Local Versus Nonlocal Computation of Length of Digitized Curves

S. R. Kulkarni, S. K. Mitter, *Fellow, IEEE*, T. J. Richardson, and J. N. Tsitsiklis

***Abstract*—In this paper, we consider the problem of computing the length of a curve from digitized versions of the curve using parallel computation. Our aim is to study the inherent parallel computational complexity of this problem as a function of the digitization level. Precise formulations for the digitization, the parallel computation, and notions of local and nonlocal computations are given. We show that length cannot be computed locally from digitizations on rectangular tessellations. However, for a random tessellation and appropriate deterministic ones, we show that the length of straight line segments can be computed locally. Implications of our results for a method for image segmentation and a number of open problems are discussed.**

***Index Terms*— Local, nonlocal, parallel computation, length, digitized curve.**

I. Introduction

THERE has been a great deal of work on designing parallel algorithms for a broad range of computational tasks, but far less work on understanding the inherent "parallel computational complexity" of specific problems. In fact, the proper notion of parallel computational complexity may depend greatly on the parallel architecture, model of computation, and the specific problem at hand. One notion of complexity for certain types of parallel computation was considered by Minsky and Papert in their study of perceptrons [19]. They defined notions of "local" and "global" computations, variations of which have been further studied and used by others (e.g., see [1], [17], [11]).

Here, we also use notions of local and nonlocal computation inspired by the definitions in [19] to study the complexity of parallel computations used to estimate the length of a curve from digitized versions of the curve. Specifically, we study whether the computations can remain local as the digitization gets finer and finer, or whether to recover length in the continuum limit the computations must necessarily become nonlocal. Hence, an important aspect of our work is the idea of studying complexity as a function of the digitization level. This is a natural quantity to consider when one is interested in an underlying continuous problem, but chooses (or is forced) to solve the continuous problem by discrete approximations.

Given a curve in the plane, we consider digitizations formed by partitioning the plane into a number of regions (pixels), and representing the curve by the union of those regions through which the curve passes. For the computation, we imagine a processor located in each region whose state indicates whether or not the curve intersects the corresponding region. Each processor contributes additively to the final result based on the states of processors in a neighborhood. The notion of the "complexity" of the parallel computation for a fixed digitization level is the size of neighborhood required by the processors. The computation is said to be local if the neighborhood size remains uniformly bounded as the digitization gets finer. If the neighborhood size grows without bound as a function of the digitization level then the computation is said to be nonlocal.

We show that for the usual rectangular tessellations, length *cannot* be computed locally. On the other hand, using a classical result from stochastic geometry [26] we show that for straight line segments length *can* be computed locally using a random tessellation. Using a result from [22], it follows that length can also be computed locally for appropriate *deterministic* tessellations.

There has been previous work done on studying digitized curves and trying to estimate length from such digitizations (e.g., see [12], [13], [20], [21]). Our results provide a new perspective on this work. Furthermore, the original motivation for the present work arose in connection with a method for the restoration and segmentation of images [23], [8], [10]. Our results show the inherent complexity of using certain discrete Markov random fields in order to approximate continuous variational formulations of these problems.

In Section II, we give a precise description of the model of computation considered and definitions of local and nonlocal computations. In Section III, we provide local/nonlocal results for various tessellations. Implications of our results for the segmentation problem are discussed in Section IV. Finally, in Section V, we discuss a number of open problems and directions for further work.

II. Definition of Local Computation

In studying the question of local versus nonlocal computation of length, we restrict ourselves to a particular type of discrete representation of curves. First, for simplicity, we consider only curves contained in the unit square. For each n, we assume that the unit square is partitioned (or tessellated)

Manuscript received August 24, 1992; revised June 13, 1993. this work was supported in part by the U.S. Army Research Office under Contract DAAL03-86-K-0171, by the Department of the Navy under Air Force Contract F19628-90-C-0002, and by the National Science Foundation under Contract ECS-8552419. Recommended for acceptance by Associate Editor D. P. Huttenlocher.

S. R. Kulkarni is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA; email: kulkarni@ee.princeton.edu.

S. K. Mitter and J. N. Tsitsiklis are with the Laboratory for Information and Decision Systems, M.I.T., Cambridge, MA 02139 USA.

T. J. Richardson is with AT&T Bell Laboratories, 600 Mountain Avenue, Murray Hill, NJ 07974 USA.

IEEE Log Number 9301644.

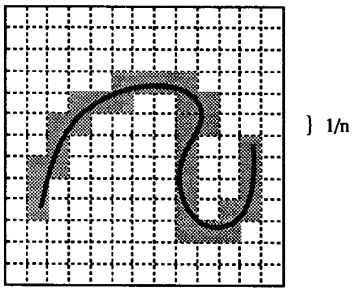


Fig. 1. Discretization method shown for rectangular tessellation with lattice spacing $1/n$.

into a number of regions $s_{n,1}, \cdots, s_{n,k(n)}$ with $k(n) \to \infty$ as $n \to \infty$. We think of n as indicating the discretization level, so that $n \to \infty$ typically corresponds to finer and finer tessellations. The discrete representation of a curve Γ on the tessellation at level n will consist of those regions $s_{n,i}$ through which Γ passes, i.e., for which $\Gamma \cap s_{n,i} \neq \emptyset$. Figure 1 shows an example of a rectangular tessellation together with a curve and its digitization on this tessellation.

