# 8 Segmentation Difference

## 8.1 Inspiration

Section 4 presented methods that could evaluate two segmentations. Many of them suffer some pitfalls, which was described in section 6. I will use these drawbacks to create new method, which would be superior to all presented methods. On the other hand, there are some basic properties, which have to be selected from possible options, but none of these options is good or bad. Namely, it is symmetry of a method. Evaluation should represent somehow the similarity of segmentations. It could be seen as a distance of these segmentations. Distance is symmetric, thus I decided to propose a symmetric method.

Two different segmentations of the same image should have the borders of segments on the same place. Still, the number of segments can differ and, therefore, not all borders could be on the same place. Because there is no predefined number of segments, typically, we cannot evaluate this difference in number of segments as a fault but as a property. However, the common borders should be on the same place. Thus the precision of common borders has to be measured and will show similarity of segmentations. On the other hand, there is also the difference in number of segments. This should be also evaluated but should not be combined with precision of borders. Difference in number of segments is just a property, but precision of borders corresponds to quality of segmentations. The only possible way is to represent the result as a couple of numbers.

Older variants of the proposed method was already published in [39] and used in [38] and [41]. Whole method will be revisited and improved here. The basis remains the same but some parts will be modified. This could bring better flexibility for special cases and more precise and stable results.

Algorithm of proposed method can be divided into three parts. First, we need to find correspondences of segments. These correspondences are used to compute difference of number of segments. Common difference of number of segments is not appropriate, thus I will call it granularity difference. The reasons will be shown later. Finally, the correspondence is used to merge segments to obtain segmentations with the same number of segments. Now, we could measure precision of all borders, because these are the common borders of original segmentations. All three steps will be described in detail in the following three subsections.

## 8.2 Correspondences

Each segment in a segmentation represents some object in an image. If the same object is represented in both segmentations by a single segment, then these segments can be called correspondent. Such correspondence is called one-one. Sometimes a single object is represented by a single segment in one segmentation, but by more than one segment in the other segmentation. This correspondence will be denoted as one-many. Finally, there can be more segments representing the same object in both segmentations. This type will be called many-many. For more details see section 7.5.

First type one-one cannot be the only one type used for correspondence. Although it is used by many methods, this cannot reflect different number of segments. Proposed method uses one-many and many-many correspondences. Still, there can be one problem with many-many correspondences. If both whole segmentations belong to a single many-many correspondence, then the common border will be the border of an image. The result of distance of borders is zero, evidently. Such result can be falsely interpreted as identical segmentations. This could not happen using one-many correspondence. In this case, such result can happen if one segmentation consists of a single segment only. Both variants with one-many and many-many correspondences will be implemented and tested. The results will show, which variant is the better.

## 8.3 Granularity Difference

Divergence of number of segments of two segmentations could be represented as plain difference

$$GD(S_1, S_2) = ||S_1| - |S_2||\,. \tag{155}$$

Still there could be some problems. The number of segments could be the same, and yet the number of segments in correspondences could differ (see figure 13). Therefore, we should evaluate each correspondence separately

$$GD(S_1, S_2) = \sum_i g(i), \tag{156}$$

where $c_i$ represents $i$-th correspondence and $g(i)$ is granularity of a correspondence. Big correspondences should influence the result more than smaller correspondences:

$$GD(S_1, S_2) = \frac{\sum_i |c_i| \cdot g(i)}{\sum_i |c_i|}, \tag{157}$$

$$|c_i| = \sum_{j \in c_{i1}} |s_{1j}| + \sum_{j \in c_{i2}} |s_{2j}|, \tag{158}$$

where $c_i$ is $i$-th correspondence, $c_{i1}$ is set of indexes of segments from segmentation $S_1$ of the $i$-th correspondence, $s_{1j}$ is $j$-th segment from segmentation $S_1$ and $g(i)$ represents granularity of correspondence $c_i$, which is about to be defined. It can be seen, that following holds

$$\sum_i |c_i| = 2 \cdot |S_1| = 2 \cdot |S_2|. \tag{159}$$

Two segmentations are evaluated without the original image. Thus we cannot assume priority of segments according to their content. One of possible solutions is to take size of segments into account. In the figure 14 we could see three different segmentations. First segmentation shows splitting of whole area into two equal parts. One segment in the second segmentation is very small, while the other covers the rest. Such small segment could be created due to noise in the original image. Typical evaluation, like in (155), would assign number two to both segmentations because they consist of two

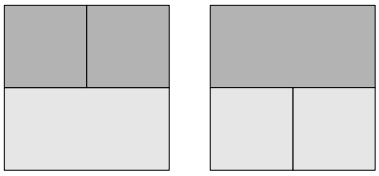

Figure 13: Two segmentations with the same number of segments but with different number of segments in correspondences. Correspondent segments are highlighted by the same color.

