ROTATIONAL INVARIANT UNIFORM LOCAL BINARY PATTERNS FOR FULL 3D VOLUME TEXTURE ANALYSIS

J. Fehr

Chair of Pattern Recognition and Image Processing
University of Freiburg, Germany

ABSTRACT

In this paper, we present a novel method for the fast computation of rotational invariant ”uniform local binary patterns” (uLBP) for texture analysis on 3D volume data.
We introduce an alternative computation method for uLBPs in frequency space, which allows to effectively approximate uniform patterns in 2D and 3D in order to avoid the enormous computational complexity of a ”naive” 3D LBP implementation.

1. INTRODUCTION

”Local Binary Patterns” (LBP) [1] have been established as a standard feature based method for texture analysis in 2D images and were applied to a wide range of different applications. Various extensions to the basic LBP algorithms have been published in recent years, including rotational invariant and computationally efficient ”uniform binary patterns” (uLBP) - an overview can be found in [1].
In this paper, we extend the basic concept of LBPs for texture analysis on 3D volume data. Recently, 3D texture features have been successfully used for segmentation and classification of biological structures in volume images recorded from e.g. 3D confocal laser scanning microscopes (LSM) and other 3D imaging devices. The algorithms introduced in [2],[3] and [4] use rotational invariant, Haar-Integration based 3D texture features for different automatic biomedical image analysis tasks. The aim of this work is to adapt LBPs to these problems and evaluate the performance in comparison to the existing methods.
Previous experiments [4] have shown, that it is favorable to use rotational invariant features for the analysis of the given mostly anisotropic biological structures.

1.1. Related Work

So far, LBPs have only been applied to 2D images and 2D time series. There are several recent publications on ”volume local binary patterns” (vLBP)[5][6][7], but confusingly these methods deal with dynamic texture analysis on 2D time series and not on full 3D volumetric data. Respectively, vLBP only provide rotational invariance towards rotations around the z-axis.

1.2. Uniform LPBs

One commonly used extension of LBPs are the so called ”uniform” LBPs (uLBP), which were introduced in [8]. The uLBPs are a subset of all theoretically possible patterns. Experiments have shown that up to 90% of all patterns belong to this subset. Fig. 1 shows a set of uLBP templates. Due to the template nature, uLBPs are very suitable for our algorithm which computes uLBPs for 3D volume data in the frequency domain.
In the next section we will discuss the problems of a ”naive” LBP implementation in 3D. In the 3rd section we motivate our algorithm in a simplified 2D setting before describing the algorithm in detail in section 4.

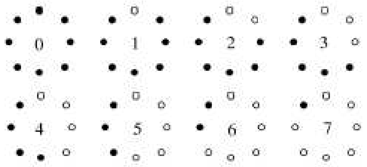

Fig. 1. Templates for uniform LBPs using 8 sampling points. Black points correspond to the binary value 1, white to 0

2. NAIVE IMPLEMENTATION OF LBPS IN 3D

At a first glimpse, the extension of LBPs to 3D seams to be straight forward: simply pick a voxel of the volume V as a center point $v \in V$, and sample a fixed number of n points $p_0 \dots p_{n-1}$ on the surrounding sphere S_r with radius r. Then one can compute $v - p_n$ for all points, and encode the binary pattern as in the usual LBP algorithm.
This appears to be very simple, but one has to face several severe problems following this direct approach: first, equidistant sampling on a sphere is a very hard task which is known as *Fejes Toth’s* problem. In general, it can not be

solved analytically. Since we need equidistant sampling in order to achieve full rotational invariance, we are limited to the few known point sets where a sampling is known [9] or use rather expensive numerical approximations. Second, rotational invariant LBPs require an order of the sampled points, this is trivial in 2D - but turns out to be a very hard problem on a sphere. And last, computational complexity becomes an issue with the vast rising number of sampling points needed on a sphere.

In order to overcome these challenges, we propose a new approach for the direct computation of uniform LBPs in frequency space, which avoids all the previous sampling and ordering issues in a computational effective way.

3. 2D ULBP COMPUTATION IN FREQUENCY SPACE

For the matter of simplicity, we first introduce the basic concept of the computation of uLBPs in frequency space for the 2D case. Please note, that we are not actually suggesting to use this method for 2D problems since the computational overhead is way to large, but we will derive the 3D case from the following algorithm later on.