The notions of local versus global computation we use are similar to those considered by Minsky and Papert [19] (see also [1], [17]). For a lattice at level n, we imagine a processor in each region $s_{n,1}, \cdots, s_{n,k(n)}$. Let $p_{n,j}$ denote the location (x, y coordinates) of the processor in region $s_{n,j}$. For simplicity, we will also let $p_{n,j}$ refer to the processor itself (as well as its position).

We assume that each processor has information as to whether or not the curve Γ passes through its region. Furthermore, each processor $p_{n,j}$ has an associated neighborhood $N_{n,j}$ which is a set of other processors at level n which provide information to the processor $p_{n,j}$. That is, processor $p_{n,j}$ performs a computation depending only on the state of the pixels in its neighborhood, which will be denoted by $\Gamma|_{N_{n,j}}$. We assume that the outputs of each of the processors are combined linearly to produce the final computed value. Hence, the computed value $\hat{L}_n(\Gamma)$ for the length of the curve Γ from its discretization at level n is given by an expression of the form

$$\hat{L}_n(\Gamma) = \sum_{j=1}^{k(n)} \phi_{n,j}(\Gamma|_{N_{n,j}}), \tag{1}$$

where $\phi_{n,j}$ is a function capturing the particular local computation performed by processor $p_{n,j}$ based on its neighborhood $N_{n,j}$.

The diameter of a neighborhood $N_{n,j}$ is the maximum distance between any two processors in $N_{n,j}$. The diameter d_n of the computation at discretization level n is the largest diameter over all neighborhoods $N_{n,j}$. Note that as $n \to \infty$, the processors necessarily get closer together, since they are all within the unit square. Since we are interested in computations in which each processor does not communicate to too many other processors, it is not sufficient simply to bound d_n as $n \to \infty$. Instead, we will bound the scaled diameter $\sqrt{k(n)}d_n$. A computation of the above form is said to be diameter limited (in the limit, or as the lattice spacing tends to zero) if $\sqrt{k(n)}d_n$ is uniformly bounded as a function of n, i.e., for some $d < \infty$ we have $\sqrt{k(n)}d_n \leq d$ for all n.

A diameter limited computation provides one notion of what we mean by a local computation. Following [19], another notion is that of an order limited computation. The order of a neighborhood $N_{n,j}$ is simply the number of processors in $N_{n,j}$, i.e. its cardinality. The order α_n of a computation at discretization level n is the maximum order over all j of $N_{n,j}$. Then, an order limited computation is one for which α_n is uniformly bounded as a function of n, so that there is some $\alpha < \infty$ such that $\alpha_n < \alpha$ for all n.

It is difficult to prove any results without imposing some additional structure on the computation. We consider the case of a translation invariant neighborhood structure and translation invariant processors meaning that for each n, the neighborhoods $N_{n,j}, N_{n,j'}$ of processors j, j' are simply translates of each other, and the $\phi_{n,j}$ are independent of j. (Of course there is an issue for processors at the boundary of the region, but for our purposes this can be ignored since we are interested in letting the digitization level get finer and finer in a such a way that the proportion of such boundary processors becomes negligible.) We also consider the case in which only those processors which are "on" can contribute to the computation. That is, we assume that the contribution $\phi_{n,j}(\Gamma|_{N_{n,j}})$ of processor $p_{n,j}$ is zero if Γ does not pass through the region associated with $p_{n,j}$. In the case of regular tessellations, these assumptions allow a simplification of the form of the computation in (1). Specifically, for an order or diameter limited computation there are a finite number K of distinct patterns for $\Gamma|_{N_{n,j}}$ (i.e., states of pixels in a neighborhood). Each processor which sees pattern i in its neighborhood contributes the same quantity $a_{n,i}$ to the total computation. Therefore, if we let t_n denote the total number of pixels through which Γ passes, and let $f_{n,i}(\Gamma)$ denote the frequency of occurrence of pattern i, then $\hat{L}_n$ is given by

$$\hat{L}_n(\Gamma) = t_n(\Gamma) \sum_{i=1}^{K} a_{n,i} f_{n,i}(\Gamma). \tag{2}$$

III. Local/Nonlocal Results for Various Tessellations

A rectangular digitization is the one most commonly used in image processing. In this case, at discretization level n the unit square is partitioned along the coordinate axes into n^2 square pixels of size $1/n$ by $1/n$. The pixels correspond to the closed lattice squares of $\frac{1}{n}Z^2$. The discrete version of a curve Γ is composed of the union of closed lattice squares of $\frac{1}{n}Z^2$ through which Γ passes (see Fig. 1). For such discretizations, we have the following result.

Theorem 1: The length of a curve cannot be computed using a diameter limited computation from discrete approximations on a rectangular tessellation. In particular, if $nd_n \leq d < \infty$ then for some straight line Γ, $\lim_{n\to\infty} \hat{L}_n(\Gamma) \neq L(\Gamma)$.

Proof: We will proceed by showing that any diameter limited computation fails to compute length appropriately in the limit on many straight lines. Consider a line segment of unit length and let θ be the angle that the extension of the segment makes with the x-axis. Since we are considering only

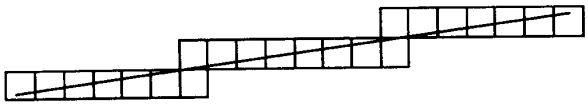


Fig. 2. A line segment with small slope and its digitization.

