Planar Pixelations and Shape Reconstruction

Brandon Rowekamp*

Abstract

Given a PL (piecewise linear) set S in the plane $\mathbb{R}^2$ we consider the set $P_\varepsilon(S)$ consisting of all pixels of size ε that touch S. This pixelation $P_\varepsilon(S)$ resembles the original set, but may not well approximate various important invariants of the original set such as Betti numbers, perimeter, curvature measures. We describe an algorithm that associates to the pixelation $P_\varepsilon(S)$ a PL-set $\mathcal{P}_\varepsilon(S)$ which approximates S in a very strong sense.

1 Introduction

The ε-pixelation of the Euclidean plane is the decomposition determined by the lines

$$x \in \varepsilon\mathbb{Z} \text{ and } y \in \varepsilon\mathbb{Z}.$$

An ε-pixel is a square of the form

$$[\varepsilon(i-1), \varepsilon i] \times [\varepsilon(j-1), \varepsilon j], \quad i, j \in \mathbb{Z}$$

with center located at

$$C[i,j] := \Big(\frac{2i-1}{2}\varepsilon, \frac{2j-1}{2}\varepsilon\Big) \tag{1.1}$$

For any compact subset of the plane we define its ε-pixelation to be the union of all the ε-pixels that touch S. We denote it by $P_\varepsilon(S)$.

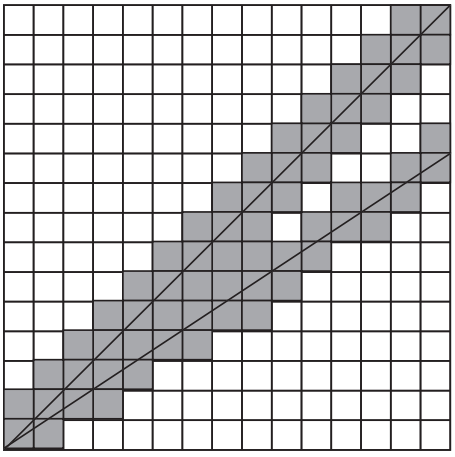

Figure 1: A pixelation of an angle

Roughly speaking, the main goal of this paper is to algorithmically associate to $P_\varepsilon(S)$ a planar PL-region $\mathcal{P}_\varepsilon(S)$ that approximates S very well as $\varepsilon \searrow 0$. More specifically, we would like to recover in the limit basic geometric and topological invariants of S such as, area, perimeter, curvature (measures) and Betti numbers.

While $P_\varepsilon(S)$ converges to S in the Hausdorff distance, this notion of convergence fails to recover even the most stable of invariants. For example lengths from the pixelation may not converge to the corresponding lengths in the original set. Additionally, topological information such as the Betti numbers may be lost at all resolutions ε.

For example, in Figure 1 we have depicted a pixelation of the angle A formed by two segments of slopes $\frac{2}{3}$ and 1 that have a common endpoint at the origin. The homotopy type of this pixelation is independent of the size of the pixel, and as seen in Figure 1, $b_1\big(P_\varepsilon(A)\big) = 2$, $\forall \varepsilon > 0$. The situation with geometric invariants such as perimeter or curvature is much worse (even in the case of a line the total curvature will explode while the perimeter will not converge to the correct length). Therefore the pixelation itself is not a reasonable approximation of the original shape. The goal of this paper is to generate algorithmically a better approximation using only information from the pixelation.

2 Basic Results

Proofs for all claims in this section are omitted due to lack of space. They are publicly available on the author's website[1].

Given the ε-pixelations of a PL subset of the plane, we would like to construct a sequence of PL approximations of the original set which converge in a strong sense to the original set. We delay a precise definition of the notion "strong convergence" until the main result. For now it suffices to say that we seek an approximation which recovers the homotopy type of the original set, as well as geometric invariants such as the perimeter, area and the curvature measures of the boundary.

As we have seen in the introduction, the pixelation itself will not recover these invariants. A better approximation is described explicitly in the Algorithm section given that the original set is PL[2]. The remainder of

*Department of Mathematics, University of Notre Dame, browekam@nd.edu

[1] http://www.nd.edu/~browekam

[2] We are currently working to extend this technique to semi-

this section describes results about pixelations which motivated the approximation which is described later.

Throughout this discussion we will use the concepts of *column* and *stack*. The column of a pixelation at the x-value x_0 is simply the union of pixels from $P_\varepsilon(S)$ which intersect the line $\{x = x_0\}$. A stack is a connected component of a column (that is to say, a series of pixels "stacked" on top of each other with no gaps).