Since we are about to compute the features in frequency space, we start of with a continuous setting: for a given center point $v \in V$ (in the 2D case V is of cause an image) we consider the gray-values which are ”touched” by the concentric circle C_r with radius r as a continuous function f. Then we then perform a fourier transform of f. Using fast convolution and so called ”circular harmonic” (CH) base functions, this transform can be computed every efficient and for all possible center points $v \in V$ simultaneously.

First, we precompute the CH base functions for the desired radius offline. For each band b of expansion, except the 0th coefficient, we split the complex coefficients into a real $(CH_b^{\Re})$ and a complex part $(CH_b^{\Im})$ (see fig. 2) using a parameterization around φ:

$$CH_b^{\Re} = \Re(e^{-ib\varphi}), CH_b^{\Im} = \Im(e^{-ib\varphi}) \qquad (1)$$

Given these base function templates, we can now expand every function f on the concentric circle around each $v \in V$ via piecewise fast convolution (denoted as $*$) of V with all CH templates. The band of expansion has to be chosen according to the desired number of samples and radius (see experiments). Hence we obtain a vector $\mathbf{v}^{CH}$ containing a series of CH coefficients of f for every center point v:

$$\mathbf{v}_{1\Re}^{CH} = V * CH_1^{\Re}, \mathbf{v}_{1\Im}^{CH} = V * CH_1^{\Im}, ... \qquad (2)$$

Since it is impossible to binarize the relation of the center v and its concentric neighbors in f in frequency space we approximate the binarization with an continuous operation: We subtract the value of v from f; since v is constant for every single expansion this operation is equal to a shift of the constant component on f and thus only effects the 0th component:

$$\mathbf{v}_0^{CH} := V * CH_0 - v \cdot b_{abs} \qquad (3)$$

where b_{abs} is the absolute number of expansion bands.

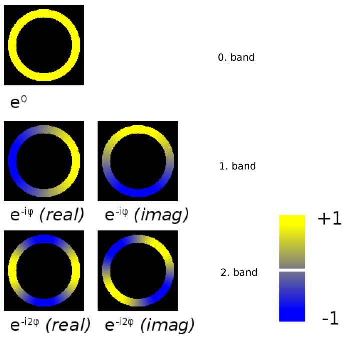

Fig. 2. ”Circular Harmonic” base functions of the first 3 bands.

We apply the same procedure to the set of continuous uLBP templates (see fig. 4) and obtain a CH expansion $\mathbf{uLBP}^{CH}$ for each of the n templates. Of cause this can be done offline in advance. Having the templates in frequency space representation, we are then computing the similarity of each $\mathbf{v}^{CH}$ with all n templates and chose the most similar one as uLBP representation at v. The Similarity s between two circular harmonic expansions can be measured in many ways. One common approach is to use the absolute value of the complex coefficients, which is invariant under rotation while neglecting the sometimes valuable phase information. We used a simple 2-norm to compare the differences of the absolute values of template and data coefficients:

$$s_{||2||} := \left[\sum_{i=0}^{b_{abs}} \left(\sqrt{\left(\mathbf{v}_{i\Re}^{CH}\right)^2 + \left(\mathbf{v}_{i\Im}^{CH}\right)^2} - \sqrt{\left(\mathbf{uLBP}_{i\Re}^{CH}\right)^2 + \left(\mathbf{uLBP}_{i\Im}^{CH}\right)^2} \right)^2 \right]^{\frac{1}{2}} \qquad (4)$$

Experiments (see section 6 for details) have shown, that this similarity measure is sufficient to approximate the correct uniform pattern in almost all cases, when the maximum band of CH expansion was chosen high enough.

It has to be noted, that is very crucial to normalize the template and data coefficients appropriately, e.g. by dividing the template coefficients by the sum over template.

4. FAST ULBP COMPUTATION IN 3D

Now we derive fast uLBPs (fuLBPs) for 3D volume data. Just as in the previous 2D case, will compute the features in the frequency domain. The only difference is, that we are now considering functions on a sphere instead of circles. Analog to the CH, we use the well established Spherical Harmonic transform (SH) [10], which forms an orthogonal basis on the 2-Sphere.

