Shape Classification According to LBP Persistence of Critical Points

Ines Janusch(✉) and Walter G. Kropatsch

Pattern Recognition and Image Processing Group,
Institute of Computer Graphics and Algorithms, TU Wien, Vienna, Austria
{ines,krw}@prip.tuwien.ac.at

Abstract. This paper introduces a shape descriptor based on a combination of topological image analysis and texture information. Critical points of a shape's skeleton are determined first. The shape is described according to persistence of the local topology at these critical points over a range of scales. The local topology over scale-space is derived using the local binary pattern texture operator with varying radii. To visualise the descriptor, a new type of persistence graph is defined which captures the evolution, respectively persistence, of the local topology. The presented shape descriptor may be used in shape classification or the grouping of shapes into equivalence classes. Classification experiments were conducted for a binary image dataset and the promising results are presented. Because of the use of persistence, the influence of noise or irregular shape boundaries (e.g. due to segmentation artefacts) on the result of such a classification or grouping is bounded.

Keywords: Shape descriptor · Shape classification · Local topology · Persistence · LBP · Local features

1 Introduction

We present in this paper a shape descriptor based on local topology and persistence in order to classify shapes. In prior work [7] we presented an approach to derive Reeb graphs using skeletons and local binary patterns (LBPs). One crucial decision when applying this approach is the choice of radius for the LBP computation. The degree of detail respectively noise (e.g. segmentation artefacts) that is captured by the representation depends on this radius. In order to analyse this influence of the parameter we conducted the following experiments: For the critical points of a skeleton (branching and end positions) all possible radii were tested. We can restrict the LBP analysis to these critical points as the parts of a skeleton in between critical points form a continuous curve representing a ribbon-like shape. These skeleton segments in between critical points correspond to edges in a Reeb graph and therefore do not represent any change in topology.

The experiments we conducted not only allow us insight into the impact of a change in the radius parameter but it also provides the basis for a new shape

N. Normand et al. (Eds.): DGCI 2016, LNCS 9647, pp. 166–177, 2016.
DOI: 10.1007/978-3-319-32360-2_13

descriptor. According to this descriptor a shape is represented by the evolution of LBP types for critical points of a skeleton and for varying LBP radii. This approach is therefore related to persistence as introduced by Edelsbrunner et al. [4]. In persistence, those features which persist for a parametrised family of spaces over a range of parameters are considered signals of interest. Short-lived features are treated as noise [5]. Such a range of parameters is for example given by succeeding scales as it is done for scale-space representations [10]. The persistence of a feature is given as its lifetime, that is the range of scales for which the feature is present. The approach presented is based on persistence over scales. We consider LBPs for ranges of radii and use the persistence of LBP types according to these radii as shape descriptors. Similar shape descriptors based on topological persistence are for example barcodes [1] or persistence diagrams [4]. The shape descriptor presented in this paper was inspired by a method to compute Reeb graphs. Nevertheless, for shape classification using a graph representation, graph matching is needed, which is not trivial and may be computationally expensive. A classification of shapes using the shape descriptor presented in this paper is done by comparison of feature vectors using the edit distance. As these feature vectors are only computed for a small number of characteristic positions within the shape, the number of feature vectors for each shape is limited. Thus, a shape classification based on the new descriptor can be done efficiently. These feature vectors are built as a scale-space representation. The sampling of the scale-space may of course limit the degree of detail of the representation. In the discrete space a complete sampling over all possible scales can be done. However, as the shape descriptor depends on this sampling of the scale-space it is not invariant to the size of the shape represented. For shape classification using shapes of varying sizes, a normalisation is needed beforehand.

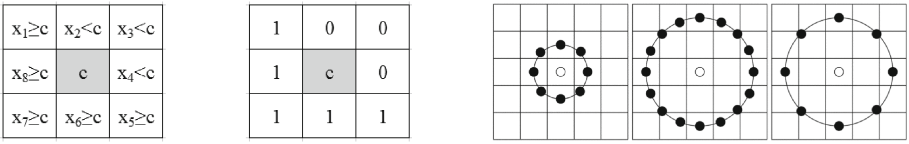
The rest of the paper is structured as follows: Sect. 2 gives a short summary of the theory of LBPs. The influence of the LBP parameters on our approach is discussed in Sect. 3. In Sect. 4 a new shape descriptor based on the variation of one such parameter is presented. Experimental results obtained for this shape descriptor are given in Sect. 5. Section 6 concludes the paper.