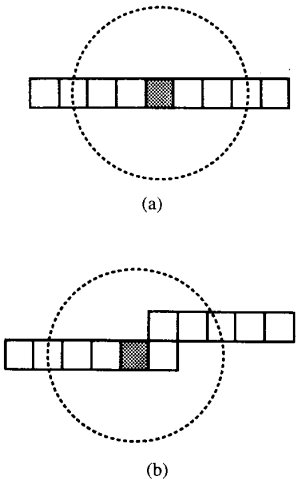


Fig. 3. Two types of patterns in diameter limited neighborhood of a pixel arising from line segments with sufficiently small slope.

the case where Γ is such a straight line segment, we will write (2) indexed by θ:

$$\hat{L}_n(\theta) = t_n(\theta) \sum_{i=1}^{K} a_{n,i} f_{n,i}(\theta). \tag{3}$$

For lines with small θ, as n gets large the digitization consists of long stretches of pixels in a row with occasional corners (or shifts) to different rows (see Fig. 2). Now, suppose the computation is diameter limited with $nd_n \leq d$. Then for $0 \leq \theta < \tan^{-1} \frac{1}{d}$ the corners are sufficiently far apart so that the digitized pattern in the neighborhood of every processor contains either no corners or exactly one corner (see Fig. 3), with the different locations of the corner in the neighborhood corresponding to different patterns.

Since we are concerned with the behavior of the computation as $n \to \infty$, we can ignore the effects at the ends of the line segment, and the effects of the offset of the line segment with respect to the digitization. For each $0 \leq \theta < \tan^{-1} \frac{1}{d}$, the frequencies of occurrence of all patterns which contain a corner are approximately the same for large n. Hence, we can simplify (3) to

$$\hat{L}_n(\theta) = t_n(\theta)[a_{n,1} f_{n,1}(\theta) + a_{n,2} f_{n,2}(\theta)], \tag{4}$$

where $f_{n,1}$ and $f_{n,2}$ denote the frequency of occurrence of patterns without a corner and with a corner respectively.

For large n, there are approximately $n \sin\theta$ corners for a unit length segment at angle θ, and the segment passes through approximately $n \cos\theta$ columns. Hence, the digitized version of the line segment contains approximately $t_n(\theta) = n\cos\theta + n\sin\theta$ total pixels. Also, since the computation is diameter limited with $nd_n \leq d$, at most d pixels see a given corner, so that $nd\sin\theta$ pixels see some corner. Therefore, $f_{n,2}(\theta) = d\sin\theta/(\cos\theta + \sin\theta)$ and $f_{n,1}(\theta) = 1 - f_{n,2}(\theta)$. Substituting these expressions in (4), letting $n \to \infty$, and assuming that $\lim_{n\to\infty} na_{n,i} = a_i < \infty$ for $i = 1, 2$ gives

$$\hat{L}(\theta) = a_1 \cos\theta + (a_1 + d(a_2 - a_1)) \sin\theta, \tag{5}$$

for $0 \leq \theta < \tan^{-1} \frac{1}{d}$. Since the line segments at all angles have unit length, in the interval $0 \leq \theta < \tan^{-1} \frac{1}{d}$, the computation is correct only for those θ for which $\hat{L}(\theta) = 1$. For finite d, from (5) $\hat{L}(\theta) = 1$ clearly cannot be satisfied for all $0 \leq \theta < \tan^{-1} \frac{1}{d}$ (in fact, it can be exactly satisfied for at most two values of θ in the desired interval).

We expect that similar results are true for the other standard regular tessellations (i.e., hexagonal and triangular). However, it is interesting that it is not true for all tessellations, as shown below.

We first consider random tessellations produced by a number of random straight lines. A "random" straight line (i.e., a uniform distribution for the set of straight lines intersecting a bounded domain) can be defined as follows. A line in the plane will be parameterized by the polar coordinates r, θ of the point on the line which is closest to the origin, where $r \geq 0$ and $0 \leq \theta \leq 2\pi$. The set (manifold) of all lines in the plane parameterized in this way corresponds to a semi-infinite cylinder. A well known result from stochastic geometry states that the unique measure (up to a scale factor) on the set of lines which is invariant to rigid transformations of the plane (translation, rotation) is $drd\theta$, i.e., uniform density in r and θ. This measure is thus independent of the choice of coordinate system, and is referred to as the uniform measure (or density) for the set of straight lines in the plane. This measure corresponds precisely to the surface area measure on the cylinder.

From this measure, a uniform probability measure can be obtained for the set of all straight lines intersecting a bounded domain. Specifically, the set of straight lines intersecting a bounded domain is a bounded subset of the cylinder. The uniform probability measure is then just the surface area measure of the cylinder suitably normalized (i.e., by the area of the subset of the cylinder).

We can now state the following classic result from stochastic geometry (e.g., see [26, section 3.2, pp. 30–32] or [6, (3), p. 4]).

Theorem 2: Let X be a bounded convex subset of R^2, and let $c \subset X$ be a rectifiable curve. Suppose lines intersecting X are chosen uniformly, and let $n(\tilde{x}, c)$ denote the number of

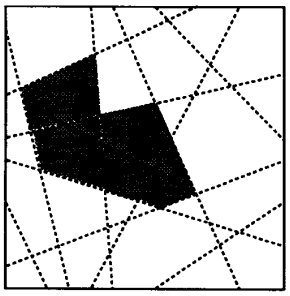


Fig. 4. An irregular tessellation produced by drawing a number of straight lines, and a digitization of a straight line segment using this tessellation.

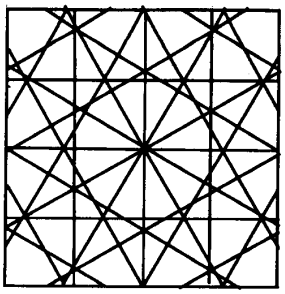


Fig. 5. A tessellation produced by a set of equally spaced lines taken at a number of angles—i.e., by sampling the set of lines regularly in both the radial and angular coordinates. Shown here are lines from 0° to 150° in 30° increments.

intersections of the random line $\tilde{x}$ with the curve c. Then

$$En(\tilde{x}, c) = \frac{2}{A}\mathcal{L}(c)$$

where $\mathcal{L}(c)$ denotes the length of the curve c and A is the perimeter of X.