4.1. Spherical Harmonics

Using SH coefficients, every spherical function can be represented by the sum of its harmonics:

$$f(\theta,\phi) = \sum_{l=0}^{\infty}\sum_{m=0}^{l} a_{lm} Y_l^m(\theta,\phi) \tag{5}$$

where l denotes the band of expansion, m the number of components for the l-th band and a_{lm} the harmonic coefficient. The harmonic base functions $Y_l^m(\theta,\phi)$ are calculated as follows:

$$Y_l^m(\theta,\phi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}\cdot P_l^m(\cos\theta)e^{im\phi} \tag{6}$$

with the associated Legendre polynomial P_l^m.

Note that in this formulation we take advantage of the symmetry in the harmonic representation, neglecting the negative coefficients. For practical reasons we also split the base components into their real and imaginary parts following the notation $Y_l^{m\Re}$ and $Y_l^{m\Im}$ respectively. Fig. (3) shows the first few spherical harmonics.

The transformation $\mathbf{v}_{(l,m)}^{SH}$ of the original volumetric data V into the harmonic domain is easily computed via fast convolution:

$$\mathbf{v}_{(l,m)}^{SH} = Y_l^{m\Re}(\theta,\phi) * V + Y_l^{m\Im}(\theta,\phi) * V \tag{7}$$

where $*$ denotes a convolution in Euclidean space and $Y_l^m(\theta,\phi)$ a spherical harmonic base component. Given the SH expansion $\mathbf{v}_{(l,m)}^{SH}$ in every center point, the 3D algorithm follows analog to the 2D case. We also use the same similarity measure as the 2D case (4).

4.2. Data depended uniform patterns

It turns out, that besides all the nice properties discussed so far, our method suffers from one major pitfall: the number of possible uniform patterns (templates) in the 3D case is drastically larger than in the 2D case. On a circle, the uLBPs are clearly defined as all patterns which have one or no binary transition (zero to one) - see fig. 1. This definition would be way to limiting in the 3D case. On the other hand, alternative uLBP definitions would produce very large sets of templates.

To overcome this problem, we suggest an adjusted process. Instead of having a fixed global set of uLBPs, which would be extremely large in the 3D case, we propose to choose the set of uLBPs dependent on the input data. Building a ”code-book” of typical templates for every new task has the advantage that only a small set of templates is needed and that we can focus on the most discriminative patterns for the given classes.

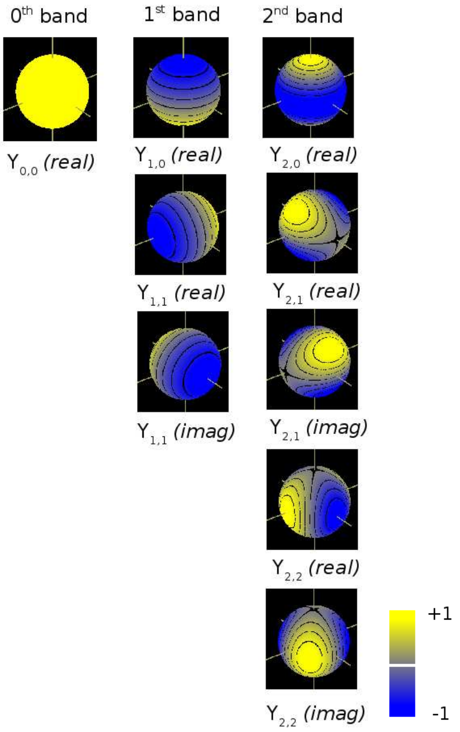

Fig. 3. Spherical Harmonic base functions of the first 3 bands.

The ”code-book” can be precomputed offline. After extracting the SH coefficients $\mathbf{v}_{(l,m)}^{SH}$ from sample data, we use k-means clustering with our similarity measure to find the k first uLBP templates - a representative of each cluster is then binarized and stored in the ”code-book”.

5. EXPERIMENTS

In order to validate our approach we performed a series of different experiments: first, we prove that uLBP can be approximated via computations in the frequency domain. Then we test the computational complexity of our approach before we evaluate the method on real 3D volume data.