2 Introduction to LBPs

LBPs were first introduced for texture classification [11] and since became popular texture operators: To determine the LBP for a pixel, this pixel is compared to the subsampled neighbourhood around it. The according position in a bit pattern is set to 1 if the value of a neighbouring pixel is larger than or equal to the value of the center pixel and to 0 otherwise (Fig. 1a and b). The two parameters P and R may be adjusted for the LBP operator. P fixes the number of sampling points along a circle of radius R around the center pixel, for which the LBP operator is computed [12]. Figure 1c shows different parameter configurations.

The bit pattern encodes the local topology. According to the LBP, the center pixel can be classified as:

- (local) maximum (the bit pattern contains only 0s),



$x_1 \geq c$	$x_2 < c$	$x_3 < c$
$x_8 \geq c$	c	$x_4 < c$
$x_7 \geq c$	$x_6 \geq c$	$x_5 \geq c$

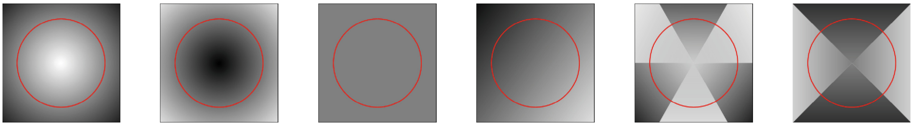
(a) comparison with neighbours

1	0	0
1	c	0
1	1	1

(b) bit pattern

(c) (P,R) = (8,1); (P,R) = (16,2); (P,R) = (8,2) - according to [12].

Fig. 1. (a) and (b) LBP computation for center pixel c and (c) variations of the parameters P (sampling points) and R (radius).



(a) max (b) min (c) plateau (d) slope (e) saddle (f) ridge

Fig. 2. Local topology encoded by an LBP at the radius marked in red (Color figure online).

- (local) minimum (the bit pattern contains only 1s),
- plateau (the bit pattern contains only 1s, but all pixels of the region have the same gray value),
- slope (the bit pattern of the region contains one connected component of 1s and one connected component of 0s - compare uniform patterns [12]),
- saddle point otherwise [6].

We further define a special case of saddles, the ridge to contain two connected components of 1s and two connected components of 0s. Figure 2 shows all these neighbourhood configurations.

3 Impact of the LBP Radius

In [7] we used a combination of LBPs and the medial axis to derive a graph representation of a binary image. The shape is first represented using a skeleton. Each skeleton pixel is then considered the center pixel of an LBP computation and the LBP type is determined. The LBP is computed for a radius R' that is larger than the radius r of the maximally inscribed circle stored for every medial axis pixel, in order to consider also the shape boundary: $R' = r + \epsilon$.

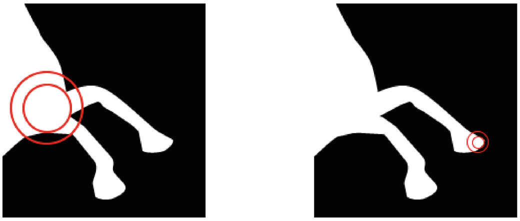
The graph representations obtained using this method depend strongly on the chosen LBP parameters. While adapting the number of sampling points P results in no or only minor changes of the graph, if the sampling density is high enough to capture the finest details, variations of the radius R have a strong influence on the resulting graph since many details inside the circle may be ignored.

The medial axis radius r of a certain skeleton pixel needs to be enlarged for the radius R' used in the LBP computation in order to take the shape boundaries into account (this also compensates for possible segmentation artefacts). Therefore, $R' = r + \epsilon$. In this process ϵ controls the radius and thus the level of detail that is captured or discarded. Enlarging ϵ may be compared to a smoothing operation along the shape's boundary. The impact on the resulting graph can be best described as graph pruning. Branches shorter than $r + \epsilon$ are discarded by this approach. However, how to choose factor ϵ is a crucial decision.