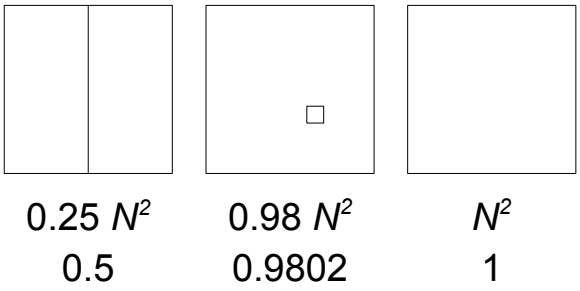


Figure 14: First segmentation consists of two segments of the same size. Second segmentation includes two segments with different sized segments. Third segmentation includes only one segment. First row of values represent unnormalized results according to (160). Second row of values are normalized results according to (161).

segments. If the small segment would have size of one pixel, then such segmentation would be much more similar to the segmentation consisting of a single segment (third segmentation in the figure 14). That is the reason why I propose to evaluate equally sized segments by different value than unequally sized segments of the same number (see figure 14 for sample results):

$$g(i) = \sum_{j \in c_{i1}} |s_{1j}|^2. \tag{160}$$

Value representing difference in granularity of segmentations should not be influenced by the resolution of the image. To be more exact, if we double the resolution of the image in each direction, we should get the same result as with the original resolution. On the other hand, higher resolution allows us to segment more precisely and we could create smaller segments as well. Such possibilities could change the result, naturally. We could normalize the results to ensure either the first or the second group of mentioned

cases. I propose normalization of the first type (see figure 14 for sample results):

$$g(i) = \sum_{j \in c_{i1}} \left( \frac{|s_{1j}|}{\sum_{k \in c_{i1}} |s_{1k}|} \right)^2. \tag{161}$$

Current formula (161) is usable and its results of some sample segmentations can be seen in figure 15. Each increasing of granularity by one step is evaluated by $0.5$. Result of two steps is $0.25$ etc. Evidently, the progression is geometric. We could use logarithm to convert it to arithmetic progression, which is more natural

$$g(i) = -\log_2 \left[ \sum_{j \in c_{i1}} \left( \frac{|s_{1j}|}{\sum_{k \in c_{i1}} |s_{1k}|} \right)^2 \right]. \tag{162}$$

This approach is similar to sound processing. Acoustic pressure is often converted into sound level represented in dB. Since the values are fractional, I use minus to obtain positive result. Base of the logarithm was chosen according to multiplier in the geometric progression. Evaluation of sample segmentations are presented in figure 16.

Current formula can evaluate segments from the first segmentation only. It can be easily adapted for arbitrary segmentation but still it can be used for one-many correspondences only. Many-many correspondences include more than one segment in both segmentations. Extension of the formula is based on the virtual splitting of the many-many correspondence into two one-many. First, we evaluate granularity difference of segments in a correspondence from the first segmentation to a single segment. Then we execute the same evaluation for the second segmentation. Simple example is shown in figure 17. Previous formula is, therefore, extended and whole proposed granularity difference is defined as follows:

$$GD(S_1, S_2) = \frac{\sum_i |c_i| \cdot g(i)}{\sum_i |c_i|}, \tag{163}$$

$$|c_i| = \sum_{j \in c_{i1}} |s_{1j}| + \sum_{j \in c_{i2}} |s_{2j}|, \tag{164}$$

$$g(i) = -\log_2 \left( \left[ \sum_{j \in c_{i1}} \left( \frac{|s_{1j}|}{\sum_{k \in c_{i1}} |s_{1k}|} \right)^2 \right] \cdot \left[ \sum_{j \in c_{i2}} \left( \frac{|s_{2j}|}{\sum_{k \in c_{i2}} |s_{2k}|} \right)^2 \right] \right), \tag{165}$$

where $c_i$ is $i$-th correspondence, $c_{i1}$ is set of indexes of segments from segmentation $S_1$ of the $i$-th correspondence, $s_{1j}$ is $j$-th segment from segmentation $S_1$ and $g(i)$ represents granularity of correspondence $c_i$.

Many-many correspondences include all segments of both segmentations. On the other hand, not all segments could be included in one-many correspondences. If some segment would cause creation of many-many correspondence by adding it to current one-many correspondence then it cannot be added. Such segment can be left unprocessed. These unprocessed segments will create one null-many correspondence for each segmentation. These two correspondences are processed in Granularity Difference (165)

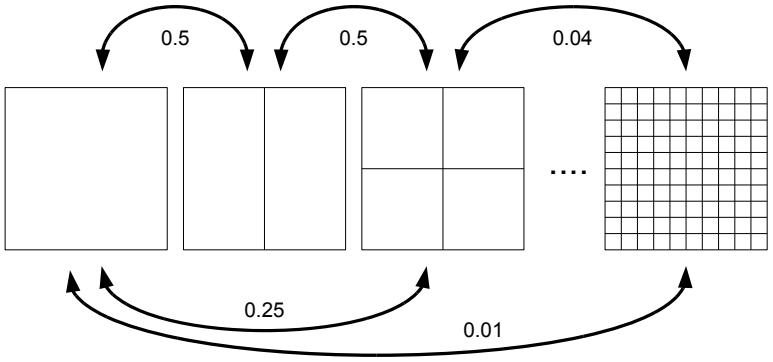


Figure 15: Granularity difference using equations (157) and (161).

