Shape Based Outlier Detection in SLIC Superpixels

Gerhard Klassen
Institute of Computer Science
Heinrich Heine University Düsseldorf
D-40225 Düsseldorf, Germany
Gerhard.Klassen@uni-duesseldorf.de

Michael Singhof
Institute of Computer Science
Heinrich Heine University Düsseldorf
D-40225 Düsseldorf, Germany
singhof@cs.uni-duesseldorf.de

ABSTRACT

Various computer vision algorithms make use of superpixels in order to segment images. Usually superpixels oversegment images and thereby are merged or erased by evaluating certain criteria. Some criteria might be a feature vector calculated from the grouped pixels, random graph walks or the size of superpixels. In this paper we introduce a novel approach based on the shape and neighborhood of superpixels. We regard the shape of superpixels as a feature for image boundaries. Our method consists of three phases: First the reference phase in which a reference superpixel is identified. This is followed by the comparison phase in which superpixels are erased with the help of a similarity condition. Finally the neighborhood analysis removes remaining outliers. Our experiments are based on a slight modification of SLIC superpixels and make use of shape matching based on Hu invariants which is also implemented in the OpenCV framework. We tested our approach on artificial images and photographs. The resulting superpixels lower the image complexity and can be regarded as an indicator for image boundaries.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; I.4.6 [**Image Processing And Computer Vision**]: Segmentation - Edge and feature detectionFeature Measurement; H.2.8 [**Database management**]: Database Applications - Data mining

Keywords

Outlier Detection, Image Segmentation, Superpixel, SLIC, Shape, Shape Matching, OpenCV, Object Recognition

1. INTRODUCTION

Superpixels are image patches which group pixels with similar properties. They have some major advantages over analyzing the underlying pixel-grid as noted in [16], [7] and [20]. First of all they reduce the complexity of images from billions of pixels to thousands, hundreds or less superpixels. Further the grouping of uniform pixels with similiar properties make every superpixel perceptually meaningful. Finally, superpixels conserve the structures of the original image.

The popularity of superpixels in computer vision applications leads to the proposal of many different superpixel algorithms. It has shown, that some approaches produce better results for specific applications than others [18]. However beside many other properties, different approaches can lead to different superpixel shapes as one can see in Figure 1. Regardless of the chosen technique, superpixels always tend to align to image boundaries and thereby conserve the structures of the original image. This property also influences the shape of the superpixel.

There are many different approaches which use this property as their basis for image segmentation. Such algorithms are proposed in [22] and in [11]. However most of the algorithms use a computed feature vector but none of the algorithms deals with the information which is given by the resulting shape of a superpixel.

The alignment of superpixels to boundaries in images allows human perception to reconstruct shapes of the original image without knowing them. In Figure 1 one can observe that plane surfaces are represented with regular formed superpixels while superpixels located next to borders are perceptual malformed.

Our goal is the definition of an image-depended measure in order to identify and filter regular formed superpixels. We consider these to hold too little information in order to contribute to the general understanding of the image.

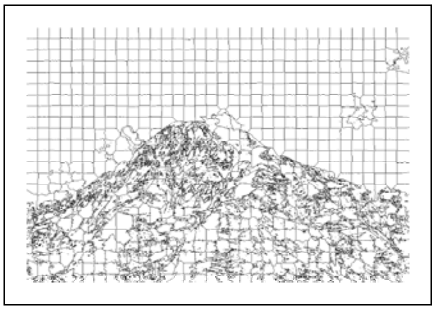
The remainder of the paper is structured as follows: In the next section we emphasize the novelty of our technique. Then our approach which consists of three phases is presented. This section is followed by the application of our method to artificial images and photographs. Finally we conclude our work in the last section.

2. RELATED WORK

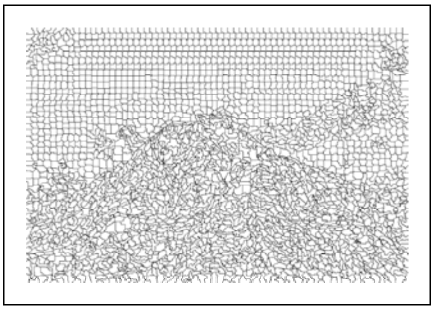
Since the introduction of the term superpixel in the year 2000 there were many different proposals of superpixel algorithms as one can read in [5], [13], [19], [4], [6], [12] and [3]. As demonstrated in Figure 3 and Figure 1, superpixels oversegment images while conserving the structures of the image. This is why these approaches are often used for further image segmentation algorithms [21], [17]. However, these approaches base on different feature sets. Therefore further image processing is highly dependend on the chosen

29th GI-Workshop on Foundations of Databases (Grundlagen von Datenbanken), 30.05.2017 - 02.06.2017, Blankenburg/Harz, Germany.