The choice of ϵ in general depends on the dataset. Moreover, ϵ should not be set to a fixed number of pixels for all skeleton pixels of the shape or even of the dataset. It should rather be defined as an adapting factor dependent on the medial axis radii. When varying the LBP radius, the LBP type of a skeleton pixel changes: For $\epsilon = 0$ the LBP sampling points are located along the maximally inscribed circle of the medial axis. Therefore, for a binary segmented image the according LBP is of type plateau. For a slightly increased radius the LBP in general equals the connectivity of the skeleton pixel (branching point, end point or ridge). For further increasing ϵ the LBP may change its type [2]. LBPs of the following types: plateau, slope, saddle and ridge may occur. The largest analysed radius is reached once an LBP of type maximum is observed. This configuration appears, once the whole shape is covered by the circle spanned by radius R'. For this maximal radius R'_{max} the following condition holds:

$$R'_{max} \leq \begin{cases} \frac{diameter}{2} & \text{for center of shape} \\ diameter & \text{everywhere} \end{cases} \tag{1}$$

Some examples of maximally inscribed circles with radius r and circles according to an enlarged radius R' are shown in Fig. 3. While the LBPs for maximally inscribed circles encode a plateau, the local topology according to the LBP of the larger circle shows a saddle (Fig. 3a) and a slope (Fig. 3b).



(a) plateau → saddle

(b) plateau → slope

Fig. 3. LBP for increasing radius captures evolution of local topology.

4 Shape Descriptor: Persistence of Critical Points

We further analyse the evolution of the LBP types for increasing radii R' starting with the medial axis radius r:

$$r \leq R' \leq diameter. \tag{2}$$

The evolution, respectively persistence of the LBP types of critical points of a skeleton (e.g. branching points and end points) can be described using a vector (LBP persistence vector) and a graph visualisation. We compute the LBP type for a range of radii and store it for every radius analysed. The persistence of such an LBP type is defined as its lifetime. In case of our persistence vector the lifetime is given as the length of an uninterrupted sequence of identical LBP type entries. We use the graph and the underlying LBP persistence vector as a shape descriptor and as a tool for shape classification.

For a binary shape, the approach presented in this paper proceeds as follows:

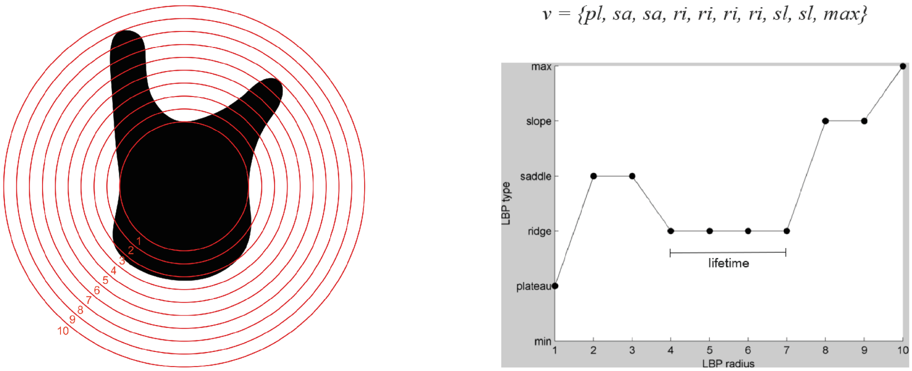
1. obtain topology preserving skeleton (using morphological thinning)
2. derive medial axis radii (Euclidean distance from boundary)
3. locate branching points of the skeleton
4. compute LBP persistence vector for each branching point
5. shape matching: compare LBP persistence vectors using the edit distance [9]

More details and properties of this approach are discussed in the following subsections.

4.1 LBP Persistence Vectors

We iteratively analyse the local topology captured by the LBP operator for the critical points using increasing LBP radii R'. We start with the radius r of the medial axis $R'_0 = r$ and increase the radius in each iteration. This computation terminates as soon as the circle spanned by radius R' covers the entire shape. Therefore, we start with an LBP of type plateau $R'_0 = r$ and stop at an LBP of type maximum. In between these two states the LBP type may alternate between ridge, saddle, and slope. An LBP of type minimum can not appear, as for this the LBP center pixel as well as all sampling points along the LBP radius would need to be in the background region, which is not possible since the LBP center pixel is a branching point of a skeleton and therefore always inside a shape.

For every critical point we store in the persistence vector the LBP type of every radius analysed. Thus, an LBP type of high persistence is captured as sequence of uninterrupted, repeated entries of the same LBP type. Alternatively the interval of radii for which one LBP type stays the same may be stored. In the persistence graph we assign the LBP types to one axis and plot the according radii along the second axis. Figure 4 shows a small toy example for a simple shape.