For simplicity, we are considering the case where X is the unit square. In this case, the theorem reduces simply to $En(\tilde{x}, c) = \frac{1}{2}\mathcal{L}(c)$.

A random tessellation can be formed by drawing a set of random lines as described above. Figure 4 shows what a tessellation of this type might look like, as well as a digitization of a straight line segment on this tessellation (although the lines in the figure were not actually generated randomly). The "lattice spacing" of the tessellation is related to the number of random lines drawn, and we will refer to a tessellation formed by n lines as a tessellation at level n. For such randomly formed tessellations, we have the following result for computing the length of any straight line segment.

Theorem 3: Let Γ be any straight line segment. There is a diameter limited computation on tessellations formed from independent lines drawn uniformly which converges to the length of Γ with probability one.

Proof: For the computation on the tessellation at level n, we will let each processor which is "on" (i.e., which Γ passes through) contribute $2/n$ independent of the state of all other processors. Hence, the neighborhood of a given processor consists only of the processor itself, so that the computation is clearly diameter limited.

Now we need to show that as $n \to \infty$ this computation recovers the length of a straight line segment Γ. Let $\beta(n, \Gamma)$ denote the number of pixels comprising the digitized version of Γ on the lattice at level n. For a random line ℓ, let $m(\ell, \Gamma)$ denote the number of intersections between ℓ and Γ. Since Γ is a straight line segment, for almost all ℓ, $m(\ell, \Gamma)$ is either zero or one. Furthermore, since the pixels consist of regions formed by straight lines and Γ is a straight line segment, the number of pixels comprising the digitized version of Γ is just one plus the number of straight lines intersecting Γ. That is, $\beta(n, \Gamma) = 1 + \sum_{i=1}^{n} m(\ell_i, \Gamma)$ so that

$$\hat{L}_n = 2 \cdot \frac{1}{n}\sum_{i=1}^{n} m(\ell_i, \Gamma) + \frac{2}{n}.$$

From the law of large numbers, we have that as $n \to \infty$, $\hat{L}_n \to 2 \cdot Em(\ell, \Gamma)$ with probability one. The stochastic geometry result mentioned above (Theorem 2) states that if Γ is any rectifiable curve in the unit square and ℓ is a random line intersecting the unit square drawn uniformly then $Em(\ell, \Gamma) = \frac{1}{2}L(\Gamma)$ (e.g., see [26]). Thus, using this result we have that $\hat{L}_n \to L(\Gamma)$ with probability one. □

The intuitive idea of the results above is that the length of a line segment is twice the area of a corresponding subset of the cylinder, namely the area of the set of lines which intersect the line segment. With a rectangular tessellation, we obtain samples on the cylinder only for $\theta = 0$ and $\theta = \pi/2$. On the other hand, with the random tessellation, we obtain a random sampling of points on the cylinder from which we can easily estimate the desired area. This suggests that there is a tradeoff between the complexity of the sampling used and the complexity of the resulting computation. It also suggests that appropriate deterministic sampling strategies which sample the set of lines in both the radial and angular coordinates on the cylinder should allow a local computation of the length of a line segment. Figure 5 shows an example of a tessellation formed in this way. The lines of the tessellation in the figure are formed from a set of equally spaced lines taken at several orientations. The theorem below shows a local computation result for such a deterministic sampling strategy. Specifically, we consider the tessellation at level n to be that formed by parallel lines with spacing h_n taken at angles $\pi j/n$ for $j = 0, \cdots, n-1$, where $h_n \to 0$ as $n \to \infty$. Clearly this tessellation samples lines over the entire cylinder of lines. (Note that for each angle we consider a whole set of parallel lines, so that we need only consider angles from 0 to π rather than 0 to 2π.) Denote this tessellation by U_{n,h_n}. Moran [22] has obtained results on estimating the length of a curve by counting intersections with the straight lines forming the tessellation U_{n,h_n}. The following result on the local computation of length for the tessellation U_{n,h_n} uses a result from [22] (attibributed in [22] to Steinhaus).

Theorem 4: Let Γ be any straight line segment. There is a local computation on the deterministic tessellation U_{n,h_n} that converges to the length of Γ.

Proof: If $P(\theta)$ denotes the integral of the projection of a curve Γ on a line at direction θ (with points counted according

to their multiplicity), then the length of Γ is given by

$$L(\Gamma) = \frac{1}{4}\int_0^{2\pi} P(\theta)\,d\theta = \frac{1}{2}\int_0^{\pi} P(\theta)\,d\theta$$

(e.g., see [22]). This suggests that the length of Γ can be estimated using samples of $P(\theta)$ at a number of angles. This can be done, and in fact Moran [22] attributes the following result to Steinhaus which essentially bounds the error in estimating the length of a curve Γ by using the above expression and measurements of $P(\cdot)$ at a set of angles. Let $P_j = P(\pi j/n)$ for $j = 0, 1, \cdots, n-1$, i.e., the P_j are the values of $P(\cdot)$ for n equally spaced angles between 0 and π. Then

$$\eta_n \cos\left(\frac{\pi}{2n}\right) L(\Gamma) \le \frac{1}{2n}\sum_{j=0}^{n-1} P_j \le \eta_n L(\Gamma) \tag{6}$$

where

$$\eta_n = \frac{\pi}{2n}\left[\sin\left(\frac{\pi}{2n}\right)\right]^{-1}. \tag{7}$$

Note that $\eta_n \to 1$ as $n \to \infty$ so that as expected the approximation gives us the true length as the number of samples tends to infinity.