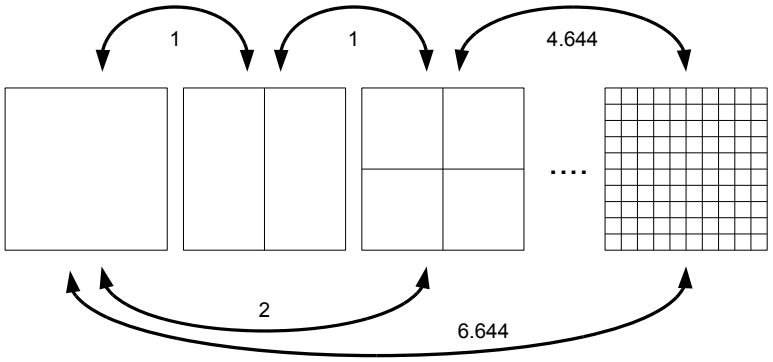


Figure 16: Granularity difference using equations (157) and (162).

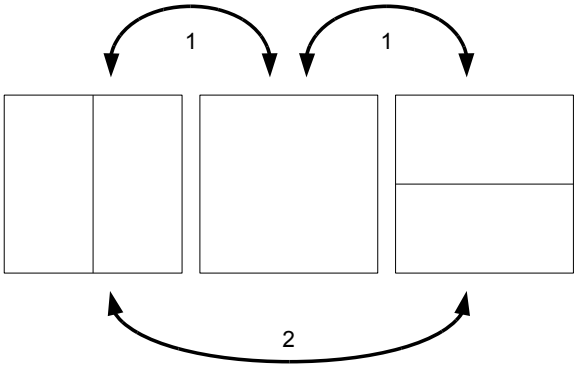


Figure 17: Evaluation of many-many correspondence of the first and the third segmentation using evaluation of one-many correspondences.

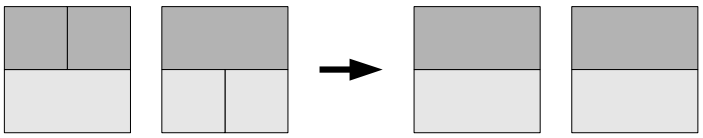

Figure 18: Merging of segments in each correspondence separately. Correspondent segments are highlighted by the same color.

but instead of null segment, whole image is taken. Therefore, the result of GD (165) would be little higher. These null-many correspondences will not be processed by following Border Distance because there is no correspondent border. We could use border of the image but these typically small segments could cause great difference in result and the overall quality could decrease rapidly.

## 8.4 Border Distance

Evaluation started by finding of correspondences. Granularity difference was calculated using the correspondences. Finally, similarity of borders will be evaluated. For such task, we need equal number of segments. This can be easily accomplished by merging segments in each correspondence (see figure 18).

All correspondences were converted into type one-one. Again, we will evaluate each correspondence separately, thus we need evaluation for two correspondent segments only. Similarity or divergence of two segments can be measured in different ways. The simplest is measuring of the common area

$$BD(S_1, S_2) = \sum_i |c_{i1} \cap c_{i2}|, \tag{166}$$

where $c_{i1}$ represents segment from $i$-th one-one correspondence from segmentation $S_1$. Such approach does not take a shape of segments into account. One of possible modifications is to compute distance of all pixels to the correspondent segment

$$BD_1(S_1, S_2) = \sum_i \left( \sum_{j \in c_{i1}} d(j, c_{i2}) + \sum_{j \in c_{i2}} d(j, c_{i1}) \right), \tag{167}$$

$$d(j, x) = \min_{k \in x} ||j - k||_2, \tag{168}$$

where $d(j, x)$ represents distance of a point to a set using $L_2$ metric. Another alternative could be measuring of distance of borders only. Results could be quite different, still, the formula is similar:

$$BD(S_1, S_2) = \sum_i \left( \sum_{j \in bord(c_{i1})} d(j, bord(c_{i2})) + \sum_{j \in bord(c_{i2})} d(j, bord(c_{i1})) \right), \tag{169}$$

where $bord(s)$ is set of border pixels of segment $s$. This solution evaluates shape of segments, as well as difference in curvature of borders. But the curvature influences the result drastically, thus we could normalize the inner part to obtain average distance:

$$BD(S_1, S_2) = \sum_i \left( \frac{\sum_{j \in bord(c_{i1})} d(j, bord(c_{i2}))}{|bord(c_{i1})|} + \frac{\sum_{j \in bord(c_{i2})} d(j, bord(c_{i1}))}{|bord(c_{i2})|} \right). \quad (170)$$