(a) Simple shape and some LBP radii marked in red.

(b) Persistence graph and LBP persistence vector v.

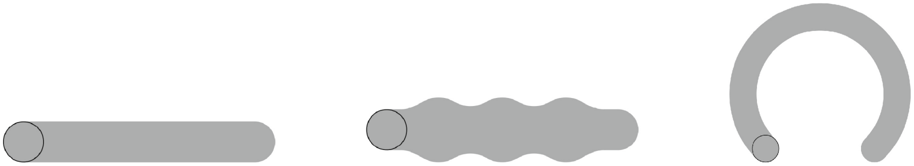
Fig. 4. Toy example for LBP persistence vector plotted as graph (Color figure online).

4.2 Critical Points of the Skeleton

Critical points of the skeleton (branching and end positions) represent changes in topology: the number of connected components changes when walking along a skeleton and passing a critical point. The skeleton segments in between these critical points correspond to topological persistence, the number of connected components does not change when moving along these segments. For these skeleton segments the LBP centered at a skeleton pixel is always of type ridge, upon discarding intersections of parts of the shape that are not in the same connected component as the center pixel for the LBP computation. Branching points thus serve as suitable locations within a shape to compute LBP persistence vectors. However, for some shapes no branching points will exist. In this case the skeleton is given as single curve with two endpoints. A skeleton given as a single point (medial axis of a perfect circle) or a closed curve (medial axis of a perfect ring) are special cases which are not considered here. For skeletons without branching points a representation of the shape is possible using LBP persistence vectors computed for the end points. The LBP persistence vectors then allow to deduct information about the curvature of the shape's boundary. Taking the sequence of LBP types for increasing radii along a branchless medial axis into consideration, bounds for the degree of curvature of the medial axis can be estimated (Fig. 5a and c). Furthermore, changes in thickness are detectable using the medial axis radius stored for the skeleton pixels (Fig. 5b). Shocks as introduced by [13] present a suitable alternative to analyse and represent such shapes.

4.3 Shape Classification

Since we compute the proposed shape descriptor (the LBP persistence vector) centered at the branching points of the skeleton, one for each branching point, we derive several shape descriptors representing one shape. We tested different



(a) straight medial axis, constant radius

(b) straight medial axis, alternating radius

(c) curved medial axis, constant radius

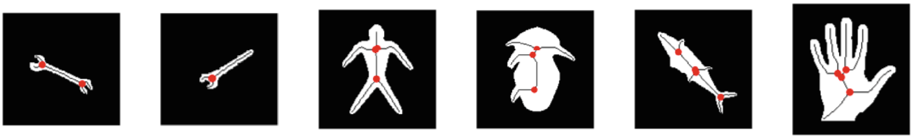
Fig. 5. Examples for shapes with no branching points in the skeleton.

approaches on how to choose from them or how to combine them in order to classify a shape in our experiments and present them in Sect. 5. However, the LBP persistence vectors corresponding to the different branching points of a single shape all individually cover the whole shape. Therefore, it is sufficient to choose just one of the branching points and its corresponding persistence vector as shape descriptor. Nevertheless, some branching points may be better suited as center of a shape descriptor (e.g. because they require less iteration to cover the whole shape). Future research may therefore be focused on methods and properties to judge the suitability of a branching point.

For classification of shapes the LBP persistence vectors are compared using the Levenshtein distance (edit distance) [9] as a measure of similarity which is also used by Sebastian et al. when comparing shock graphs [13]. Since every shape descriptor starts with an LBP of type plateau and ends with an LBP of type maximum, edit operations only alter the LBP types in between. These are of type saddle, slope or ridge. LBPs of these three types may alternate in an LBP persistence vector, therefore every edit operation generates a valid LBP persistence vector. The LBP type is stored as entry in the LBP persistence vector for every radius analysed, a sequence of identical entries represents an LBP type of high persistence. To transform such a vector to a sequence of alternating LBP types (therefore low persistence) many edit operations are needed. Thus, the information about the lifetime of one LBP type over a range of radii is taken into account when using the edit distance. In addition, this shape descriptor is robust in regard to small changes of the shape (e.g. small perturbations along the shape's contour), since such details do not persist over a large range of scales (radii) and thus have low persistence. These small changes therefore lead to changes of short sequences in the LBP persistence vector. Regarding a comparison using the edit distance, differences in such few entries are inexpensive and therefore the two shapes (with one of them showing small perturbations) will show high similarity.

5 Experiments

We tested the proposed shape descriptor on a subset of the Kimia 99 dataset [13]. For the subset used in our evaluation of shape classification we chose shape



(a) class1 (b) class2 (c) class3 (d) class4 (e) class5 (f) class6

Fig. 6. Representative image for each class in the dataset together with its skeleton and branching points marked in red (Color figure online).

classes for which at least four images are in the class. Furthermore, we did not use the partially occluded shapes of the dataset. Following this rules, our dataset contains 38 images, which are grouped into six classes holding four to eleven images each. Figure 6 shows one representative image together with its skeleton and branching points of the skeleton (marked in red) for all classes of the dataset[1].

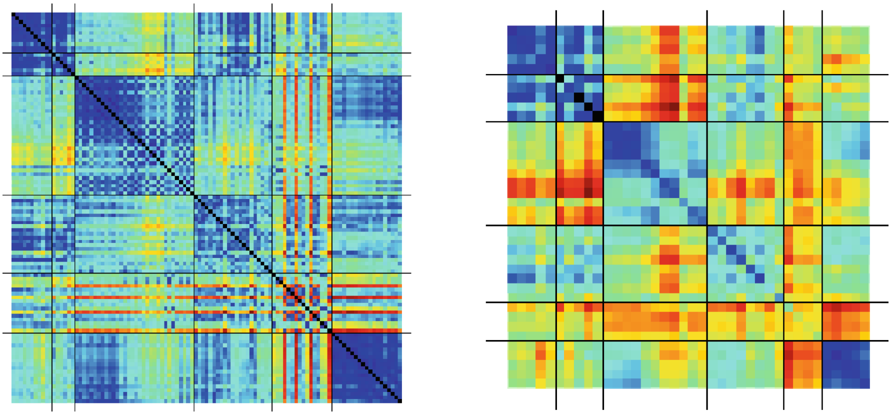
To evaluate the classification efficiency of our shape descriptor we performed a leave-one-out cross validation for each of the 38 shapes in the dataset and for the 105 LBP persistence vectors corresponding to the 105 branching points in the dataset. For every shape several descriptors (LBP persistence vectors), one corresponding to each branching point of the skeleton, may exist. The shape can be classified using one of these descriptors or a combination of them. We tested the following approaches:

1. according to the LBP persistence of only one branching point,
2. according to the average LBP persistence of all branching points of the shape,
3. according to the combined LBP persistence of all branching points of the shape using a majority vote,
4. according to the combined LBP persistence of all branching points of the shape using a weighted majority vote.

5.1 Classification Using Single LBP Persistence

Every image is classified by a comparison of the LBP persistence vector of only one of its branching points to all other LBP persistence vectors. For the dataset used in this experiment 105 branching points and respectively 105 LBP persistence vectors represent the 38 images. Figure 7a shows the confusion matrix for the distances between all 105 vectors. For every LBP persistence vector the class with the smallest edit distance, is chosen as resulting class. We evaluated this classification using leave-on-out cross validation for the 105 vectors. Only 15 out of 105 LBP persistence vectors were wrongly classified, 86 % were correctly matched. The confusion matrix in Fig. 7a shows that smallest distances correspond mostly to comparisons within one class for class 1, 3 and 6. This is

[1] The complete dataset with marked skeletons and branching points can be found at: prip.tuwien.ac.at/staffpages/ines/docs/dataset_dgci16.pdf.



(a) distances between all LBP persistence vectors

(b) average distances of the LBP persistence vectors of all images

Fig. 7. Confusion matrix: distances of the (a) 105 LBP persistence vectors and (b) 38 average LBP persistence vectors. Blue: low distances; red: high distances. The lines separate the individual classes: class1 to class6 from left to right and top to bottom (Color figure online).

well visible by the dark blue blocks along the diagonal. Especially for class 5 distances are in general high both for comparisons within and outside the same class. This may be due to the high variation of the shapes within this class.