5.1. Proof of Concept

In order to show that it is possible to perform the uLBP computation in the frequency domain, we first validated our method on the the uLBP templates. Fig. 4 shows the continuous equivalents to the templates 1,2 and 3 (compare to Fig.1) which were transformed to CH domain and then reconstructed by the inverse transformation with different bands of expansion. Note, that due to the later convolution, we use a cyclic translation of the actual circle. We also performed these experiments for the 3D case, but these results are hard to visualize. 3D results were analog to the 2D case. Fig. 4 clearly indicates, that the correct choice of the extension band is crucial, e.g. a 5-band expansion does not have the appropriate high frequencies in order to reconstruct template 1. On the other hand, the computational complexity is increasing linearly with the number of bands. We further investigated this trade-off while simultaneously evaluating the performance of our similarity measure: We rotated a set of templates in various angles around the center point and compared the approximated templates to the unrotated version. Figures 5 through 7 show several important properties of our method: first, the band of expansion has to match the number of samples for the 2D case - in 3D, the band should match the maximum number of samples which can be found on a single circle bound to the surface of the sphere. Second, the number of samples of an uLBP which can be approximated by our method is limited by the radius. And last: our method works perfectly for radii $>= 4$, but has some problems with smaller radii - this is due the fact, that for small radii the discrete approximation of circles/spheres causes problems.

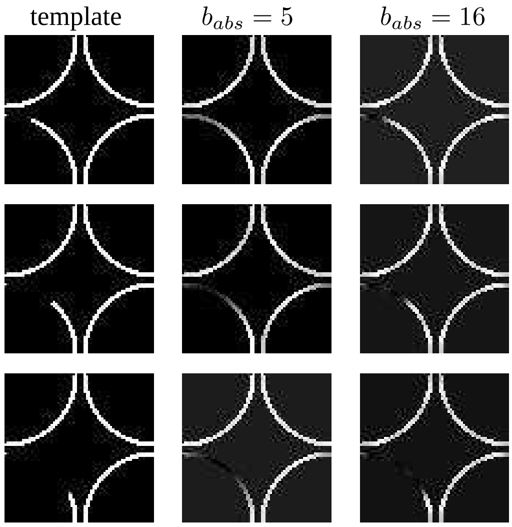

Fig. 4. **Left:** continuous templates 1,2 and 3 (from top down) for 16 sample uLBP. **Center:** reconstruction using base functions up to band 5.**Right:** reconstruction using base functions up to band 16.

In a final experiment for the concept proof, we directly evaluated our method with 2D uLBP results (we did this experiment only for 2D, because as mentioned before, uniform patterns are not defined for the 3D case). Fig.8 shows that

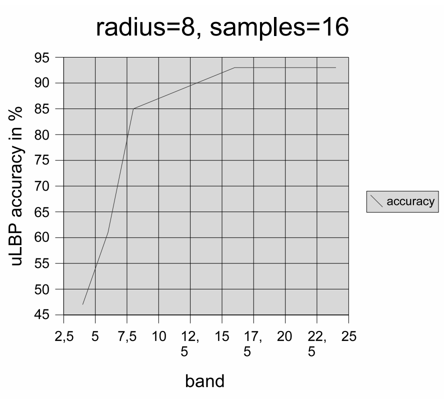

Fig. 5. Experiments with 2D templates with a fixed radius of 8 and 16 sampling points show an increas of accuracy until the band reaches the number of sampling points.

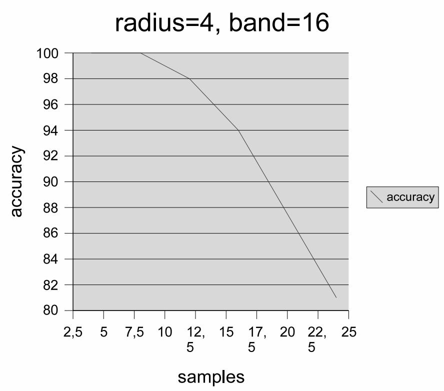

Fig. 6. Experiments with 2D templates with a fixed radius of 4 and an expansion to the 16th band indicate that the accuracy is not only limited by the band of expansion, but also by the number of samples for a given radius.

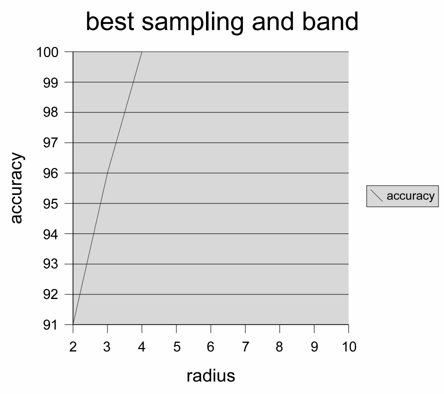

Fig. 7. Given the best possible combination of sampling rate and expansion band, the approximation has some problems for small radii while uLBP for lager radii can be computed without loss.