An *elementary set* is a region of the following form:

$$\{(x, y) \in \mathbb{R}^2 : x \in [a, b], \beta(x) \leq y \leq \tau(x)\}.$$

where β and τ are continuous piecewise C^2 functions defined on $[a, b]$ with the property that $\beta(x) \leq \tau(x)$ for all $x \in [a, b]$.

Proposition 1 *If S is an elementary set, then it is contractible and $P_\varepsilon(S)$ is also contractible for every resolution ε.*

Therefore, the pixelation of an elementary set has the same homotopy type as the elementary set itself. We will approximate elementary sets by connecting points along the tops and bottoms of the stacks that make up their pixelations. To ensure convergence of perimeter and total curvature we will connect points from about every σ-th column, where σ is a number determined by ε (this number is explicitly shown in the algorithm). If σ satisfies certain constraints (see (3.1)), then this method of approximation will recover the desired invariants.

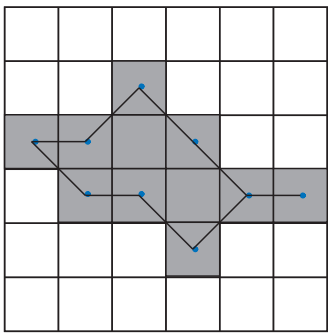

Figure 2: This approximation of an elementary set is the region outlined by the black line.

As suggested by Figure 1, for non elementary sets there is no guarantee of convergence in homotopy type. Therefore for the approximation of a general piecewise linear set we must determine which cycles in the pixelation come from the original shape, and which cycles are artifacts of the pixelation process. Intuitively these fake cycles must be located very close to columns that undergo a change in the number of their stacks (since fake cycles will have small area). We define the function $\boldsymbol{n}_\varepsilon : \mathbb{R} \setminus \varepsilon\mathbb{Z} \to \mathbb{Z}_{\geq 0}$, where

$$\boldsymbol{n}_\varepsilon(x_0) = \#\text{ of stacks of } P_\varepsilon(S) \text{ in the column at } x_0.$$

A point $x_0 \in \mathbb{R} \setminus \varepsilon\mathbb{Z}$ is called a *jumping point* of $\boldsymbol{n}_\varepsilon$ if

$$\boldsymbol{n}_\varepsilon(x_0 + \varepsilon) \neq \boldsymbol{n}_\varepsilon(x_0).$$

A column over a jumping point is called a *jumping column*. From investigating examples of pixelations, we expect topological noise to occur "near" jumping points. The following theorem gives a precise sense of how "near" topological noise must be to jumping points.

Proposition 2 *If S is a generic piecewise linear set, i.e., no two of its vertices lie on the same vertical line. Then the following hold.*

1. *There is an integer $k = k_S$ depending only on S such that any fake cycles of $P_\varepsilon(S)$ is within at most k-columns from a jumping column.*

2. *The function $S \ni (x, y) \overset{h}{\mapsto} x \in \mathbb{R}$ is a stratified Morse function in the sense of [7] and for any jumping point x_0 of $\boldsymbol{n}_\varepsilon$ there exists a critical value x_0^h of h such that $|x_0 - x_0^h| < k\varepsilon$.*

In practical terms this proposition states that the fake cycles only occur in narrow vertical strips of the plane containing the jumping points of $\boldsymbol{n}_\varepsilon$. Moreover the jumping points of $\boldsymbol{n}_\varepsilon$ cannot be too far from the critical values of the function h. Thus we can assume that any cycle which occurs close to a jumping column is fake and so our approximation should fill it in (the meaning of "close" here will be explicitly shown in the algorithm). We can fill these fake cycles somewhat carelessly, since they take up a very small area of the plane for small resolutions. We call the columns which can contain fake cycles "noise" and cover each connected component of $P_\varepsilon(S)$ within in these columns by rectangles.

So far we have a way to approximate two situations. The first is for elementary regions, and the second is for noise columns that accumulate near the critical values of h. Away from the critical values of h the set S is a disjoint union of elementary sets. Therefore these two approximation techniques suffice to approximate the entire set.

The next section gives an explicit algorithm for generating an approximation of a PL set from its pixelations.

3 The algorithm

The input for this algorithm is a pixelation. We encode a pixelation by an $m \times m$ matrix A with 0, 1 entries,

algebraic subsets of the plane.