5.2 Classification Using Average LBP Persistence

Here we compute the distance of one image of the dataset to all other images as the average of the edit distances of all LBP persistence vectors representing one image. We then classify every image using leave-one-out cross validation. Again the class of the image with the smallest distance is taken as classification result The confusion matrix for this experiment is shown in Fig. 7b. For classes 1, 3, 4 and 6 the smallest distances are found within the same class. This is again visible by the blue blocks along the diagonal. 8 of the 38 images were wrongly classified, 79 % of the images are correctly classified. The classification error is slightly higher in comparison with the method presented in Sect. 5.1.

5.3 Classification Using a Majority Vote for Combined LBP Persistence

For this experiment we do not combine the distances. We first determine for all LBP persistence vectors of one image the class of smallest distance (as it is done in Sect. 5.1). The final classification is obtained using a majority vote on these classes. In case no majority can be found a choice is made randomly. We again classified all 38 images of the dataset using a leave-one-out cross validation:

Table 1. Classification results using a majority vote (result1) and a weighted vote (result2). Wrongly classified samples are highlighted in bold font.

Image	img1	img2	img3	img4	img5	img6	img7	img8	img9	img10	img11	img12	img13
Class	1	1	1	1	1	2	2	2	2	2	3	3	3
result1	1	1	1	1	1	**1**	**1**	**1**	**1**	**1**	3	3	3
result2	1	1	1	1	1	2	**1**	**1**	**1**	**1**	3	3	3
Image	img14	img15	img16	img17	img18	img19	img20	img21	img22	img23	img24	img25	img26
Class	3	3	3	3	3	3	3	3	4	4	4	4	4
result1	3	3	3	3	3	3	3	3	4	4	4	4	4
result2	3	3	3	3	3	3	3	3	4	4	4	4	4
Image	img27	img28	img29	img30	img31	img32	img33	img34	img35	img36	img37	img38	
Class	4	4	4	5	5	5	5	6	6	6	6	6	
result1	4	**1**	**2**	5	5	5	5	6	6	6	6	6	
result2	4	**1**	**2**	5	5	5	5	6	6	6	6	6	

7 images were wrongly classified, thus 82 % of the dataset were correctly classified. Table 1 shows the classification results in row *result1*. All samples of class 2 were wrongly classified as class 1. This may be due to the quite high similarity of the shapes apart from symmetry (see Fig. 6). However, all samples of class 5, which has a high variation within the class, are correctly classified in contrast to the two classification procedures presented in Sects. 5.1 and 5.2.

5.4 Classification Using a Weighted Vote for Combined LBP Persistence

This classification procedure works similar to the one presented in Sect. 5.3. However, here the LBP persistence vectors of one image are not equally involved in the voting but have a weighted vote according to their distance values. We compute the weight as the ratio of the distance of one LBP vector to the sum of distances of all vectors belonging to one image. Smallest distances are associated with the highest weights. The final class of an image is determined as the class with the highest vote when summing up the weights. The 38 images of the dataset were classified using leave-one-out cross validation. The results are shown in Table 1 in row *result2*. 6 images were wrongly classified, a correct classification was obtained for 84 % of the dataset. This result is very similar to the one obtained using a majority vote, except for one shape, which was wrongly classified using the majority vote, but is correctly classified using the weighted vote.

5.5 Discussion of the Experiments

Because of multiple branching points in a skeleton, several shape descriptors may exist for one shape when using the presented approach. We tested four methods to choose one or to combine these shape descriptors when classifying a shape. Choosing just one shape descriptor based on a branching point of the shape randomly yields the best results as the shape descriptor of one branching point in any case takes the whole shape into account. Combinations of the descriptors seem to introduce noise and therefore reduce the representational power.

The experiments show the classification capability of the presented approach: 79 % to 86 % of the dataset (subset of the Kimia 99 dataset [13] - occluded shapes and single shapes were not considered) are correctly classified. The authors of [13] report a 100 % (respectively 87 %) correct shape classification when using their shock graphs and considering the top 3 (respectively top 10) matches. Using only the best match and a variation of the dataset, 86 % accuracy in shape classification are reached by our algorithm. However, the authors of [13] also mention the high computational time needed to classify a shape. The classification presented in this paper is very efficient as it is computed only for a small number of locations or even only one location, within the shape. Furthermore the approach presented does not analyse all points of a shape, but employs the efficient LBP approach and computes the shape descriptor only on a subset of the shape's points (given as subsampling of a circle with origin inside the shape).