Perimeter could be quite different from area of a segment, therefore small segments with very long perimeter could influence the result heavily. To penalize such extreme cases, each correspondence could be weighted by its size:

$$BD_2(S_1, S_2) = \frac{\sum_i |c_i| \left( \frac{\sum_{j \in bord(c_{i1})} d(j, bord(c_{i2}))}{|bord(c_{i1})|} + \frac{\sum_{j \in bord(c_{i2})} d(j, bord(c_{i1}))}{|bord(c_{i2})|} \right)}{\sum_i |c_i|}, \quad (171)$$

$$|c_i| = \sum_{j \in c_{i1}} |s_{1j}| + \sum_{j \in c_{i2}} |s_{2j}|. \quad (172)$$

Both versions of proposed border distance (167) and (171) are still dependent on resolution. If we double width and height, the area will be four times larger but distances and border lengths will be just two times longer. Compensation should account these facts and each version of border distance will be adjusted differently:

$$BD_1(S_1, S_2) = \frac{\sum_i \left( \sum_{j \in c_{i1}} d(j, c_{i2}) + \sum_{j \in c_{i2}} d(j, c_{i1}) \right)}{w \cdot h \cdot \sqrt{w \cdot h}}, \quad (173)$$

$$BD_2(S_1, S_2) = \frac{\sum_i |c_i| \left( \frac{\sum_{j \in bord(c_{i1})} d(j, bord(c_{i2}))}{|bord(c_{i1})|} + \frac{\sum_{j \in bord(c_{i2})} d(j, bord(c_{i1}))}{|bord(c_{i2})|} \right)}{\sqrt{w \cdot h} \cdot \sum_i |c_i|}, \quad (174)$$

$$d(j, x) = \min_{k \in x} ||j - k||_2, \quad (175)$$

$$|c_i| = \sum_{j \in c_{i1}} |s_{1j}| + \sum_{j \in c_{i2}} |s_{2j}|, \quad (176)$$

where $c_{i1}$ represents segment from $i$-th one-one correspondence from segmentation $S_1$, $bord(s)$ is set of border pixels of segment $s$, $w$ and $h$ is width and height of an image, respectively.

Previous modification compensated influence of the resolution. Moreover, we could normalize the results into interval $< 0, 1 >$. We could take the worst possible case of segmentations and divide the formula by its result. Although it is theoretically the best way, practical segmentations would not reach such extreme values, thus the results would be very low. For typical cases, it is more convenient to use pseudonormalization. We take one practical pair of segmentations, which has high result and we divide the formula by the result. Using pseudonormalized formula, similar segmentations will have results in the interval $< 0, 1 >$. Results higher than 1 indicate totally different segmentations. Often it is case of discrete segments, where parts of the same segments lie between parts of other segments.

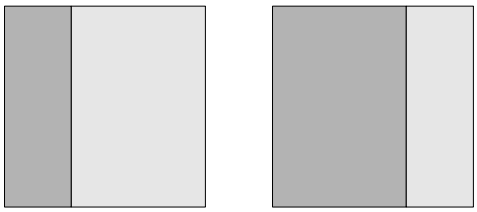

Figure 19: Two segmentations used for pseudonormalization of border distance. Correspondence is indicated by common color. Smaller segments take $1/3$ of the total area.

Proposed fomulas will be pseudonormalized according to segmentations in figure 19. If one of the small segments would be even smaller, then the correspondence would be different, therefore this is a boundary state. Although the correspondent segments look differently, they are still treated as correspondent. Derivation of normalization constant follows:

$$BD_1(S_1, S_2) = \frac{\sum_i \left( \sum_{j \in c_{i1}} d(j, c_{i2}) + \sum_{j \in c_{i2}} d(j, c_{i1}) \right)}{w \cdot h \cdot \sqrt{w \cdot h}}, \quad (177)$$

$$= \frac{2 \cdot \left( \frac{1}{18} w^2 \cdot h \right)}{w \cdot h \cdot \sqrt{w \cdot h}}, \quad (178)$$

$$= \frac{1}{9} \sqrt{\frac{w}{h}}, \quad (179)$$

where $w$ and $h$ is width and height of an image, respectivelly. Result is evidently not dependent on resolution but on aspect ratio of an image. If we rotate segmentations from figure 19 by $90°$, the result will be the same but with swapped width and height. Still, the numeric constant would remain the same. As we are not interested about aspect ratio, we will use just the numeric constant.

