose a modification of SLIC (Achanta et al., 2010) that computes more accurate superpixels with a transparent control of their compactness. (This algorithm was also presented in Schick et al. (2012).) SLIC is based on an iterative k-means clustering and computes superpixels utilizing both a distance in color space as well as Euclidean space. While the k-means algorithm leads to very accurate clusters, it does not guarantee that the clusters remain connected which is essential for superpixels. Therefore, a postprocessing step is required that reconnects superpixels that have been ripped apart, thereby bypassing the two distance terms.

We solve this problem by not working on all image pixels simultaneously, but only on boundary pixels. We can thereby guarantee that the superpixels stay connected during the segmentation. We will now explain the algorithm in detail.

The image segmentation is initialized with a rectangular grid of superpixels. Their initial size is the first parameter of the algorithm. Then, the superpixel boundaries are iteratively refined until the segmentation converges. The refinement works on boundary pixels only. For each boundary pixel p, a score is computed that