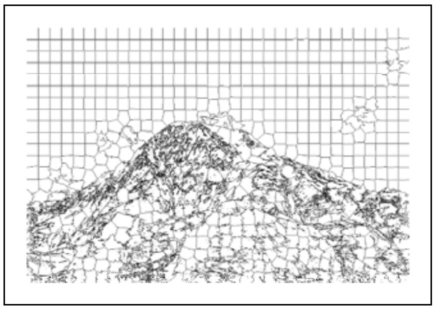
(a) original



(b) SLIC [19]



(c) SEEDS [4]



(d) LSC [13]

Figure 1: Comparison of superpixel algorithms

superpixel algorithm. In [10] the superpixel size is used as a feature for image segmentation. Another segmentation algorithm working with superpixels is proposed in [15]. It is based on markov random walks to achieve an image segmentation. In comparison to that [11] uses the entropy rate of random graph walks. After all, the novelty of our approach is the analysis of the superpixel shape as until now there is no segmentation algorithm which regards the shape of superpixels as a feature. A famous algorithm that groups pixels is the simple linear iterative clustering (SLIC) introduced in 2010 [19]. It is especially simple and implements a local k-means clustering to generate the superpixels. Our approach is based on a slight modification of SLIC superpixels.

3. METHOD

In this section we present our approach which consists of three main phases. The first phase, the reference phase, targets the selection of an appropriate reference superpixel sp_{ref}. This selection depends on the processed image and has a major impact on the next phases.
When an appropriate superpixel is selected the next phase is started. The goal of the comparison phase is the classification of superpixels as well-formed or malformed. This is done with the help of a similarity function which can be chosen freely. Depending on the targeted result superpixels of one certain class are filtered.
Finally there is a last phase. The neighborhood analysis reviews the neighborhood of the remaining superpixels. It includes a filter process which depends on the previous analysis.
Our method uses a slight modification of the SLIC Superpixel algorithm. As one can see in Figure 3(b) the original shape of the superpixels tends to be a hexagon. We achieve this shape by altering the starting points of the clusters as proposed in [8].
In the context of the SLIC algorithm there are k centroids chosen which result in k superpixels. Let $id : (x, y) \mapsto \{1, ..., k\}, x \in (1, width)$ and $y \in (1, height)$ be the function which returns an unique identifier for every of these centroids where $height$ and $width$ is the image height and width respectively. As a result of the clustering algorithm every pixel $p = (x, y)$ of the image is labeled with the id of the corresponding centroid

$$p^d = (p, d) \text{ with } d = id(centroid(p)) \qquad (1)$$

where $centroid(p)$ returns the centroid p has been assigned to, and $d \in 1, ..., k$. The resulting cluster is called a superpixel sp and consists of a set of pixels which are assigned to the same centroid

$$sp^d = \{p_1^d, ..., p_m^d\} \text{ with } p_l^d = (x, y),\ l \in [1, m] \qquad (2)$$

According to the label of the pixels of a certain centroid the same label is assigned to the superpixel and denoted as sp^d. The shape of a superpixel is defined by a subset sx^d of its pixels. Therefore the neighborhood of every pixel in a superpixel is analyzed.