Now we consider the effect of estimating the P_j as opposed to obtaining the exact value. If Γ is a straight line segment, then P_j is simply the length of the projection of Γ in the direction corresponding to P_j. This length can be estimated by simply counting the number of intersections with parallel lines orthogonal to the direction corresponding to P_j and multiplying by the spacing of the parallel lines. If the spacing of the lines is h_n, then the maximum possible error in estimating P_j in this manner is also h_n. Thus, the maximum possible error in estimating $\frac{1}{2n}\sum_{j=1}^{n} P_j$ by counting the total number of intersections of Γ with the tessellation U_{n,h_n} is $h_n/2$.

Now, as in Theorem 3, for straight line segments the number of intersections with the lines forming U_{n,h_n} is just the number of pixels in the digitization of Γ minus one. Hence, for the computation on the tessellation U_{n,h_n}, we will let each processor which is "on" (i.e., through which Γ passes) contribute $\frac{h_n}{2n}$ independent of the state of all other processors. Then,

$$\hat{L}_n(\Gamma) = \frac{h_n}{2n} \cdot (\#\text{ "on" pixels}) \tag{8}$$

$$= \frac{h_n}{2n} \cdot (1 + \#\text{ intersections with lines in } U_{n,h_n}). \tag{9}$$

Therefore,

$$\left|\hat{L}_n(\Gamma) - \frac{1}{2n}\sum_{j=0}^{n-1} P_j\right| \le \frac{h_n}{2} + \frac{h_n}{2n} = \frac{h_n}{2}\left(1 + \frac{1}{n}\right). \tag{10}$$

Combining this with the bound of (6) from [22] we obtain

$$\eta_n \cos\left(\frac{\pi}{2n}\right) L(\Gamma) - \frac{h_n}{2}\left(1 + \frac{1}{n}\right) \le \hat{L}_n(\Gamma) \le \eta_n L(\Gamma) + \frac{h_n}{2}\left(1 + \frac{1}{n}\right). \tag{11}$$

Since $h_n \to 0$ and $\eta_n \to 1$ as $n \to \infty$, we have that $\hat{L}_n(\Gamma) \to L(\Gamma)$. Also, since the neighborhood of a given processor consists only of the processor itself, the computation is clearly diameter limited. □

IV. Relation to a Method for Image Segmentation

Our original motivation for considering the question of local versus nonlocal computation of length arose in connection with a problem in machine vision. Specifically, a variational method for image segmentation and restoration was proposed by Mumford and Shah in [23] (see also Blake and Zisserman [7], [8]). Their approach involves minimizing a cost functional over a space of boundaries with suitably smooth functions within the boundaries. If g represents the observed image defined on $\Omega \subset R^2$, then a restored image f and its associated edges Γ are found by minimizing

$$E(f,\Gamma) = c_1 \int\int_{\Omega} (f-g)^2 dxdy + c_2 \int\int_{\Omega\setminus\Gamma} \| \nabla f \|^2 \, dxdy + c_3 L(\Gamma), \tag{12}$$

where c_1, c_2, c_3 are constants and $L(\Gamma)$ denotes the length of Γ.

Discrete versions of these problems have also been proposed [8], [23]. In these discrete problems, the original image g is defined on a subset of the lattice $\frac{1}{n}Z^2$ with lattice spacing $\frac{1}{n}$. The restored image f is defined on the same lattice, while the boundary Γ consists of a set of line segments joining neighboring points of the dual lattice. For the discrete problem, f and Γ are found by minimizing

$$E(f,\Gamma) = c_1 \sum_{i\in\Omega} \frac{1}{n^2}(f_i - g_i)^2 + c_2 \sum_{\substack{i,i'\in\Omega \\ \text{adjacent} \\ \overline{ii'}\cap\Gamma=\emptyset}} (f_i - f_{i'})^2 + c_3 L(\Gamma), \tag{13}$$

where $\overline{ii'}$ is the line segment joining the lattice points i and i'. Similar discrete problems arise in the context of using Markov random fields for problems in vision as proposed by Geman and Geman [10] and studied in recent years by many other researchers (e.g., see [18]).

It has been pointed out [14], [16] that, unfortunately, the discrete version above does not properly approximate the continuous version as the lattice spacing tends to zero. The problem arises with the third term involving the length of the boundary. Clearly, by simply summing the lengths of the boundary elements in the discrete approximation one does not, in general, recover the true length of the curve for fine lattice spacings. For example, the diagonal of the unit square has length $\sqrt{2}$ but a discrete approximation to the diagonal using boundary elements from the standard discretization has length close to 2 (see Fig. 6). Thus, for certain images and values of the constants c_1, c_2, c_3 solutions to the discrete problem do not approximate solutions to the continuous problem (even for arbitrarily small lattice spacings).

In [14]–[16], some discretizations which properly approximate the continuous problem were presented. These formulations have the advantage that solutions to the discrete versions converge to a solution of the continuous problem as the lattice

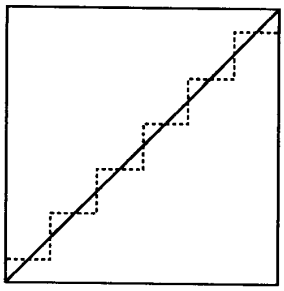


Fig. 6. A diagonal of the unit square and a discrete approximation using the standard discretization. Note that the length of the diagonal is $\sqrt{2}$ while for fine discretizations the length of the approximation is almost 2.

spacing tends to zero. However, they have the disadvantage that for discrete boundaries the cost functional is considerably more difficult to evaluate than for the standard discretization. This has important implications as to the suitability of these methods for computation on parallel architectures.