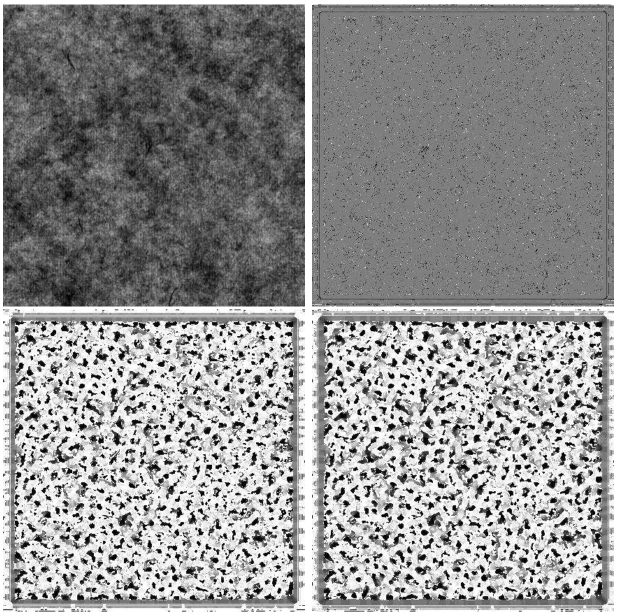

Fig. 8. Direct comparison of 2D uLBPs and the approximation in the frequency domain. **Top left:** original 2D texture image (paper texture). **Top right:** difference image. **Bottom left:** uLBP response with radius 8, 16 samples. **Bottom right:** approximation with expansion to the 16th band.

our method is able to approximate uLBPs fairly good - even though there are some errors shown in the difference image. By evaluating the histograms, we could see that the error is evenly distributed among all template patterns. As indicated by the experiments before, we found that the error is decreasing with an increase in radius.

For the 3D case one has to expect larger errors: due to the problems of discrete representations of spheres and the increase in coefficients more errors are likely to occur.

5.2. Complexity

In theory, the complexity of our method, as well as the original uLBP, can be given directly. Regardless the dimensionalty, for given N data points and s samples, uLBP has a complexity of $O(N \cdot s^2)$ were s samples have to be evaluated at all N points; finding the right pattern takes another $O(s)$. While our approach lies in $O((NlogN) \cdot (b+1) + N \cdot b^2)$, $O((NlogN)$ for the fourier transform of b bands plus the cost of finding the right pattern. So, for 2D, where $b = s$, our method is just to expensive - but in 3D, where $s >> b$ the additional overhead for the fourier transform will pay off drastically.

In practice, it turns out that the actual complexity is strongly influenced by some linear terms, which are not considered by the O-notation. The following tables give some results for experimental complexities based on a 2.8 GHz P4 mashine. In the 3D case, we used naive standard LBPs for the comparison with our fuLBPs.

radius/samples=band	uLBP 2D	fuLBP 2D
4/16	4s	7s
8/16	8s	8s
16/32	16s	23s

Fig. 9. Complexity for the computation on a 256^2 image.

radius/samples/band	LBP 3D	fuLBP 3D
4/24/6	10m 27s	1m 8s
4/124/8	24m	1m 40s
8/124/16	-	5m 31s

Fig. 10. Complexity for the computation on a 256^3 volume.

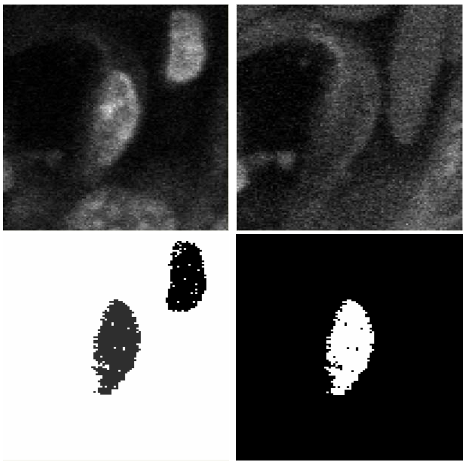

Fig. 11. Sample database entry, xy-slices of 3D volumetric data. From left to right: YoPro marker, Cy3 marker, ground truth labeling of the cell nuclei, binary mask for the database entry.

5.3. 3D Volume Data

We evaluated the texture analysis performance of the fuLBPs on 3D volumetric biological data and compared the results the methods presented in [11]. A database of 229 3D volume datasets of 3 different classes of cell-nuclei was given. The cells were recorded in tissue via confoncal laser microscopy using two different anti-body markers, YoPro and Cy3, which were recorded in separate channels. For this experiment we used only the YoPro channel. A sample database entry is shown in Fig. 11, please refer to [11] or [2] and [3] for further details on the database.