Pseudonormalization of second formula is quite similar to previous:

$$BD_2(S_1, S_2) = \frac{\sum_i |c_i| \left( \frac{\sum_{j \in bord(c_{i1})} d(j, bord(c_{i2}))}{|bord(c_{i1})|} + \frac{\sum_{j \in bord(c_{i2})} d(j, bord(c_{i1}))}{|bord(c_{i2})|} \right)}{\sqrt{w \cdot h} \cdot \sum_i |c_i|}, \quad (180)$$

$$= \frac{2wh \cdot 2 \cdot \frac{\frac{1}{3} wh + 2 \cdot \frac{1}{18} w^2}{\frac{2}{3} w + h}}{\sqrt{w \cdot h} \cdot 2 \cdot w \cdot h}, \quad (181)$$

$$= \frac{2}{3} \sqrt{\frac{w}{h}} \frac{\left( \frac{1}{3} w + h \right)}{\left( \frac{2}{3} w + h \right)}. \quad (182)$$

| | $BD_1$ (188) | $BD_2$ (190) |
|---|---|---|
| one-many | $SD_1$ | $SD_3$ |
| many-many | $SD_2$ | $SD_4$ |

Table 6: Table of variants of Segmentation Difference using different types of correspondence and different definition of Border Distance.

Pseudonormalization need not to be exact, because there is no theoretically the worst couple of segmentations which could practically occur. Therefore, we could slightly adjust current result (182). Expressions in brackets in numerator and denominator are nearly the same, thus we will divide them without remainder and we obtain following much simpler form

$$\frac{2}{3}\sqrt{\frac{w}{h}}. \tag{183}$$

This normalization formula is similar to the previous (179). It consists of a numeric constant and square root of aspect ratio. Again, we will ignore the aspect ratio and we will use only the numerical constant for pseudonormalization.

## 8.5 Proposed Segmentation Difference

Previous subsections defined parts of proposed Segmentation Difference. Following formulas summarize all necessary parts and its final form:

$$SD(S_1, S_2) = (GD(S_1, S_2), BD_x(S_1, S_2)), \tag{184}$$

$$GD(S_1, S_2) = \frac{\sum_i |c_i| \cdot g(i)}{\sum_i |c_i|}, \tag{185}$$

$$g(i) = -\log_2 \left( \left[ \sum_{j \in c_{i1}} \left( \frac{|s_{1j}|}{\sum_{k \in c_{i1}} |s_{1k}|} \right)^2 \right] \cdot \left[ \sum_{j \in c_{i2}} \left( \frac{|s_{2j}|}{\sum_{k \in c_{i2}} |s_{2k}|} \right)^2 \right] \right), \tag{186}$$

$$BD_1(S_1, S_2) = \frac{\sum_i \left( \sum_{j \in c_{i1}} d(j, c_{i2}) + \sum_{j \in c_{i2}} d(j, c_{i1}) \right)}{9 \cdot w \cdot h \cdot \sqrt{w \cdot h}}, \tag{187}$$

$$\tag{188}$$

$$BD_2(S_1, S_2) = \frac{2 \cdot \sum_i |c_i| \left( \frac{\sum_{j \in bord(c_{i1})} d(j, bord(c_{i2}))}{|bord(c_{i1})|} + \frac{\sum_{j \in bord(c_{i2})} d(j, bord(c_{i1}))}{|bord(c_{i2})|} \right)}{3 \cdot \sqrt{w \cdot h} \cdot \sum_i |c_i|}, \tag{189}$$

$$\tag{190}$$

$$|c_i| = \sum_{j \in c_{i1}} |s_{1j}| + \sum_{j \in c_{i2}} |s_{2j}|, \tag{191}$$

$$d(j, x) = \min_{k \in x} ||j - k||_2. \tag{192}$$

Granularity difference was proposed unambiguously. However, there were two possible types of correspondences and two possible algorithms for evaluating of borders.

It is hard to estimate which combination of correspondences and border algorithm will bring the best results. Thus all variants was proposed. Each combination will be compared with other variants and other methods. Comparison will show the best variant, which will be preferred for segmentation evaluation. Table 6 summarizes all variants and shows indication of the variants by subscript.

# 9 Comparison of Evaluation Methods

## 9.1 Methodology

All implemented methods are practically tested on image data sets. They consist of sample images and their ground truth segmentations. Each testing is dependent on type of method, thus we divide methods into three different classes:

- implemented image-segmentation methods:
  $GU$ (15), $LN$ (18), $FOC$ (19), $ZC$ (21), $SE$ (29), $LY$ (30), $BC$ (31);
- implemented segmentation-segmentation asymmetric methods:
  $SM_I$ (35), $SM_{II}$ (36), $SYD$ (37), $L$ (50), $MH$ (52), $YD$ (53), $F$ (54), $MC$ (55), $SFOM$ (58), $SCHD$ (67), $H_{2\mu}$ (69), $W_I$ (89), $OCA$ (103);
- implemented segmentation-segmentation symmetric methods:
  $GCE$ (40), $LCE$ (41), $BCE$ (42), $GBCE$ (43), $HD$ (45), $PD$ (49), $VD$ (51), $JC$ (87), $FM$ (88), $M$ (91), $PRI$ (94), $NMI$ (108), $VI$ (109), $BGM$ (111), $SD$ (184).