Specifically, as we mentioned above, the discrete formulations have a close relationship to problems arising from a probabilistic approach using Markov random fields (MRF's), which are attractive for a number of theoretical and practical reasons. One major reason for the attractiveness of MRF's is their local neighborhood structure. A very useful property of the standard discretization is its small neighborhood structure independent of the level of discretization. In fact, for the standard discretization the MRF requires no interaction between the boundary sites regardless of the discretization level. On the other hand, the discretizations in [14]–[16] require complex potentials over a neighborhood whose size is required to grow (unboundedly) as the lattice spacing tends to zero. For very fine discretizations, the neighborhood structure is highly nonlocal and the advantages of the MRF structure are essentially lost. This is due to the choice of the cost for the discrete boundaries, which were selected for their convergence properties to the true length of the curves in the continuum limit.

The distinction can also be formalized along the following lines. Consider a distributed implementation in which there is a processor at each lattice site. The state of a processor is either zero or one depending on whether or not the boundary passes through the associated lattice square. To compute the length terms in the approximations studied in [14]–[16], each processor must perform a computation depending on the state of a very large number (tending to infinity) of other processors as the lattice spacing tends to zero. On the other hand, for the cost term of the usual discretization, the contribution of a particular processor to the total cost depends on the state of the processor but is independent of the state of all other processors (regardless of the discretization level). Hence, if implemented in parallel architectures in the natural way, the methods that possess the proper convergence properties require computations that are in some sense nonlocal as the lattice spacing tends to zero, while the usual discretization results in a local computation (independent of the discretization level) but fails to have the right convergence properties. Note that for the method using piecewise linear approximations, if implemented in the natural way, the computation can be done locally but each processor requires an unbounded number of states in the continuum limit (to indicate whether an endpoint of the line segment is present at that processor and, if so, at which processor the other endpoint lies).

A natural question is whether the computational difficulties discussed above can be circumvented by a clever discrete approximation. That is, is it possible to retain the convergence properties with computations using local neighborhood structures? The results of Section III suggest that for rectangular lattices (and probably for other regular tessellations such as hexagonal or triangular) the difficulties are not merely due to a poor choice of discrete approximations, but are inherent difficulties associated with any discrete approximation to measures of length. However, interestingly the results of Section III also suggest that the problems with nonlocal computation can be avoided for appropriate random and deterministic tessellations (although other computational or algorithmic difficulties may arise due to the nature of the tessellations).

Also, as alluded to above, the nonlocal computations can likely be avoided if the processors are allowed to have infinitely many states. For example, this could correspond to associating a direction (or local tangent) to each boundary element in addition to just its presence or absence. Hence, in the MRF formulations this might correspond to coupled intensity and boundary fields both of which are real valued. A somewhat different approach to having real valued boundary elements is suggested by an important result of Ambrosio [3], [4]. He obtained an interesting approximation to the original variational problem. Specifically, he showed that the functional

$$E^h(f,v) = c_1 \int\int_\Omega (f-g)^2 + c_2 \int\int_\Omega (1-v^2)^h \parallel \nabla f \parallel^2 + c_3\left(\int\int_\Omega (1-v^2)^h \parallel \nabla v \parallel^2 + \frac{h^2 v^2}{16}\right)$$

Γ-converges (e.g., see [5]) to $E(f,\Gamma)$ of (12) as $h \to \infty$, so that minimizers of $E^h(f,v)$ converge to a minimizer of $E(f,\Gamma)$ as $h \to \infty$. Here, f is as before and $v{:}\Omega \to [0,1]$ plays the role of the boundaries. For finite h, v represents a sort of smoothed version of Γ in the sense of having a value close to 1 near Γ and having a value of 0 away from Γ, and varying continuously in between. This result suggests a natural digitization of $E(f,\Gamma)$ of (10) by taking a finite difference approximation to $E^h(f,v)$ as discussed in [25] and [9]. However, as far as we know, a proof of convergence for such finite difference approximations is lacking in this case. We expect that convergence should hold as long as $h \to \infty$ appropriately as the lattice spacing $1/n \to 0$, namely $h/n \to 0$. Such a conjecture is natural in light of the results of [14], [16] and was in fact stated in [25]. Furthermore, convergence issues aside, it is not clear that computational difficulties are avoided with these approaches. For example, in Ambrosio's approximation, there may be some computational or numerical problems as $h \to \infty$. Further work needs to be done to understand whether any computational difficulties arise in this case.

V. Discussion and Open Problems

There are a number of interesting questions and directions to pursue along the lines of this paper. It would be interesting to extend the local computation results for both the random and deterministic tessellations (Theorems 3 and 4) to general curves. The difficulty is that for general curves there isn't a simple correspondence between the number of regions intersected by the curve and the number of intersections the lines make with the curve. It would also be interesting to extend the local and nonlocal results to other tessellations. For example, we conjecture that nonlocal results similar to Theorem 1 hold for regular tessellations such as triangular or hexagonal. Likewise, we conjecture that results similar to Theorem 3 hold for other random tessellations such as Voronoi tessellations obtained from homogeneous planar Poisson point processes. (For work on random tessellations see for example [2], [29].) The results of [27] may be useful in proving results of this type. One difficulty in dealing with tessellations which are not formed by a number of straight lines is that the duality between intersections of the lines and sampling on the cylinder (manifold of straight lines) is lost. Perhaps there is a more general way in which to view the sampling which works for other tessellations.

