# Chapter 1

# Extended Berkeley Segmentation Benchmark

Most measures provided as part of the Berkeley Segmentation Benchmark are unsuited for evaluating superpixel algorithms. Therefore, in this section, we discuss most of the available measures used to evaluate superpixel algorithms. These have, except for Boundary Recall, in the course of this thesis, been implemented into the framework of the Berkeley Segmentation Benchmark.

## 1.1 Boundary Recall

Boundary Recall is part of the Precision-Recall-Framework introduced in [MFM04]. Let $S$ be a superpixel segmentation and $G$ be a ground truth segmentation, then we define:

**True Positives, $TP(S,G)$:** Boundary pixels in $G$ for which there is a boundary pixel in $S$ in range $r$.

**False Negatives, $FN(S,G)$:** Boundary pixels in $G$ for which there is no boundary pixel in $S$ in range $r$.

where $r$ defines a tolerance parameter controlling the allowed deviation from the ground truth. In practice, $r$ is set to 0.0075 times the image diagonal[1]. The Boundary Recall $Rec(S,G)$ is the fraction of all boundary pixels within the ground truth segmentation $G$ which are correctly detected within the superpixel segmentation $S$, that is

$$Rec(S,G) = \frac{|TP(S,G)|}{|TP(S,G)| + |FN(S,G)|}. \quad (1.1)$$

As we want superpixels to respect the boundaries within the image, a high Boundary Recall is desirable.

[1] This is the default value as used by the Berkeley Segmentation Benchmark and we chose not to change this value to keep our results comparable to other publications.

## 1.2 Undersegmentation Error

The Undersegmentation Error describes the leakage or "bleeding" [LSK+09] of a superpixel with respect to a specific ground truth segment. In [LTRC11] and [VBM10], the Undersegmentation Error is defined as

$$UE(S,G) = \frac{1}{N}\left(\sum_{G_i \in G}\left(\sum_{S_j \cap G_i \neq \emptyset} |S_j|\right) - N\right). \tag{1.2}$$

This equation computes the total amount of leakage for all ground truth segments and normalizes the sum by the number of pixels[2]. We note that the normalization used above may not be sufficient. For example consider a superpixel segmentation with a single superpixel covering the whole image. Then, if the ground truth consists of more than two segments, the Undersegmentation Error will be greater than one. In addition, as stated by Neubert and Protzel [NP12] as well as Achanta et al. [ASS+10] this definition has another serious disadvantage. Consider a large superpixel covering one ground truth segment perfectly except for some pixels. In the above equation, such superpixels receive high penalties [NP12]. Achanta et al. adapt equation (1.2) to tolerate a small amount of leakage per superpixel:

$$UE(S,G) = \frac{1}{N}\left(\sum_{G_i \in G}\left(\sum_{|S_j \cap G_i| > B} |S_j|\right) - N\right) \tag{1.3}$$

where $B$ is the corresponding tolerance parameter. To avoid an additional parameter, we implemented the Undersegmentation Error as proposed by Neubert and Protzel:

$$UE(S,G) = \frac{1}{N}\sum_{G_i \in G}\sum_{S_j \cap G_i \neq \emptyset} \min\{|S_j \cap G_i|, |S_j - G_i)\}. \tag{1.4}$$

where for each superpixel only the smaller part is considered leakage.

## 1.3 Achievable Segmentation Accuracy

When using superpixels as pre-processing step, we want the performance of subsequent steps to be as far as possible unaffected [LTRC11]. Of course, as we inevitably loose information, this is not possible. However, we would like to quantify the accuracy achievable by subsequent steps, as for example classical segmentation. Achievable Segmentation Accuracy labels superpixels according to their underlying ground truth segments and counts the correctly labeled pixels [LTRC11]:

$$ASA(S,G) = \frac{1}{N}\sum_{S_j \in S}\max_{G_i}\{|S_j \cap G_i|\}. \tag{1.5}$$

[2]We note that equation (1.2) uses a slightly different normalization when compared to the Undersegmentation Error as used by Levinshtein et al. in [LSK+09].

Therefore, Achievable Segmentation Accuracy represents an upper bound on the accuracy achievable by a subsequent segmentation step [LTRC11].