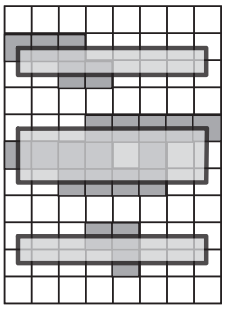

Figure 3: An example of approximating in noise columns (near jumping points of $\boldsymbol{n}_\varepsilon$).

where $a_{ij} = 1$ if and only if the pixel with center $C[i,j]$ belongs to our pixelation. We think of the parameter m as defining a $m \times m$ subdivision of a computer screen, consisting of squares of size $\varepsilon := \frac{1}{m}$. Define

$$\sigma(\varepsilon) = \lfloor m^\rho \rfloor,$$

where ρ is a fixed rational number $\rho \in (\frac{1}{2}, 1)$. Note that

$$\lim_{\varepsilon \searrow 0} \varepsilon(\sigma(\varepsilon))^2 = \infty, \quad \lim_{\varepsilon \searrow 0} \varepsilon\sigma(\varepsilon) = 0. \tag{3.1}$$

We denote by $P(A)$ the pixelation determined by the matrix A. The output of the algorithm will be a PL set $\mathcal{P}_\varepsilon(A)$ that decomposes in a canonical fashion as a finite union of trapezoids with vertical bases. We will refer to such regions as *polytrapezoids*. We allow for degenerate trapezoids, such as points, segments, or triangles.

The algorithm uses several basic subroutines. The first one is the the subroutine stack. Its input is a list

$$C = C_1, \ldots, C_m, \;\; C_i = 0, 1,$$

which encodes a column of the pixelation $P(A)$. The output of stack is a list of nonnegative integers

$$\boldsymbol{n}(C); \;\; b_1 \le t_1 < b_2 \le t_2 < \cdots < b_{\boldsymbol{n}(C)} \le t_{\boldsymbol{n}(C)},$$

where $\boldsymbol{n}(C)$ is the number of stacks in the column encoded by C, and the location of the bottom and top pixel in the j-th stack is determined by the integers b_j, t_j. More formally

$$C_k = 1 \Longleftrightarrow \exists 1 \le j \le \boldsymbol{n}(C) : \;\; b_j \le k \le t_j.$$

If $C = C_i$, the i-th column of $P(A)$, i.e.,

$$C_i = a_{i,1}, \ldots, a_{i,m}$$

then we will denote the output $\mathsf{stack}(C_i)$ by

$$\boldsymbol{n}_i, \;\; b_{i,1} \le t_{i,1} < \cdots < b_{i,\boldsymbol{n}_i} \le t_{i,\boldsymbol{n}_i}.$$

A number $1 \le i \le m-1$ is called a *jump point* if

$$\boldsymbol{n}_i \neq \boldsymbol{n}_{i+1}.$$

The next subroutine that we need is called jump. Its input is an integer $k \in [1, m)$ and the output is an integer $j_k = \mathsf{jump}(k)$ defined as follows. If

$$\big\{ i \in [k, m) \cap \mathbb{Z}; \;\; i \text{ is a jump point} \big\} = \emptyset,$$

then we set

$$\mathsf{jump}(k) := m + 1.$$

Otherwise

$$\mathsf{jump}(k) = \min\big\{ i \in [k, m) \cap \mathbb{Z}; \;\; i \text{ is a jump point} \big\}.$$

The noise region is determined by a finite collection of intervals

$$[\ell_1, r_1], \ldots, [\ell_\alpha, r_\alpha] \subset [1, m]$$

where the integers ℓ_k, r_k are determined inductively as follows.

$$\ell_1 = \max\big(\mathsf{jump}(1) - 2\sigma(\varepsilon), 1 \big),$$

$$r_1 = \min\big(m, \mathsf{jump}(1) + 2\sigma(\varepsilon) \big).$$

Suppose that $\ell_1, r_1, \ldots, \ell_j, r_j$ are determined. If $\mathsf{jump}(r_j) > m$ we stop. Otherwise we set

$$\ell_{j+1} = \max\big(\mathsf{jump}(r_j) - 2\sigma(\varepsilon), 1 \big),$$

$$r_{j+1} = \min\big(m, \mathsf{jump}(r_j) + 2\sigma(\varepsilon) \big).$$

The intervals $[\ell_1, r_1], \ldots, [\ell_\alpha, r_\alpha]$ may not be disjoint, but their union is a *disjoint* union of intervals

$$[a_1, b_1], \ldots, [a_J, b_J], \;\; b_i < a_{i+1}.$$

The intervals $[a_j, b_j], 1 \le j \le J$ are the *noise intervals*. The intervals

$$[1, a_1], [b_1, a_2], \ldots, [b_{J-1}, a_J], [b_J, m]$$

are the *regular intervals*.