Methods in first class can compare an image with a segmentation. If the selected segmentation belongs to the current image, then the result of the method should be low. On the other hand, if the segmentation is ground truth of another image, then the result should be high. Finally, we could set a threshold, which could split these low and high results. Practically, there will be erroneous results. These can be classified into two types. False positives are low results, which should be high instead. False negatives are high results, which should be low. Rates of these errors could be computed and summed. This overall error is dependent on the position of the threshold. We cannot define the threshold a priori because authors do not propose such value. Moreover, it is dependent on type of images. Therefore, the threshold will be set after a test of a method on the whole set. The position of the threshold will be set to minimize the overall error. Evaluation methods are transformed into decision methods using the threshold. This is a possible way how to measure similarity of tested methods. We could call the similarity as a quality of a method. Finally, we could sort tested methods according to its quality and find the best methods in the class.

Second and third class are quite similar. The only difference is in their symmetry. Symmetric methods will compare segmentations from the data sets without defining which segmentation is or is not a ground truth. We just notice if the compared pair belongs to the same image or not. The principle of evaluation of the methods will be the same as for the first class. Results of pairs of segmentations from the same image should be on the one side of a scale, while results of pairs of segmentations from different images should be on the other side. Again, we will set a threshold to minimize overall error a we will find the best method.

Asymmetric methods will be tested on the same data sets as symmetric methods but with one difference. Each pair will be tested twice. Once, one of the pair will be declared

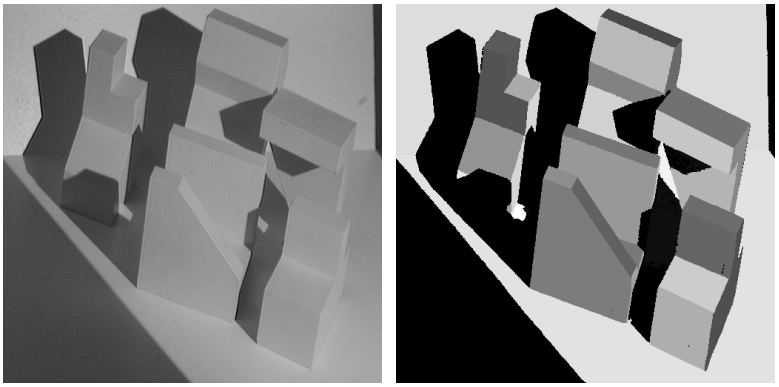

Figure 20: Image and its ground truth segmentation from ABW image set.

as the ground truth. In the second evaluation the segmentations will be swapped. Since all segmentations are defined as ground truth of the image, we can evaluate both possibilities. Number of comparisons will be twice as high as in symmetric methods. Still, the overall error rate is not dependent on number of comparisons. This allows us to merge the results from the second and the third class.

Images in the data sets could have different resolution and they can be single (gray-scale) or multi-channel (color). Some methods from the first class are able to process color images. For the methods that cannot, color image will be converted to gray-scale images before the evaluation. Images in each data set will be divided into subset according to their resolution. None of presented methods can evaluate segmentations with different resolution and different aspect ratios. Therefore, each subset will be evaluated separately.

## 9.2 Image Data Sets

Implemented methods will be tested on image data sets. Each set consists of images and their ground truth segmentations. First set ABW includes different views on simple geometric objects in 3D space (see figure 20). Segments in ground truth images represent planes of corresponding objects. Second image set consists of simple real objects like abacus (see figure 21). Segmentation of each image is quite unambiguous, thus there is only one ground truth segmentation.

Last set is created from real pictures of people, animals and landscapes (see figure 22). It was presented in [22]. Whole database consists of images and each image has four ground truth segmentations at least. Complexity of images is much higher than in the previous image sets, therefore the segmentations can be ambiguous. Each human segmented images differently. This is the only set allowing evaluation of segmentation-segmentation methods due to more ground truth segmentations per image. We could compare two or more different segmentations, which belong to a single image. Images in the database are longitudinal and perpendicular. Still, they have the same resolution if they are appropriately rotated.

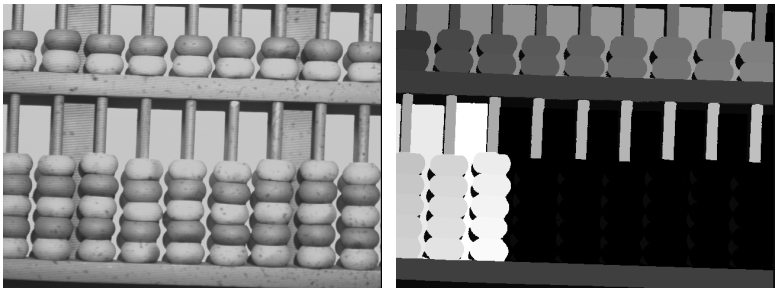

Figure 21: Image and its ground truth segmentation from K2T image set.