Another direction to pursue is to try to relax some of the assumptions on the computation such as translation invariance, etc. However, it seems that proving results in these cases will be difficult. One extension that we feel should go through is to prove a nonlocal result like Theorem 1 for order limited computations as opposed to just diameter limited. Also, it would be useful to obtain error bounds in terms of the diameter (or order) of the computation, since it is likely that although an exact computation in the limit may be nonlocal, a good approximation can be obtained with a small diameter (order). In fact, one could use (5) to obtain a lower bound on the achievable error for diameter limited computations.

A natural question is whether the tessellations which allow local computation of length can be used to construct a discrete version of the segmentation problem which is local and yet possesses the appropriate convergence properties in the continuum limit. We expect that this can be done, although the corresponding MRF structure would be somewhat complicated due to the irregular placements of lattice sites.

The notion of local versus nonlocal computations appears to be of fundamental importance. The work presented in this paper suggests many other general directions which may be interesting to pursue. It may be worthwhile investigating whether other computations (e.g., determining convexity or connectedness from discrete approximations of a set) can be done locally. One could consider questions of local/nonlocal computations using other discretizations or in which the processors have access to other types of data, as opposed to just data from a discretization on a tessellation as considered here. It might also be interesting to consider forms of computation other than just those of the type in (1), as well as to investigate other notions of local and nonlocal computations. Our results show that certain local lattice systems may inherently lack the ability to perform specific computations due to the arrangement and connections of the lattice sites. It may be interesting to study whether other lattice-type systems such as cellular automata or spin systems in statistical mechanics also possess inherent computational limitations arising from their architecture.

References

[1] H. Abelson, "Towards a theory of local and global computation," *Theoretical Comput. Sci.*, vol. 6, pp. 41–67, 1978.

[2] R. V. Ambartzumian, Ed, "*Stochastic and Integral Geometry*. New York: Kluwer Academic, 1987.

[3] L. Ambrosio, "Existence theory for a new class of variational problems," Center for Intelligent Contr. Syst. Rep. CICS-P–93, MIT, 1988.

[4] ——, "Variational problems in SBV," Center for Intelligent Contr. Syst. Rep. CICS-P-86, MIT, 1988.

[5] H. Attouch, *Variational Convergence for Functions and Operators*. Marshfield, MA: Pitman, 1984.

[6] A. J. Baddeley, "Stochastic geometry and image analysis," in *CWI Monographs*. Amsterdam: North Holland, 1986.

[7] A. Blake and A. Zisserman, "Invariant surface reconstruction using weak continuity constraints," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, Miami, 1986, pp. 62–67.

[8] A. Blake and A. Zisserman, *Visual Reconstruction*. Cambridge, MA: MIT Press, 1987.

[9] D. Geiger and A. Yuille, "A common framework for image segmentation by energy functions and nonlinear diffusion," *Int. J. Comput. Vision*, vol. 6, pp. 227–243, 1991.

[10] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 721–741, 1984.

[11] T. Y. Kong and A. Rosenfeld, "If we use 4- or 8-connectedness for both the objects and the background, the Euler characteristic is not locally computable," *Pattern Recognit. Lett.*, vol. 11, pp. 231–232, 1990.

[12] J. Koplowitz and A. Bruckstein, "Design of perimeter estimators for digitized planar shapes," *SPIE vol. 1001 Visual Commun. Image Processing*, 1988, pp. 756–763.

[13] ——, "Design of perimeter estimators for digitized planar shapes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 611–622, 1989.

[14] S. R. Kulkarni, S. K. Mitter, and T. J. Richardson, "An existence result and lattice approximations for a variational problem," *Signal Processing, Part I: Signal Processing Theory*, Auslander *et al.*, Eds. New York: Springer-Verlag, 1990, IMA vol. 22, pp. 189–210.

[15] S. R. Kulkarni and S. K. Mitter, "Variational versus Markov random field methods for image segmentation," *SPIE, Neural Stochastic Methods in Image and Signal Processing III*, vol. 2304, 1994.

[16] S. R. Kulkarni, "Problems of computational and information complexity in machine vision and learning," Ph.D. thesis, Dept. of Electrical Eng. and Comput. Sci., M.I.T., 1991.

[17] C. N. Lee and A. Rosenfeld, "Simple Connectivity is Not Locally Computable for Connected 3D Images," *Comput. Vision, Graphics, and Image Processing*, vol. 51, pp. 87–95, 1990.

[18] J. Marroquin, S. Mitte, and T. Poggio, "Probabilistic solution of ill-posed problems in computational vision," *J. Amer. Statist. Soc.*, vol. 82, no. 397, pp. 77–89, 1987.

[19] M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA: The MIT Press, 1969.

[20] U. Montanari, "Continuous skeletons from digitized images," *J. ACM*, vol. 16, no. 4, pp. 534–549, 1969.

[21] U. Montanari, "A note on minimal length polygonal approximation to a digitized contour," *Commun. ACM*, vol. 13, no. 1, 1970.

[22] P. A. P. Moran, "Measuring the length of a curve," *Biometrika*, vol. 53, pp. 359–364, 1966.

[23] D. Mumford and J. Shah, "Boundary detection by minimizing functionals," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, San Francisco, CA, 1985, pp. 22–26.