The heart of the algorithm consists of two procedures, one for dealing with the noise intervals and the other for dealing with the regular intervals. These procedures will return a number of polytrapezoids,

First some notation. Given a collection of points

$$B_0, T_0, \ldots, B_N, T_N \in \mathbb{R}^2$$

such that

$$x(B_i) = x(T_i), \;\; y(B_i) \le y(T_i), \;\; \forall i = 0, \ldots, N,$$

$$x(B_{j-1}) < x(B_j), \;\; \forall 1 \le j \le N,$$

we denote by $\mathsf{polygon}(B_0, T_0, \ldots, B_N, T_N)$ the region surrounded by the simple closed PL-curve obtained as the union of line segments

$$[B_0, B_1], \ldots, [B_{N-1}, B_N],$$

$$[B_N, T_N], \ldots, [T_1, T_0], [T_0, B_0].$$

Note that each of the quadrilaterals $B_{i-1}, B_i, T_i, T_{i-1}$ is a (possibly degenerate) trapezoid with vertical bases.

Consider first the regular intervals. Given a regular interval $I := [p, q]$ we observe that the number of stacks $\boldsymbol{n}_i$ is independent of $i \in [p, q]$. We denote this shared number by $\boldsymbol{n} = \boldsymbol{n}(I)$.

We construct inductively a sequence of numbers $i_0 < \cdots < i_N$ as follows:

- We set $i_0 = p$.
- If $q - p < 2\sigma(\varepsilon)$ we set $N = 1$ and $i_1 = q$.
- If $i_0, \ldots, i_k$ are already constructed, then, if $q - i_k < 2\sigma(\varepsilon)$ we set $N = k + 1$ and $i_{k+1} = q$, else $i_{k+1} = i_k + \sigma(\varepsilon)$.

Note that if $q - p > \sigma(\varepsilon)$, then $N \geq 1$, $i_0 = p$, $i_N = q$ and

$$N = 1 \text{ if } q - p < \sigma(\varepsilon).$$

We have

$$\mathsf{stack}(C_{i_k}) = \boldsymbol{n}, \;\; b_{i_k,1}, t_{i_k,1}, \ldots, b_{i_k,\boldsymbol{n}}, t_{i_k,\boldsymbol{n}}.$$

For $j = 1, \ldots, \boldsymbol{n}$, and $k = 0, \ldots, N$ we denote by $B_{k,j}$ the center of the ε-pixel corresponding to the element entry $b_{i_k,j}$ in the column C_{i_k}. Similarly we denote by $T_{k,j}$ the center of the pixel corresponding to the entry $t_{i_k,j}$ of the column C_{i_k}. For $1 \leq j \leq \boldsymbol{n}(I)$, we set

$$\mathcal{P}_j(I) := \mathsf{polygon}(B_{0,j}, T_{0,j}, \ldots, B_{N,j}, T_{N,j}).$$

Define

$$\mathcal{P}(I) = \bigcup_{j=1}^{\boldsymbol{n}(I)} P_j(I), \;\; \mathcal{P}_{\text{reg}} := \bigcup_{I \text{ regular interval}} P(I).$$

Suppose now that $I = [p, q]$ is a noise interval. We modify the column

$$C_p = a_{p,1}, \ldots, a_{p,m}$$

to a column

$$C'_p = a'_{p,1}, \ldots, a'_{p,m},$$

by setting

$$a'_{p,k} := \begin{cases} 1, & \text{if } \sum_{i=p}^{q} a_{i,k} > 0 \\ \\ 0, & \text{if } \sum_{i=p}^{q} a_{i,k} > 0. \end{cases}$$

We apply the subroutine stack to the new column $C'_p$and the output is

$$\mathsf{stack}(C'_p) = \boldsymbol{n}(I), \;\; b_1 \leq t_1 < \cdots < b_n \leq t_n.$$