We used 12 different features of varying radii, number of samples and expansion bands. For each radius a separate

Celltype	result in [11]	fuLBP
Erythrocyte	93,3%	88,7%
Endothelial cell	84,6%	75,8%
Fibroblast	79,8%	74,2%
Background	94,1 %	90,9

Fig. 12. Rersults of the nuclei classification comparing the Haar-Intergral based fetures from [11] with fuLBPs.

code-book of size 10 was generated via k-means clustering. After feature extraction, we performed a voxel-wise classification via support-vector machine (SVM) following the algorithms in [11], [2] and [3]. Results are shown in table 12.

5.4. Results

The experiments for the 2D case clearly indicate that our proposed method is able to approximate uLBPs very well. The accuracy of our approach clearly depends on the radius and the band of expansion - while there are some problems with very small radii, mid and large scale radii fuLBPs work very well when the CH/SH transforms are expanded to bands corresponding to the number of samples and the radius.

For the 3D case, our method is significant faster than a naive uLBP implementation, even when a expansion to high bands is needed.

First experiments with 3D fuLBPs on the biomedical datasets indicate that this new method is competitive, although fuLBPs were outperformed by the Haar-Intergral based features from [11]. The reason for this may be manifold: first, the Haar-features are not gray scale invariant, just robust to nonotonic gray scale changes - it might turn out, that after all a true gray scale invariance is not favorable for the given task. Second, uLPBs have a larger spacial resolution, but a weaker gray scale resolution than Haar-features. And third, which is probably the most important point, the continuous feature space of the Haar-features is easier to handle for the SVM than the discrete space generated by the uLBPs.

There is still plenty of room for further improvement of the presented approach: one could increase the size of the code-book, or even more important, design a better similarity measure.

6. REFERENCES

[1] T. Mäenpää and M. Pietikäinen, ," in *Handbook of Pattern Recognition and Computer Vision*, CH. Chen and PSP. Wang, Eds., chapter Texture analysis with local binary patterns., pp. 197–216. World Scientific.

[2] O. Ronneberger and J. Fehr, "Voxel-wise gray scale invariants for simultaneous segmentation and classification.," in *In Proceedings of the 27th DAGM Symposium, in number 3663 LNCS, Springer, Vienna, Austria,*, 2005.

[3] Burkhardt H. Fehr, J., "Phase based 3d texture features," in *in Proceedings DAGM 2006*. vol. LNCS 4174, pp. 263–272, Springer.

[4] C. Sauer, J. Fehr, O. Ronneberger, H. Burkhardt, K. Saudau, and H. Kurz, "Automated identification of large cell numbers in intact tissues - self-learning segmentation, classif ication, and quantification of cell nuclei in 3-d volume data via voxel-based gray scale invariants," .

[5] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using local binary patterns with an application to facial expressions.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, in press., 2007.

[6] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using volume local binary patterns.," in *Proc. ECCV 2006 Workshop on Dynamical Vision*, Graz, Austria, 2006, p. 12 pp.

[7] G. Zhao and M. Pietikäinen, "Improving rotation invariance of the volume local binary pattern operator.," in *Proc. IAPR Conference on Machine Vision Applications*, Tokyo, Japan, 2007, pp. 327–330.

[8] T. Mäenpää, T. Ojala, M. Pietikäinen, and M. Soriano, "Robust texture classification by subsets of local binary patterns.," in *Proc. 15th International Conference on Pattern Recognition, Barcelona, Spain*, 2000.

[9] H. Cundy and A. Rollett, *Mathematical Models*, Oxford Univ. Press, 2nd ed., 1961.

[10] H. Groemer, *Geometric Applications of Fourier Series and Spherical Harmonics*, Cambridge University Press, 1996.

[11] J. Fehr, H. Kurz, C. Sauer, and H. Ronneberger, O.and Burkhardt, "Identifikation von zellen in intaktem gewebe - selbst-lernende segmentierung und klassifikation von zellkernen in 3d volumendaten mittels voxel-weiser grauwertinvarianten," in *Handels H. Ehrhardt J. Editors, Informatik Aktuell, Bildverarbeitung fr die Medizin 2006, Hamburg 19. - 21.3.06*. pp. 368–373, Springer-Verlag.