Figure 22: Images and its ground truth segmentations from Berkeley segmentation database.

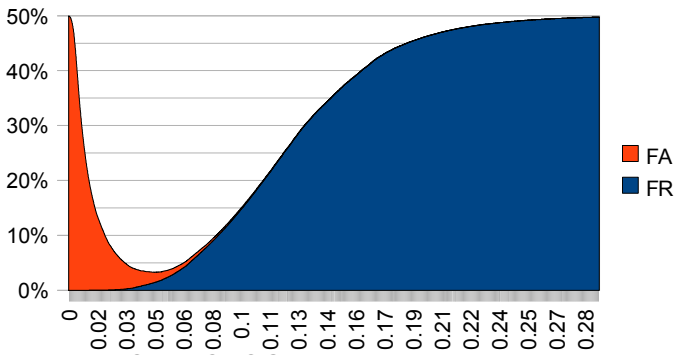


Figure 23: Rates of errors for different thresholds of method $SD_3$ (184).

## 9.3 Results

Segmentation evaluation methods were described and nearly all of them were implemented. Each method outputted one result for one combination of image and segmentation or two segmentations. For some defined threshold, we could compute number of results under and over this threshold. Moreover, we could specify which results should be higher than the threshold and which should be lower. Number of these false results is divided by number of all results and we gain two error rates. They are called false acceptations (FA) and false rejections (FR). These rates variates according to set threshold. Figure 23 shows graph consisting of 6000 thresholds and their corresponding error rates for method $SD_3$ (184).

For final comparison, we take the threshold with the minimum sum of error rates. In case of $SD_3$ (184) it is around $0.05$. Total error cannot exceed 100%. Still, methods does not exceed 50% with any threshold, typicaly. Minimal error rate for arbitrary method can be 50% at most. Even for the worst method, we could set the threshold lower than the lowest result and we get 0 error in FA and 50% error in FR or vice versa. Some the following results were published in [40].

First data set ABW consists of 40 grayscale images and 40 segmentations. Total number of evaluations was 1600 for each method. White boxes with shadows in images is a simple task for majority of methods. The rest: $LN$ (18), $BC$ (31) and $LY$ (30) are evidently poorly defined. Results can be seen in figure 24.

Second data set K2T consists of real object in simple environment. They are stored in 60 grayscale image with 60 segmentations. Number of evaluations for each method raised to 3600. Results can be found in figure 25. More complex images and segmentations changed the order of the first four methods, still the three worst are the same as in ABW data set.

Last and the largest Berkeley segmentation data set consists of 300 color images and 3269 segmentations. Each image has nearly 11 different segmentations in average. In case of image-segmentation methods, each of them evaluated 980700 couples. Images, as

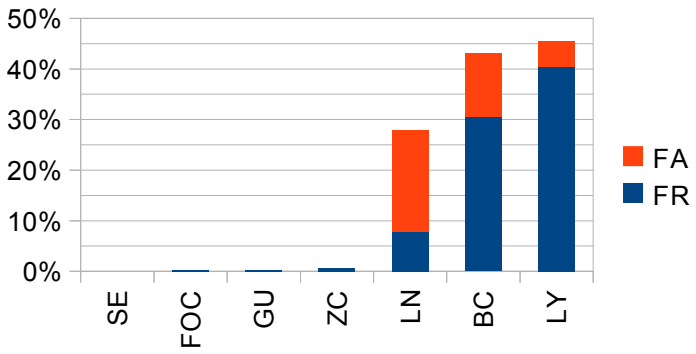


Figure 24: Results of methods for ABW data set.

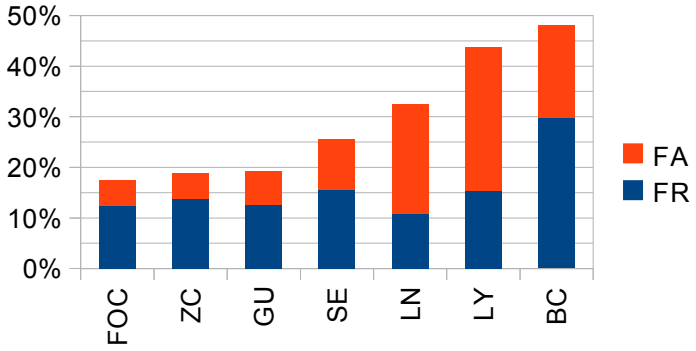


Figure 25: Results of methods for K2T data set.

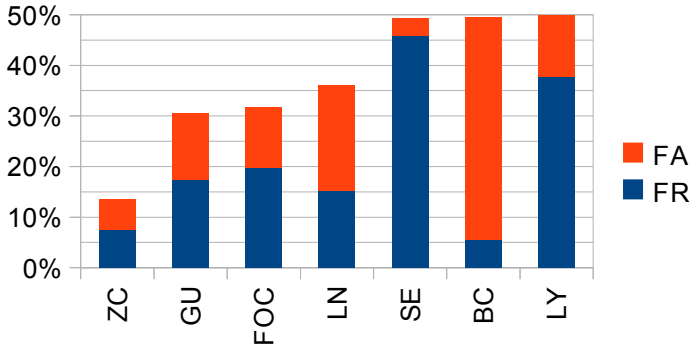


Figure 26: Results of methods for Berkeley data set.

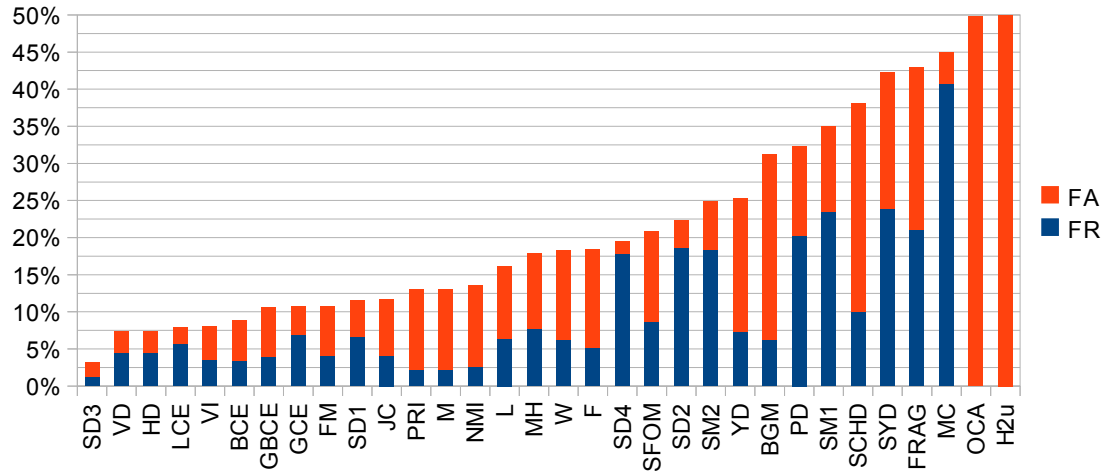


Figure 27: Results of segmentation-segmentation methods for Berkeley data set.

well as segmentations, were complex, and thus some methods were not able to evaluate them correctly (see figure 26). Method $SE$ (29) was one of the worst methods in contrast to previous evaluations. Quality of this method decreases with complexity of curvature of the borders. The only method, which was in all three evaluations between the best methods, was Zeboudj contrast (21). In the last evaluation it produced much less errors than the other methods. On the other hand, great influence of noise was shown in section 6.2. All images in all three sets include very low level of noise, thus the results of the method was not influenced by noise much. Other methods that kept the error rate at low level as well, were $FOC$ (19) and $GU$ (15).

Segmentation-segmentation methods can be evaluated only on data set consisting of more than one segmentation per image. Therefore, ABW and K2T were not suitable for such evaluation. Berkeley data set consists of multiple ground truth segmentations, thus all these implemented methods were tested on these segmentations. Number of evaluations for each method depends on symmetry of the method. Asymmetric methods evaluate each couple of segmentations twice but symmetric methods just once. Property of symmetry of each method is listed in table 3. Symmetric methods evaluated 3138496 couples of segmentations, while each asymmetric method processed 6276992 couples of segmentations. Since horizontal and vertical segmentations were not evaluated to each other, number of couples is lower than number of all couples. This task is easily parallelizable, still, computing of a single method could take tens of hours on common 4-core CPU.

Results of all implemented segmentation-segmentation methods are presented in figure 27. Order of methods correspond to smaller evaluation presented in [38]. All methods reaching 50% are practically unusable. The lowest error was made by method $SD_3$ (184).

It uses one-many correspondences and distance of border pixels instead of segment pixels. Other variants ($SD_1$, $SD_2$ and $SD_4$) were much worse, thus the change of specific parts leads to very different results, obviously. Since $VD$ (51) and $HD$ (45) has the same basis of the formula, the results are equal. Moreover, the method $VD$ (51) is a metric. Another metric with low error rate is $VI$ (109).

There is another interesting thing that could be seen in results. $LCE$ (41), as well as $SD$ (184), is tolerant to local refinement, while $BCE$ (42) is intolerant and $GCE$ (40) is tolerant to global refinement only. Evidently, tolerance to local refinement produces less errors than no tolerance and much less than tolerance to global refinement. Methods $FRAG$ (112) and $OCA$ (103) compute number of segments only. Their results are, therefore, very poor. They do not include topology and location of segments in a segmentation, which provide much more information than number of segments. Moreover, great difference in number of segments can occur in the same image.