For $j = 1, \ldots, \boldsymbol{n}(I)$ we set

$$B_{0,j} := C[p, b_j], \;\; T_{0,j} := C[p, t_j],$$

$$B_{1,j} := C[q, b_j], \;\; T_{0,j} := C[q, t_j],$$

where $C[i, j]$ is defined by (1.1). Next, for $j = 1, \ldots, \boldsymbol{n}(I)$ we define the rectangle

$$\mathcal{R}_j(I) := \mathsf{polygon}(B_{0,j}, T_{0,j}, B_{1,j}, T_{1,j}),$$

and we set

$$\mathcal{R}(I) = \bigcup_{j=1}^{\boldsymbol{n}(I)} \mathcal{R}_j(I), \;\; \mathcal{P}_{\text{noise}} := \bigcup_{I \text{ noise interval}} R(I).$$

The output of the algorithm is the polytrapezoid

$$\mathcal{P}_\varepsilon(A) := \mathcal{P}_{\text{regular}} \cup \mathcal{P}_{\text{noise}}.$$

4 The main result

We wish to prove that the above algorithm produces an approximation of the original set which preserves the Euler characteristic, perimeter of the boundary, total curvature of the boundary and other important invariants. One way to precisely state this is to use the concept of *normal cycle* of a (subanalytic) set. The definition of a normal cycle uses basic concepts of geometric measure theory (for an introduction to the subject see [4, 8]).

Recall that a 1-*current* on a smooth manifold is an element of the dual space of compactly supported differential 1-forms. Thus, a current is an object T which associates a number $T(\omega)$ to a compactly supported 1-form ω. The number $T(\omega)$ can be thought of as the integral of ω over the current ω. For example an oriented smooth arc is a 1-current which acts through integration. The mass of a current T is the quantity

$$\sup\big\{\, T(\omega) : \omega \text{ is a 1-form}, \sup_x ||\omega_x|| \leq 1 \,\big\},$$

where $\| - \|$ denotes the Euclidean norm on the dual of $\mathbb{R}^2$. The mass of a 1-current defined by an oriented compact arc is equal to its length.

The normal cycle of of a planar PL set S is 1-current N^S living on the unit tangent sphere bundle of $\mathbb{R}^2$. It consists of a finite collection of compact, oriented real analytic arcs with multiplicities such that the resulting singular chain is a cycle. Each of these arcs is a Legendrian curve with respect to the canonical contact structure on the unit tangent sphere bundle. This cycle is uniquely characterized by the Morse theoretic properties of the restrictions to S of the linear functions on $\mathbb{R}^2$. Roughly speaking, the normal cycle is obtained as follows.

Denote by $T_\varepsilon(S)$ the tube of radius ε around S,

$$T_\varepsilon(S) := \big\{ p \in \mathbb{R}^2; \;\; \text{dist}\,(p, S) \leq \varepsilon) \,\big\}.$$

This is a domain in the plane and its normal cycle is the current $N^{T_\varepsilon(S)}$ defined the graph of the Gauss map of

the boundary. If we canonically identify the unit sphere bundle with the Cartesian product $S^1 \times \mathbb{R}^2$, then the graph of the Gauss map can be identified with the collection of points $(\nu(p), p) \in S^1 \times \mathbb{R}^2$, where $p \in \partial T_\varepsilon(S)$ and $\nu(p)$ is the outer unit normal vector to $\partial T_\varepsilon(S)$ at p. Then

$$N^S = \lim_{\varepsilon \searrow 0} N^{T_\varepsilon(S)}.$$

For a precise definition of the normal cycle we refer to [1, 2, 6, 9, 10]. In particular, [9] gives a beautiful description of the normal cycle, together with specific examples of normal cycles and how to retrieve geometric information from them.

In this setting, we can show the following theorem:

Theorem 3 *Suppose S is a generic PL subset of $\mathbb{R}^2$, i.e., no two vertices lie on the same vertical line. Fix a function $\sigma : \mathbb{R}^+ \to \mathbb{Z}^+$ satisfying (3.1).*

Let $\mathcal{P}_\varepsilon(S)$ be the PL approximation of S constructed via the Algorithm described above. Then the normal cycle $N^{\mathcal{P}_\varepsilon(S)}$ of $\mathcal{P}_\varepsilon(S)$ converges weakly to the normal cycle N^S of S as $\varepsilon \to 0$.

Proof: Due to lack of space we will omit most of the details. The complete proof is publicly available on the author's website[3]. We confine ourselves to outlining the most salient features of the proof.

The theorem is largely a consequence of the approximation theorem for normal cycles proved by Joseph Fu in [5]. This theorem implies the following result.