6 Conclusion and Outlook

The presented approach uses the persistence of LBP types computed at characteristic positions within the shape to classify it. It therefore derives information about local topology and persistence using a texture operator at increasing scales. The feature vectors used for classification are given as the evolution respectively persistence of the LBP types as LBP persistence vectors. For each shape the computation is guided by a skeleton representation of the shape. LBP persistence vectors are only computed for a very limited number of locations - the branching points of the skeleton.

For future work we would like to further test our shape classification using partially occluded shapes. Besides the edit distance, used in the experiments of this paper, further distance measurements need to be tested. For the decision on critical points that are used in the shape descriptor computation, we will evaluate methods to judge the suitability of the critical points. We may for example consider the diameter of the shape that is based on the maximum eccentricity of a shape [8]. Branching points of the skeleton close to the boundary mainly represent small irregularities of the shape boundary, e.g. segmentation artefacts. This estimation of the position of branching points in the shape can be used to rate the importance of branching points according to Eq. 1. Thus, the set of critical points can be reduced to the most important ones.

Moreover, as LBPs are not limited to binary images, we would like to test an application of the presented approach on gray-scale images. An initial segmentation may still be needed to derive the skeleton which guides all further operations. However, to reduce the impact of segmentation artefacts the LBP computations may then be done on the original gray-scale data. Couprie et al. presented grayscale skeletons [3] that may be used as an alternative to an initial segmentation. Moreover, a gray-level image can be interpreted as a landscape according to the gray-values. Critical points of this landscape as they are computed by Cerman et al. [2], may be used as an alternative and a segmentation and skeleton representation may no longer be needed.

References

1. Carlsson, G., Zomorodian, A., Collins, A., Guibas, L.J.: Persistence barcodes for shapes. Int. J. Shape Model. **11**(02), 149–187 (2005)
2. Cerman, M., Gonzalez-Diaz, R., Kropatsch, W.: LBP and irregular graph pyramids. In: Azzopardi, G., Petkov, N., Effenberg, A.O. (eds.) CAIP 2015. LNCS, vol. 9257, pp. 687–699. Springer, Heidelberg (2015). doi:10.1007/978-3-319-23117-4_59
3. Couprie, M., Bezerra, F.N., Bertrand, G.: Topological operators for grayscale image processing. J. Electron. Imaging **10**(4), 1003–1015 (2001)
4. Edelsbrunner, H., Letscher, D., Zomorodian, A.: Topological persistence and simplification. Discrete Comput. Geom. **28**(4), 511–533 (2002)
5. Ghrist, R.: Barcodes: the persistent topology of data. Bull. Am. Math. Soc. **45**(1), 61–75 (2008)
6. Gonzalez-Diaz, R., Kropatsch, W.G., Cerman, M., Lamar, J.: Characterizing configurations of critical points through LBP. In: SYNASC 2014 Workshop on Computational Topology in Image Context (2014)
7. Janusch, I., Kropatsch, W.G.: Reeb graphs through local binary patterns. In: Liu, C.-L., Luo, B., Kropatsch, W.G., Cheng, J. (eds.) GbRPR 2015. LNCS, vol. 9069, pp. 54–63. Springer, Heidelberg (2015)
8. Kropatsch, W.G., Ion, A., Haxhimusa, Y., Flanitzer, T.: The eccentricity transform (of a digital shape). In: Kuba, A., Nyúl, L.G., Palágyi, K. (eds.) DGCI 2006. LNCS, vol. 4245, pp. 437–448. Springer, Heidelberg (2006)
9. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady **10**, 707–710 (1966)
10. Lindeberg, T.: Scale-space theory: a basic tool for analyzing structures at different scales. J. Appl. Stat. **21**(1–2), 225–270 (1994)
11. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recogn. **29**(1), 51–59 (1996)
12. Pietikäinen, M., Hadid, A., Zhao, G., Ahonen, T.: Computer Vision Using Local Binary Patterns. Computational Imaging and Vision, vol. 40. Springer, London (2011)
13. Sebastian, T., Klein, P., Kimia, B.: Recognition of shapes by editing shock graphs. In: IEEE International Conference on Computer Vision, vol. 1, p. 755. IEEE Computer Society (2001)