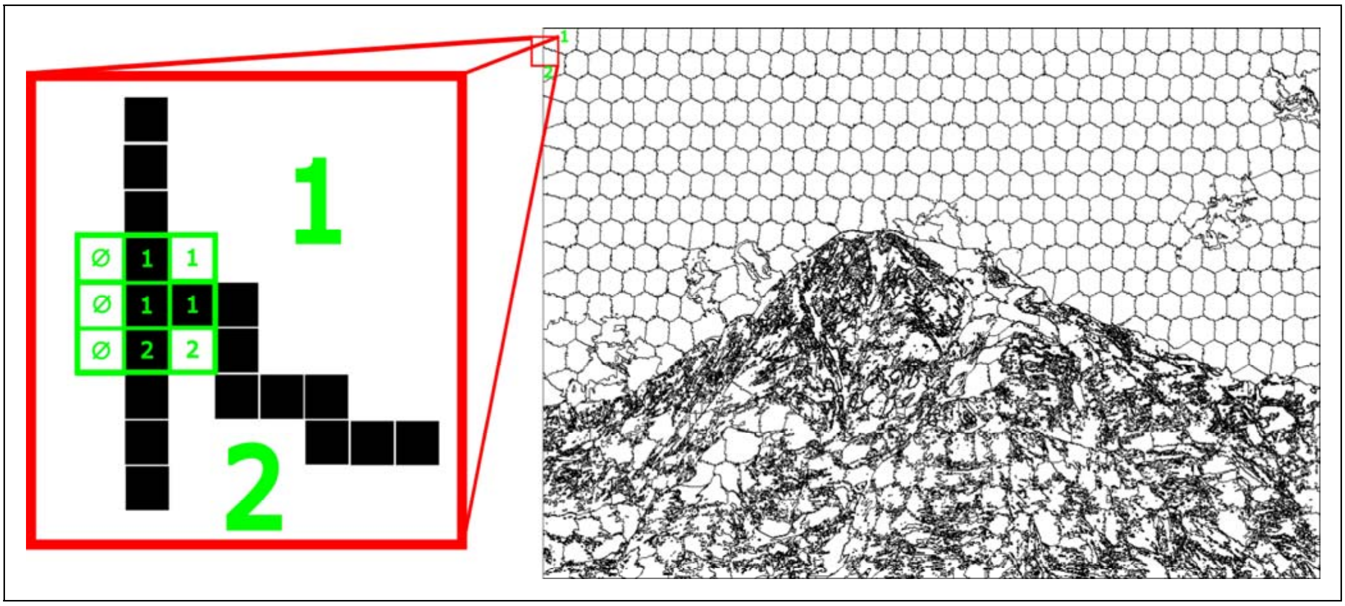


Figure 2: Neighborhood Analysis with Labled Superpixels

$$\begin{aligned} sx^d = \{p_i^d = (x,y) \in sp^d | \\ id((x-1,y-1)) \neq id(p_i^d) \\ \vee\, id((x,y-1)) \neq id(p_i^d) \\ \vee\, id((x+1,y-1)) \neq id(p_i^d) \\ \vee\, id((x-1,y)) \neq id(p_i^d) \\ \vee\, id((x+1,y)) \neq id(p_i^d) \\ \vee\, id((x-1,y+1)) \neq id(p_i^d) \\ \vee\, id((x,y+1)) \neq id(p_i^d) \\ \vee\, id((x+1,y+1)) \neq id(p_i^d)\} \end{aligned} \tag{3}$$

The following usage of the term superpixel refers to its shape. We show that the shape of a superpixel can be a useful feature for image boundaries. Therefore we define well-formed and malformed superpixels which depend on a selected reference superpixel sx^{ref}.

3.1 Reference Phase

The selection of the reference superpixel sx^{ref} has a major impact on the results of the presented approach. Intuitively one could calculate a regular hexagon as the reference superpixel. However it emerged that although there are regular superpixels looking like a perfect hexagon, they usually have a low computed similarity to a perfect hexagon. This is why we decided to extract the reference superpixel from the superpixels generated from the original image. Therefore we first calculate the average similarity $\overline{sim}$ for every superpixel sx_k with $k \in \{1, ..., n\}$ and n is the number of superpixels generated, as follows

$$\overline{sim}(sx^k) = \frac{1}{n}\sum_{i=1}^{n} sim(sx_k, sx_i). \tag{4}$$

The similarity function sim can be any similarity function which compares two shapes. In this paper we used the OpenCV [2] implementation of Hu invariants [9].
Finally we choose the superpixel with the highest average similarity $\overline{sim}$. That means the reference superpixel sx^{ref} is defined as

$$sx^{ref} = arg\,max_{sx_i \in sx_1,...,sx_n} \overline{sim}(sx_i). \tag{5}$$

3.2 Comparison Phase

In this phase superpixels are classified. Although we consider the problem as a two class problem, the definition of more classes is imaginable. The classes defined in our approach are well-formed and malformed superpixels. A superpixel sx which is categorized as well-formed must hold the condition

$$sim(sx, sx^{ref}) > \delta \tag{6}$$

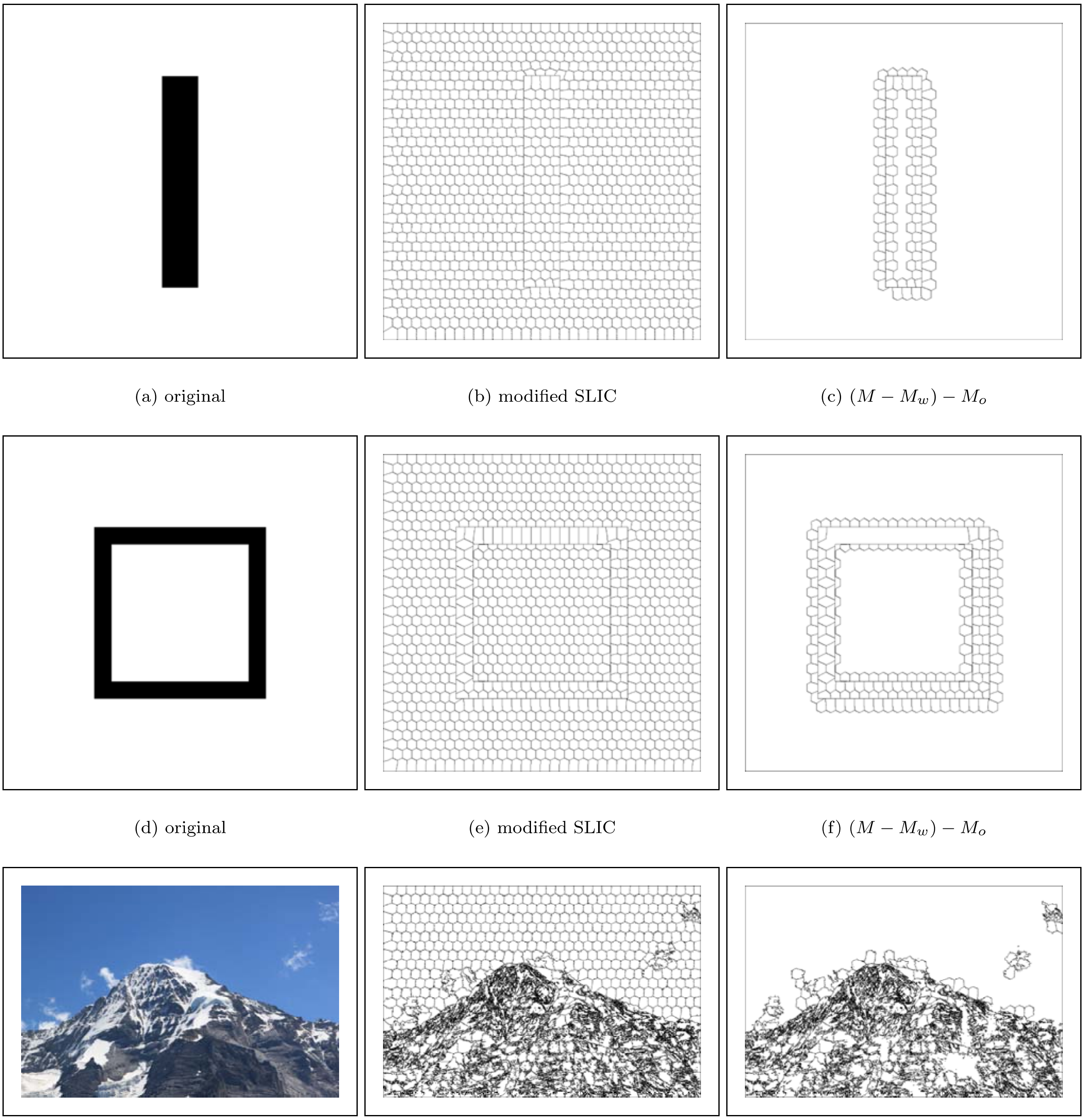
while a malformed superpixel must hold

$$sim(sx, sx^{ref}) < \delta. \tag{7}$$

With δ is a chosen threshold and sim is the same similarity function which has been chosen in section 3.1. For further processing every superpixel is labeled with its appropriate class c

$$sx^c = (sx, class(sx)). \tag{8}$$

The classification function is defined as



(a) original (b) modified SLIC (c) $(M - M_w) - M_o$

(d) original (e) modified SLIC (f) $(M - M_w) - M_o$

(g) original (h) modified SLIC (i) $(M - M_w) - M_o$

Figure 3: Application of the presented approach

$$class(sp) = \begin{cases} w, & \text{if } sim(sx, sx^{ref}) > \delta, \\ m, & \text{if } sim(sx, sx^{ref}) < \delta. \end{cases} \quad (9)$$

Finally superpixels of one of the determined classes are filtered. This results in a set of either well-formed or malformed superpixels. Let $M = \{sx_1, ..., sx_k\}$ be the set of all superpixels then M_w and M_m are the sets which contain only the well-formed and malformed superpixels respectively. Drawing the superpixel-boundaries of $M - M_w$ or $M - M_m$ leads to an image which contains superpixels which either represent edge-free areas of the original image or areas which contain boundaries within the original image depending on the removed class of M. In the following the removed class is denoted as M_{del}. However the resulting set of superpixels still contains some outliers which are identified in the next phase.

3.3 Neighborhood Analysis

The goal of the neighborhood analysis is the identification of outliers which remained from the previous processing.
This phase requires the definition of the neighborhood of a superpixel. Therefore we inspect the neighborhood N of every pixel $p = (x, y)$ within a remaining superpixel. The neighborhood N_p of a pixel p is defined as

$$N_p = \{n(x-1, y-1), n(x, y-1), n(x+1, y-1)), \\ n(x-1, y), n(x+1, y), n(x-1, y+1), n(x, y+1), \\ n(x+1, y+1)\} \tag{10}$$

with

$$p \mapsto (id \circ centroid)(p) := n(p). \tag{11}$$

and in case $x \notin \{1, width\}, y \notin \{1, height\}$ or $p^d = (x, y) \in sx^d$ with $sx^d \in M_{del}, n(x, y) = \emptyset$ is defined. A demonstration for one pixel of a superpixel can be seen in Figure 2.
Then the neighborhood of a superpixel sx with m pixels and the label d is defined as

$$N_{sx^d} = \{N_{p_1^d} \cup ... \cup N_{p_m^d}\}. \tag{12}$$

Now a superpixel can be classified as an outlier if it holds

$$|N_{sx^d}| < \phi \tag{13}$$

where $|\cdot|$ is cardinality (set size) and ϕ is a predefined threshold. Let M_o be the set of superpixels which hold the outlier condition. Then the result of this phase can be written as $(M - M_{del}) - M_o$.

4. APPLICATION

In this section the previous described method is applied to artificial images and photographs. Therefore 25 artificial images with common geometric figures such as triangles, circles and rectangles on white backgrounds were created. The set of 203 photographs is taken from [1].
In order to realize the experiments following choices were made: The original SLIC algorithm requires the amount of superpixels as a parameter. Here we have chosen 900 superpixels. Furthermore we have chosen the OpenCV [2] implementation of Hu invariants as the similarity function.
Figure 3 shows the application of the presented approach to two artificial images and one photograph. It turned out, that the selection of δ and ϕ depends on the image and the resulting superpixels. Nevertheless, once one has identified good values for δ and ϕ, well-formed or malformed superpixels are recognized successfully. The experiments have also shown that the parameter-choice for photographs is much more complex than for the artificial images. This can be explained with the high variation of superpixel-shapes in photographs.
However, as can be seen in Figure 3(c), processed artificial images can still contain superpixels which do not align to image boundries. This happens because border-aligned-superpixels also influence the shape of their neighbours as can be seen in Figure 3(b).
Another interesting observation can be made in Figure 3(e). It can be seen that the shape of superpixels is also dependend on the orientation of the image. This property leads to different superpixel-shapes and thereby to different results of our approach for equal or similar borders. The influence of the superpixel-size is also noticeable. Since the cell-size of the superpixels is too big, the edge-free area within the borders in Figure 3(f) is not always identified as such.

5. CONCLUSION

In this paper we presented a method to identify well-formed and malformed superpixels in order to detect areas with image boundaries. As a base for our approach we used a slight modification of the SLIC algorithm. It was assumed that well-formed superpixels represent edge-free areas within an image. In the sense of image structures these areas hold too little information to contribute to the understanding of the image. In the course of this paper it could be shown that the shape of superpixels contains information about the image structure. The application of the technique on photographs and artificial images has shown that the assumptions we made have been proved true. Although the exact location of the detected edges could not be identified, it is possible to estimate the location in the area of the remaining superpixels. Furthermore the deletion of a certain class of superpixels lowers the complexity of an image which is beneficial for further processing. Nevertheless, once these parameters are identified, good results can be expected. Future work should deal with this parameterization-problem. Then the exact location of the borders should be calculated. These results should be evaluated with the help of the Berkeley Segmentation Dataset [14]. Another interesting field of study is the adoption of the presented method to other superpixel algorithms and their comparison. It is also assumed that different similarity functions lead to different results.

6. REFERENCES

[1] G. Baatz, O. Saurer, K. Koser, and M. Pollefeys. Large Scale Visual Geo-Localization of Images in Mountainous Terrain. In *Computer Vision - ECCV 2012*. 2012.

[2] G. Bradski. Opencv. *Dr. Dobb's Journal of Software Tools*, 2000.

[3] C. Conrad, M. Mertz, and R. Mester. *Contour-Relaxed Superpixels*, pages 280–293. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[4] M. V. den Bergh, X. Boix, G. Roig, and L. J. V. Gool. Seeds: Superpixels extracted via energy-driven sampling. *CoRR*, abs/1309.3848, 2013.

[5] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *Int. J. Comput. Vision*, 59(2):167–181, Sept. 2004.

[6] B. Fulkerson and S. Soatto. *Really Quick Shift: Image Segmentation on a GPU*, pages 350–358. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[7] A. A. E. Greg Mori, Xiaofeng Ren, J. M. at UC Berkeley, and U. of Oxford. Recovering human body configurations: Combining segmentation and recognition. 2014.

[8] X. X. Hai Wang, Xiongyou Peng and Y. Liu. Bslic: Slic superpixels based on boundary term. 2017.

[9] M.-K. HU. Visual pattern recognition by moment invariants. 1962.

[10] P. Kovesi. Image segmentation using slic superpixels and dbscan clustering.

[11] L. M.-Y. T. O. R. S. C. R. M. E. R. LABORATORIES. Entropy rate superpixel segmentation. 2011.

[12] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi. Turbopixels: Fast superpixels using geometric flows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(12):2290–2297, Dec. 2009.

[13] Z. Li and J. Chen. Superpixel segmentation using linear spectral clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 1356–1363, 2015.

[14] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[15] F. Perbet, B. Stenger, and A. Maki. Homogeneous superpixels from markov random walks. *IEICE Transactions*, 95-D(7):1740–1748, 2012.

[16] X. Ren and U. o. C. a. B. Jitendra Malik. Learning a classification model for segmentation. 2014.

[17] C. Rother, V. Kolmogorov, and A. Blake. "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, Aug. 2004.

[18] D. Stutz. Superpixel segmentation: An evaluation. In J. Gall, P. Gehler, and B. Leibe, editors, *Pattern Recognition*, volume 9358 of *Lecture Notes in Computer Science*, pages 555 – 562. Springer International Publishing, 2015.

[19] R. A. A. S. K. S. A. L. P. F. S. Susstrunk. School of computer and communication sciences (ic) ecole polytechnique fedrale de lausanne (epfl). 2010.

[20] O. Veksler, Y. Boykov, and P. Mehrani. Superpixels and supervoxels in an energy optimization framework. In *Proceedings of the 11th European Conference on Computer Vision: Part V*, ECCV'10, pages 211–224, Berlin, Heidelberg, 2010. Springer-Verlag.

[21] M. Y. Yang. Image segmentation by bilayer superpixel grouping. In *Asian Conference on Pattern Recognition*, Nov. 2013.

[22] N. Y. Zhenguo Li Xiao-Ming Wu Shih-Fu Chang Dept. of Electrical Engineering, Columbia University. Segmentation using superpixels: A bipartite graph partitioning approach. 2012.