Proposition 4 *Suppose S is a PL subset of the plane and for each $\varepsilon \in (0,1)$ $L_\varepsilon(S)$ is a PL set such that the following hold*

1. *There is a compact subset K of the plane such that $L_\varepsilon(S) \subset K$ for all $\varepsilon \in (0,1)$.*
2. *There is a $M > 0$ such that*

$$\text{mass}\,(N^{L_\varepsilon(S)}) < M, \quad \forall \varepsilon \in (0,1).$$

3. *For almost every half plane H,*

$$\lim_{\varepsilon \searrow 0} \chi(H \cap L_\varepsilon(S)) = \chi(H \cap S),$$

where χ denotes the Euler characteristic.

Then the normal cycles $N^{L_\varepsilon(S)}$ converge weakly to the normal cycle N^S.

The bulk of the proof consists of verifying all the conditions in the above theorem for $\mathcal{P}_\varepsilon(S)$. Note that the first condition follows immediately from the construction of $\mathcal{P}_\varepsilon(S)$.

The second condition is a bit more difficult. If X is a PL set, then the mass of its normal cycle can be expressed in terms of its perimeter and its total curvature; see [9]. The conditions (3.1) allow us to produce upper bounds on the perimeter and the total curvature of $\mathcal{P}_\varepsilon(S)$ that are independent of ε. Condition 2 follows immediately from these upper bounds.

The third condition is the most challenging. To verify it we associate to each approximation $\mathcal{P}_\varepsilon(S)$ a graph Γ_ε, and a key observation is the fact that for ε sufficiently small the graph Γ_ε is isomorphic to the Reeb graph, [3, VI.4], of the map $h : S \to \mathbb{R}$ defined by the projection onto the x-axis. As a matter of fact, our entire algorithm can be viewed as a discretization of the Morse theory of the above map. □

5 Conclusion

Weak convergence in normal cycles implies the convergence of Euler characteristic, perimeter, curvature. This implies that the approximation $\mathcal{P}_\varepsilon(S)$ created by our algorithm converges in a very strong way to the original set, recovering information that the pixelation destroys.

A simple example of the approximation algorithm applied to the case of two intersecting line segments is shown in the Figure 4. Note the rectangle around the point of intersection indicating a noise region.

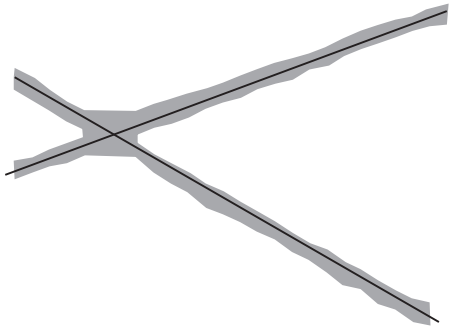

Figure 4: An example of $L_\varepsilon(S)$ where S is the union of two intersecting lines.

We note that the current approximation technique only applies to piecewise linear sets. This is because the construction relied on Proposition 2, which is only true for PL sets. In more general cases, such as semi-algebraic sets, fake cycles can stray further away from the singular points of S. (Think of the pixelation of a cusp.) Therefore more of the approximation will need

[3] `http://www.nd.edu/~browekam`

to be devoted to noise intervals in this case. We can show[4] that with a cleverer choice of σ, the general approximation method still works for semi-algebraic sets.

References

[1] A. Bernig: *The normal cycle of compact definable sets*, Israel J. Math., **159**(2007), 373-411.

[2] J. Cheeger, W. Müller, R. Schrader: *Kinematic and tube formulas for piecewise linear spaces*, Indian Univ. Math. J., **35**(1986), 737-754.

[3] H. Edelsbrunner, J. Harer: *Computational Topology. An Introduction*, Amer. Math. Soc., 2010.

[4] H. Federer: *Geometric Measure Theory*, Springer Verlag, 1969.

[5] J. Fu: *Convergence of curvatures in secant approximations*, J. Diff. Geom. **37**(1993), 177-190.

[6] J. Fu: *Curvature measures of subanalytic sets*, Am. J. Math. **116**(1994), 819-890.

[7] M. Gorseky, R. MacPherson: *Stratified Morse Theory*, Springer Verlag, 1988.

[8] F. Morgan: *Geometric Measure Theory: A Beginner's Guide*, Elsevier, 2009.

[9] J.M. Morvan: *Generalized Curvatures*, Springer Verlag, 2008.

[10] L.I. Nicolaescu: *On the normal cycles of subanalytic sets*, Ann. Glob. Anal. Geom., **39**(2011), 427-454.

[4] This is work in progress.