## 1.4 Compactness

Schick et al. [SFS12] propose a compactness measure for superpixels based on the isoperimetric quotient. Given a superpixel $S_j$, the perimetric quotient relates the area $A(S_j)$ of the superpixel to the area of a circle with the same perimeter $U(S_j)$:

$$\frac{4\pi A(S_j)}{U(S_j)^2}. \tag{1.6}$$

As the circle represents the most compact form, the perimetric quotient measures the compactness of the superpixel, reaching 1 if and only if the superpixel has the shape of a circle. The proposed compactness measure considers the perimetric quotient of all superpixels weighted by their area:

$$CO(S) = \frac{1}{N} \sum_{S_j \in S} |S_j| \frac{4\pi A(S_j)}{U(S_j)^2}. \tag{1.7}$$

Although we assume superpixels to represent connected components, in practice, this will not always be the case. Therefore, we need to enforce connectivity before being able to compute the perimeter of the superpixels. Then, the above compactness measure captures the notion of spatial coherence within superpixels. The need of being able to measure compactness is also supported by Ren and Malik who introduce the concept of superpixels as “local” and “coherent” [RM03].

## 1.5 Sum-of-Squared Error

To measure the quality of the superpixel segmentation without being dependent on a ground truth, we propose to use the Sum-of-Squared Error as used for clustering evaluation as well:

$$SSE(S) = \frac{1}{N} \sum_{S_j \in S} \sum_{x_n \in S_j} d(x_n, S_j)^2 \tag{1.8}$$

where $d(x_n, S_j)$ can be an arbitrary distance. In our case the euclidean distance in color space is suitable:

$$d(x_n, S_j) = \|I(x_n) - I(S_j)\|_2. \tag{1.9}$$

By coloring each pixel according to the corresponding superpixel's mean color, the superpixel segmentation is interpreted as reconstruction of the original image and the Sum-of-Squared Error measures the reconstruction error.

## 1.6 Explained Variation

Another measure not depending on a ground truth segmentation is the Explained Variation. The Explained Variation quantifies how well the color variation within the image is captured by the superpixel segmentation [FCT$^+$14]:

$$EV(S) = \frac{\sum_{S_j \in S} (I(S_j) - \mu)^2}{\sum_{n=1}^{N} (I(x_n) - \mu)^2} \tag{1.10}$$

where $\mu$ is the mean color of the image. The information carried by an image is primarily defined by variation in color or intensity. Thus, Explained Variation measures the fraction of information captured by the superpixel segmentation.

# Bibliography


[ASS+10] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels. Technical report, École Polytechnique Fédérale de Lausanne, Lusanne, Switzerland, June 2010.

[FCT+14] H. Fu, X. Cao, D. Tang, Y. Han, and D. Xu. Regularity preserved superpixels and supervoxels. *Multimedia, Transactions on*, 16(4):1165–1175, June 2014.

[LSK+09] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi. TurboPixels: Fast superpixels using geometric flows. *Pattern Analysis and Machine Intelligence, Transactions on*, 31(12):2290–2297, December 2009.

[LTRC11] M. Y. Lui, O. Tuzel, S. Ramalingam, and R. Chellappa. Entropy rate superpixel segmentation. In *Computer Vision and Pattern Recognition, Converence on*, pages 2097–2104, Providence, Rhode Island, June 2011.

[MFM04] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *Pattern Analysis and Machine Intelligence, Transactions on*, 26(5):530–549, May 2004.

[NP12] P. Neubert and P. Protzel. Superpixel benchmark and comparison. In *Forum Bildverarbeitung*, Regensburg, Germany, November 2012.

[RM03] X. Ren and J. Malik. Learning a classification model for segmentation. In *Computer Vision, International Conference on*, pages 10–17, Nice, France, October 2003.

[SFS12] A. Schick, M. Fischer, and R. Stiefelhagen. Measuring and evaluating the compactness of superpixels. In *Pattern Recognition, International Conference on*, pages 930–934, Tsukuba, Japan, November 2012.

[VBM10] O. Veksler, Y. Boykov, and P. Mehrani. Superpixels and supervoxels in an energy optimization framework. In *Computer Vision, European Conference on*, volume 6315 of *Lecture Notes in Computer Science*, pages 211–224. Springer Berlin Heidelberg, 2010.