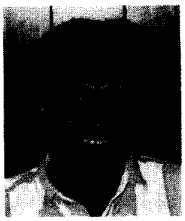
[24] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," Ctr. for Intelligent Contr. Syst. Rep. CICS-P-68, 1988. Submitted to *Commun. Pure Appl. Math.*.

[25] T. J. Richardson, "Scale independent piecewise smooth segmentation of images via variational methods," Ph.D. thesis, Dept. of E.E.C.S., M.I.T., 1989.

[26] L. A. Santalo, *Integral Geometry and Geometric Probability*., , vol. 1 of *Encyclopedia of Mathematics and its Applications*. Reading, MA: Addison-Wesley, 1976.

[27] J. M. Steele, "Subadditive Euclidean functionals and nonlinear growth in geometric probability," *Ann. Probab.*, vol. 9, no. 3, pp. 365–376, 1981.

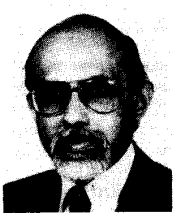
[28] H. Steinhaus, "Length, shape, and area," *Colloq. Math.*, vol. 3, pp. 1–13, 1954.
[29] D. Stoyan, W. S. Kendall, and J. Mecke, *Stochastic Geometry and Its Applications*. New York: Wiley (Series in probability and mathematical statistics), 1987.



Sanjeev R. Kulkarni received the B.S. degree in mathematics, the B.S. degree in electrical engineering, the M.S. degree in mathematics from Clarkson University in 1983, 1984, and 1985, respectively, the M.S. degree in electrical engineering from Stanford University in 1985, and the Ph.D. degree in electrical engineering from M.I.T. in 1991.

From 1985 to 1991, he was a Member of the Technical Staff at M.I.T. Lincoln Laboratory working on the modeling and processing of laser radar measurements. In the spring of 1986, he was as a part-time faculty at the University of Massachusetts, Boston. Since 1991, he has been an Assistant Professor of Electrical Engineering at Princeton University.

Dr. Kulkarni received an ARO Young Investigator Award in 1992 and an NSF Young Investigator Award in 1994. His research interests include pattern recognition, image processing, machine learning, and statistical inference.



Sanjoy K. Mitter (M'68–SM'77–F'79) received the Ph.D. degree from the Imperial College of Science and Technology, University of London, in 1965.

He had previously worked as a research engineer at Brown Boveri & Co. Ltd., Switzerland (now ASEA Brown Boveri) and Battelle Institute in Geneva, Switzerland. He taught at Case Western Reserve University from 1965 to 1969 and joined MIT in 1969 first as a Visiting Professor and then in 1970 as Associate Professor in the Department of Electrical Engineering and Computer Science. He is currently Professor of Electrical Engineering and Co-Director of the Laboratory for Information and Decision Systems. He is also Director of the Center for Intelligent Control Systems, an inter-university (Brown–Harvard–MIT) center for research on the foundations of intelligent systems. He has held visiting positions at the Tata Institute of Fundamental Research, Bombay, India; Scuola Normale Superiore, Pisa, Italy; Imperial College of Science and Technology; Institut National de Recherche en Informatique et en Automatique, France; University of Groningen, the Netherlands, and several universities in the United States.

His research has spanned the broad areas of Systems, Communication and Control. Although his primary contributions have been on the theoretical foundations of the field, he has also contributed to significant engineering applications, notably in the control of interconnected power systems and automatic recognition and classification of electrocardiograms. His current research interests are theory of stochastic dynamical systems, nonlinear filtering, stochastic and adaptive control; mathematical physics and its relationship to system theory; image analysis and computer vision; and structure, function and organization of complex systems.

Professor Mitter has served on several advisory committees and editorial boards for IEEE, SIAM, AMS, NSF, and ARO. He is currently Associate Editor of Acta Applicandae Mathematicae; Circuits, Systems, and Signal Processing; Journal of Applied Mathematics and Optimization; SIAM Review; and the Ulam Quarterly. In 1988, he was elected to the National Academy of Engineering.

T. J. Richardson recieved the B.Sc. and M.Sc. degrees from the University of Toronto in 1983 and 1986, respectively, both in electrical engineering, and the Ph.D. degree in electrical engineering, from the Massachusetts Institute of Technology in 1990.

Since 1990, he has been a employed by the Mathematical Sciences Research Center, AT&T Bell Laboratories. His current research interests include integral and computational geometry, computer vision, and dynamical systems.

John N. Tsitsiklis was born in Thessaloniki, Greece, in 1958. He received the B.S. degree in mathematics (1980), and the B.S. (1980), M.S. (1981) and Ph.D. (1984) degrees in electrical engineering, all from the Massachusetts Institute of Technology, Cambridge, Massachusetts.

During the academic year 1983–84, he was an acting Assistant Professor of Electrical Engineering at Stanford University, Stanford, CA. Since 1984, he has been with the Massachusetts Institute of Technology, where he is currently Professor of Electrical Engineering. His research interests are in the areas of systems and control theory, operations research, and parallel and distributed computation.

Dr. Tsitsiklis is a coauthor (with D. Bertsekas) of *Parallel and Distributed Computation: Numerical Methods* (1989). He has been a recipient of an IBM Faculty Development Award (1983), an NSF Presidential Young Investigator Award (1986), an Outstanding Paper Award by the IEEE Control Systems Society (for a paper coauthored with M. Athans, 1986), and of the Edgerton Faculty Achievement Award by M.I.T. (1989). He was a plenary speaker at the 1992 IEEE Conference on Decision and Control. Finally, he is an Associate Editor of *Automatica and Applied Mathematics Letters*, and has been an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL.