Fast Nearest Neighbor Search in Medical Image Databases

Flip Korn
Dept. of Computer Science
Univ. of Maryland
College Park, MD
flip@cs.umd.edu

Nikolaos Sidiropoulos
Inst. for Systems Research
Univ. of Maryland
College Park, MD
nikos@isr.umd.edu

*Christos Faloutsos**
Institute for Systems Research,
Institute for Advanced Computer Studies
and Dept. of Computer Science
Univ. of Maryland
College Park, MD 20740
christos@cs.umd.edu

Eliot Siegel, M.D. and *Zenon Protopapas, M.D.*
UM Medical School and
VA Medical Center
Baltimore, MD
{eliot,zenon}@ea.net

Abstract

We examine the problem of finding similar tumor shapes. Starting from a natural similarity function (the so-called 'max morphological distance'), we showed how to lower-bound it and how to search for nearest neighbors in large collections of tumor-like shapes.

Specifically, we used state-of-the-art concepts from morphology, namely the 'pattern spectrum' of a shape, to map each shape to a point in n-dimensional space. Following [19, 36], we organized the n-d points in an R-tree. We showed that the L_∞ (= max) norm in the n-d space lower-bounds the actual distance. This guarantees no false dismissals for range queries. In addition, we developed a nearest-neighbor algorithm that *also* guarantees no false dismissals.

Finally, we implemented the method, and we tested it against a testbed of realistic tumor shapes, using an established tumor-growth model of Murray Eden[15]. The experiments showed that our method is up to 27 times faster than straightforward sequential scanning.

*His work was partially supported by the National Science Foundation under Grants IRI-9205273 and IRI-8958546, with matching funds from Empress Software Inc. and Thinking Machines Inc.

1 Introduction

This paper proposes an algorithm which rapidly searches for "similar shapes". Such an algorithm would have broad applications in electronic commerce (e.g., '*find shapes similar to a screw-driver*'), photo-journalism [20], etc., but would be particularly useful in medical imaging. During the past twenty years, the development of new modalities such as Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) have substantially increased the number and complexity of images presented to radiologists and other physicians. Additionally, the recent introduction of large scale PACS (Picture Archival and Communication Systems) has resulted in the creation of large digital image databases. A typical radiology department currently generates between 100,000 and 10,000,000 such images per year. A filmless imaging department such as the Baltimore VA Medical Center (VAMC) generates approximately 1.5 terabytes of image data annually.

An algorithm that would be able to search for similar shapes rapidly would have a number of useful applications in diagnostic imaging. Both "experts" such as radiologists and non-experts could use such a system for the following tasks:

1. Diagnosis/Classification: distinguish between a primary or metastatic (secondary) tumor based on shape and degree of change in shape over time correlating this with data about diagnoses and symptoms. Computer-aided diagnosis will be especially useful in increasing the reliability of detection of pathology, particularly when overlapping structures create a distraction or in other cases where limitations of the human visual system hamper diagnosis [37].

2. Forecasting/Time Evolution Analysis: predict the degree of aggressiveness of the pathologic process or try to distinguish a particular histology based on patterns of change in shape. In this setting, we would like to find tumors in the database with the similar history as the current tumor.

3. Data Mining: detect correlations among shapes, diagnoses, symptoms and demographic data, and thus form and test hypotheses about the development and treatment of tumors.

In all of the above tasks, the central problem is similarity matching: '*find tumors that are similar to a given pattern*' (including shape, shape changes, and demographic patient data). We mainly focus on matching similar shapes. In section 6, we discuss how our approach can be naturally extended to handle more complicated queries.

Some terminology is necessary. Following [19], we distinguish between (a) *range queries* (e.g., *find shapes that are within distance ϵ from the desirable query shape*) and (b) *nearest neighbor* queries (e.g., *find the first k closest shapes to the query shape*) An orthogonal axis of classification distinguishes between *whole-matching* and *sub-pattern matching*: In whole-matching queries, the user specifies an

$S \times S$ query image and requires images of $S \times S$ that are similar; in sub-pattern matching queries, the user specifies only a small portion and requires all the (arbitrary-size) images that contain a similar pattern

In this work we focus on whole-matching, because this is the stepping stone for the sub-pattern matching, and all the problems listed above, as discussed in section 6.

For the whole matching problem, there are two major challenges:

- How to measure the dis-similarity/distance between two shapes. In the tumor application, as well as in most other shape applications, the distance function should be invariant to rotation and translation. Moreover, we would like a function that pays attention to details at several scales, as we explain later.
- Given such a distance function, how can we do better than sequential scanning of the whole database? This faster method, however, should not compromise the correctness: it should have *no false dismissals*; that is, it should return exactly the same response set as sequential scanning would do.

Next we provide solutions to the above two challenges. This paper is organized as follows: Section 2 gives the survey. Section 3 gives an introduction to morphology and tumor-shape modeling. Section 4 presents our main result: the lower-bounding of the 'max-morphological' distance, as well as a k-nearest neighbor algorithm, without false dismissals. Section 5 gives the experiments. Section 6 discusses how to use the proposed algorithms for more general problems. Section 7 gives the conclusions.

2 Survey

2.1 Multimedia Indexing

The state of the art in multimedia indexing is based on feature extraction [36, 19]. The idea is to extract n numerical features from the objects of interest, mapping them into points in n-dimensional space. Then, any multi-dimensional indexing method can be used to organize, cluster and efficiently search the resulting points. Such methods are traditionally called *Spatial Access Methods* (SAMs). A query of the form *find objects similar to the query object* Q becomes the query *find points that are close to the query point* $\mathbf{q}$, and thus becomes a range query or nearest neighbor query. Thus, we can use the SAM to identify quickly the qualifying points, and, from them, the corresponding objects. Following [1], we call the resulting index as the '*F-index*' (for 'feature index'). This general approach has been used in several settings, such as searching for similar time-sequences [1] (e.g., trying to find quickly stock prices that move like *MacDonalds*), color images [18, 20], etc.

The major challenge is to find feature extraction functions that preserve the dis-similarity/distance between the objects as much as possible. In [1, 19] we showed that the F-index method can guarantee that there will not be any false dismissals, if the actual distance is lower-bounded by the distance in feature space.

Mathematically, let O_1 and O_2 be two objects (e.g., time sequences, bitmaps of tumors, etc.) with distance function $D_{object}()$ (e.g., the sum of squared errors) and $F(O_1)$, $F(O_2)$ be their feature vectors (e.g., their first few Fourier coefficients), with distance function $D_{feature}()$ (e.g., the Euclidean distance, again). Then we have:

Lemma 1 (Lower-Bounding) *To guarantee no false dismissals for range queries, the feature extraction function $F()$ should satisfy the following formula:*

$$D_{feature}(F(O_1), F(O_2)) \leq D_{object}(O_1, O_2) \tag{1}$$

Proof: In [19].

Thus, the search for range queries involves two steps. For a query object Q with tolerance ϵ, we

1. Discard quickly those objects whose feature vectors are too far away. That is, we retrieve the objects X such that $D_{feature}(F(Q), F(E)) < \epsilon$.
2. Apply $D_{object}()$ to discard the false alarms (the clean-up stage).

2.2 Spatial access methods

Since we rely on spatial access methods as the eventual indexing mechanism, we give a brief survey of them. These methods fall in the following broad classes: methods that transform rectangles into points in a higher dimensionality space [32]; methods that use linear quadtrees [23] [3] or, equivalently, the z-ordering [51] or other space filling curves [17] [35]; and finally, methods based on trees (R-tree [27], k-d-trees [6], k-d-B-trees [56], hB-trees [41], cell-trees [26], etc.)

One of the most promising approaches in the last class is the R-tree [27] and its numerous variants (Greene's variation [25], the R^+-tree [58], R-trees using Minimum Bounding Polygons [34], the R^*-tree [5], the Hilbert R-tree [38], etc.). We use R-trees, because they have already been used successfully for high-dimensionality spaces (10-20 dimensions [18]); in contrast, grid-files and linear quadtrees may suffer from the 'dimensionality curse'.

2.3 Tumor growth models

Our target class is a collection of images of tumor-like shapes. As a preliminary testbed, we use artificial data generated by a certain stochastic model of simulated tumor growth. Our particular model is a discrete-time version of Eden's tumor growth model [15], whose idea is illustrated in Figures 1 and 2. At time t=0, only one grid-cell is 'infected'; each infected grid-cell may infect its four non-diagonal neighbors with equal probability p at each time-tick.

On the basic Eden model, we have added the notion of East-West/North-South bias, to capture the effects of anisotropic growth patterns, due to anisotropies in the surrounding tissue (e.g., lesions shaped by their location within the lung, breast, or liver.) Thus, in our model, an infected grid-cell has probability p_{NS} to infect its North and South neighbors, and probability p_{EW} to infect its East/West ones, with p_{NS} not necessarily equal to p_{EW}.

		6	9	9				
8	5	4	7	4	6	9		
	6	3	2	1	6	8		
	9			9	9			

Figure 1: Lattice at $t = 9$. The infection time of each infected cell is marked.

t = 1 t = 5 t = 10 t = 25 t = 50 t = 100

Figure 2: Initial seed (left column) and snapshots of tumor at later time steps, with $p = .7$

The Eden model is the simplest of a hierarchy of models: in increasing generality, we have the Williams-Bjerkness model [62], the *Contact Process* [31], and the *Interacting Particle Systems* [13, 40]. See [21, 30, 55, 10] for a more comprehensive survey on tumor growth models.

2.4 Shape Representation and Matching

Shape representation is an interesting enough problem to have attracted many researchers and generated a rich array of approaches [52]. There are two closely related problems: (a) how to describe a single shape compactly and (b) how to measure the difference of two shapes, so that it corresponds to the humanly perceived difference.

With respect to representations, the most popular methods are:

- representation through 'landmarks': for example, in order to match two faces, information about the eyes, nose, etc., are extracted manually [4] or automatically. Thus, a shape is represented by a set of landmarks and their attributes (area, perimeter, relative position etc). Again, the distance between two images is the sum of the penalties for the differences of the landmarks.

- representation through numerical vectors, such as (a) samples of the 'turning angle' plot [33] (that is, the slope of the tangent at each point of the periphery, as a function of the distance traveled on the periphery from a designated starting point) (b) some coefficients of the 2-d Discrete Fourier Transform (DFT), or, more recently, the (2-d) Discrete Wavelet Transform [43] or (c) the first few moments of inertia [20, 18]. In these cases, we typically use the (weighted) Euclidean distance of the vectors.

- representation through a simpler shape, such as polygonalization [24, 50, 54, 60, 39] and *Mathematical Morphology* [63, 47, 44, 11, 7], which we shall examine in detail next.

Among them, representations based on morphology are very promising, because they have two very desirable characteristics for our applications:

- they can be easily designed to be essentially invariant to rotation and translation (= rigid motions).

- they are inherently 'multi-scale', and thus they can highlight differences at several scales, as we explain next.

The multi-scale characteristic is important, especially for tumors, because the 'ruggedness' of the periphery of a tumor contains a lot of information about it [9]. Thus, given two tumor-like shapes, we would like to examine differences at several scales before we pronounce the two shapes as 'similar'.

Even for general shapes, there exists substantial evidence that scale-space behavior is an important and highly discriminating shape "signature" [61, 42, 8].

3 Introduction to Morphology

Our goal is to choose a distance function between shapes which will be invariant to translation and rotation, and which will 'give attention' to all levels of detail. One such function is founded on ideas from the field of *Mathematical Morphology.* See [12] for a very accessible introduction. Next, we present the definitions and concepts that we need for our application. Table 3 lists the symbols and their definitions.

Some definitions are in order: Consider black-and-white images in 2-d space; the 'white' points of an image are a subset of the 2-d address space, while the background is, by convention, black. More formally, let $\mathcal{X}$ (the "shape space") be a set of compact subsets of $\Re^2$, and $\mathcal{R}$ be the group of rigid motions $R : \mathcal{X} \mapsto \mathcal{X}$.

3.1 Introduction to Morphology

Mathematical Morphology is a rich quantitative theory of shape, which incorporates a multiscale component. It has been developed mainly by Matheron [48, 49], Serra [59, 14], and their collaborators.

Symbol	Definition
$\Re$	the set of reals
$\Re_+$	the set of non-negative reals
$\mathcal{Z}$	the set of integers
$\mathcal{Z}_+$	the set of non-negative integers
$\oplus$	the operator for dilation
$\ominus$	the operator for erosion
$\circ$	the operator for Morphological Opening
$\bullet$	the operator for Morphological Closing
$\lvert X \rvert$	area of a shape X
$f_m^H(X)$	a smoothed version of X at scale n wrt structural elt H
$\mathbf{y}_X^H$	the size-distribution ($\equiv$ cumulative pattern spectrum) of X wrt structural elt H
$d(\cdot,\cdot)$	the set-difference distance between two shapes
$d^*(\cdot,\cdot)$	the floating shape distance
$d_\infty(\cdot,\cdot)$	the max-morphological distance between two shapes
$\delta_\infty(\cdot,\cdot)$	the max-granulometric distance between two shapes
a	response set size (number of actual hits)
N	database size (number of images)
n	number of features in feature space

Table 1: Symbol table

Since the 1980's, morphology and its applications have become extremely popular.

In mathematical morphology, mappings are defined in terms of a *structural element*, a "small" primitive shape (set of points) which interacts with the input image to transform it, and, in the process, extract useful information about its geometrical and topological structure. The basic morphological operators are: $\oplus, \ominus, \circ, \bullet$ (dilation, erosion, Morphological Opening, and Closing, respectively), and they are defined as follows.

Let X_h denote the translate of shape X by the vector h, and let H^s denote the symmetric of shape H with respect to the origin:

$$H^s = \{-h \mid h \in H\} \tag{2}$$

Definition 1 *The* **dilation**, *$X \oplus H^s$, of a shape $X \subset \Re^2$ by a structural element H, is defined as*

$$X \oplus H^s = \bigcup_{h \in H} X_{-h} = \left\{(x,y) \in \Re^2 \mid H_{(x,y)} \cap X \neq \emptyset\right\} \tag{3}$$

Figure 3 shows a 'butterfly' shape X, dilated by the unit circle H. Intuitively, a dilation 'blows-up' the original shape X by tracing its perimeter (and all the internal points) with a 'brush' of foot H.

We also use mH, $m \in \mathcal{Z}_+$ to denote $H \oplus H \oplus \cdots \oplus H$ ($m-1$ times), i.e., a structural element of size m. Intuitively, if the structural element H is the unit ball, then the shape mH is a ball of radius m.

Definition 2 *The* **erosion**, $X \ominus H^s$, *of a shape* $X \subset \Re^2$ *by a structural element* H, *is defined as*

$$X \ominus H^s = \bigcap_{h \in H} X_{-h} = \left\{(x,y) \in \Re^2 \mid H_{(x,y)} \subseteq X\right\} \tag{4}$$

Figure 3 shows the butterfly shape X eroded by the unit circle H, which is a dilation of the complement of X. Intuitively, an erosion deletes part of the original shape X by tracing its perimeter with an 'eraser' of foot H.

Two important composite Morphological operators are opening and closing.

Definition 3 *The* **opening**, $X \circ H$, *of a shape* $X \subset \Re^2$ *by a structural element* H, *is defined as an erosion followed by a dilation:*

$$X \circ H = (X \ominus H^s) \oplus H \tag{5}$$

Figure 3 shows the opening of shape X. Intuitively, the opening is the set of points that a brush of foot H can reach, when the brush is confined inside the shape, and is barely allowed to touch the periphery of the shape.

Definition 4 *The* **closing**, $X \bullet H$, *of a shape* $X \subset \Re^2$ *by a structural element* H, *is defined as a dilation followed by an erosion:*

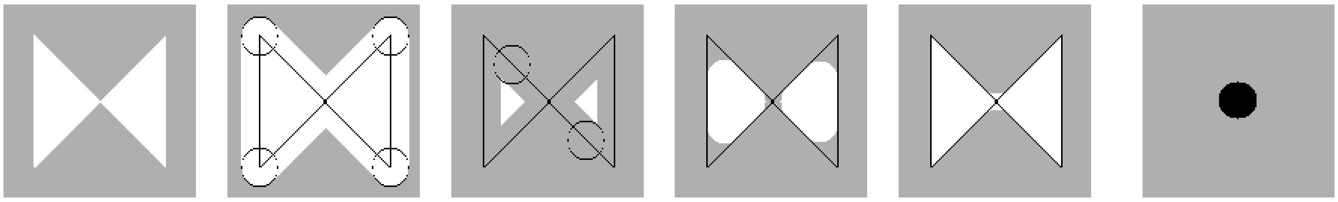
$$X \bullet H = (X \oplus H^s) \ominus H \tag{6}$$

Figure 3 shows the closing of shape X, which is the opening of the complement of X. Intuitively, the closing is the set of points that remain after an eraser of foot H sweeps the outside of the dilated X.

Thus, the opening by a circle of radius n in effect 'cuts the corners'; that is, it eliminates the protruding details of the shape X, with radius less than n. Symmetrically, the closing 'fills the concavities' of the appropriate scale.

3.2 Granulometries and the Pattern Spectrum

The concept of the *Pattern Spectrum* as a compact shape-size descriptor has been developed by Maragos [46], based on earlier seminal work on openings of sets in Euclidean spaces by Matheron [48, 49, 28, 29], who called them *Granulometries*. Serra [59, 14] and his collaborators have used Lebesgue measures of openings by a size-parameterized family of structural elements to develop shape-size sensitive measurements of shape attributes which they called *Size Distributions*.



original X — dilation $X \oplus H^s$ — erosion $X \ominus H^s$ — opening $X \circ H$ — closing $X \bullet H$ — structural element H

Figure 3: Original image (left column) and dilation, erosion, opening, closing, and structural element.

Definition 5 *The size distribution $\mathbf{y}_X^H$ of a shape $X \in \mathcal{X}$, with respect to a structural element H is defined as*

$$\mathbf{y}_X^H \triangleq \left[|f_{-M}^H(X)|, \cdots, |f_{-1}^H(X)|, |f_0^H(X)|, |f_1^H(X)|, \cdots, |f_M^H(X)|\right]^T \tag{7}$$

with

$$f_m^H(X) \triangleq \begin{cases} X \circ mH & 1 \leq m < M \\ X & m = 0 \\ X \bullet mH & -M \leq m \leq -1 \end{cases} \tag{8}$$

where H is some structural element.

Intuitively, $|f_m^H(X)|$ is the area of a smoothed version of X at scale m, i.e., for $|f_0^H(X)|$ is the area of X, $|f_1^H(X)|$ is the area of $X \circ H$, etc. In other words, the vector $\mathbf{y}_X^H$, contains measurements of the area of X at different scales, or degrees of shape smoothing, thus constituting the size distribution.

The pattern spectrum, as discussed by Maragos [46] contains exactly the same information. Its elements are backward differences of the size distribution. In other words, the size distribution can be thought of as the 'cumulative pattern spectrum'. The intuitive meaning of the pattern spectrum is the amount of detail (= additional area) that the next closing will add, or that the next (larger-radius) opening will subtract. Figure 4 shows the pattern spectrum of a circular disc of radius 5, as well as of a square of side 10, with respect to a a unit disc structural element H. Notice that the disc has only one 'spike', at m=5, while the pattern spectrum of the square has details at several scales. (Of course, the situation would be reversed, if the structural element H was the unit square).

The importance of the size distribution and the pattern spectrum is that they summarize important shape characteristics in the sense that they possess high discriminatory power, as reported in [2, 53].

3.3 Distance functions

Given two shapes X_1 and X_2, a natural distance function involves penalizing the non-common areas. Formally,

circular disc pattern spectrum square pattern spectrum

Figure 4: Image and respective pattern spectrum histograms of (a) a circular disc of radius 5, and (b) a square of side 10.

Definition 6 *Let $d(\cdot,\cdot)$ denote the area of the symmetric set difference distance measure, i.e., for $X_1, X_2 \in \mathcal{X}$*

$$d(X_1, X_2) = |X_1 \backslash X_2| = |X_1 \cup X_2| - |X_1 \cap X_2| \tag{9}$$

We can show that $d(\cdot,\cdot)$ is a distance metric over $\mathcal{X} \times \mathcal{X}$. However, we need a distance function that allows rotations and translations. This is achieved by requiring that the two shapes are first optimally aligned by allowable motions. Formally, we have a new distance function:

Definition 7 *Define the* **floating shape distance** *$d^*()$ of two shapes X_1 and X_2 as*

$$d^*(X_1, X_2) = inf_{R \in \mathcal{R}} d(X_1, R(X_2)) \tag{10}$$

where $\mathcal{R}$ is the set of rigid motions. The process of optimal alignment of two shapes is called *registration*.

The $d^*()$ distance is very natural and intuitive; it only fails in one account, namely, to consider details at several levels. Figure 5 illustrates the point: X_1 is a square, X_2 is an identical square, with a line segment coming out of its left side, and X_3 is identical to X_1, with a line segment cutting into it. At the current scale, the distance $d^*()$ among any pair of them is small. For example, if X_1 and X_2 are optimally aligned, making the two squares to coincide, then the area of the disjoint part is the area of the protruding line segment, which is negligible. However, the visual difference between the two is non-negligible. The same is true for X_1 and X_3. These counter-intuitive results can be remedied by applying the newly introduced tools of morphology: after applying a closing (see third column), we see that the protruding line segment in X_2 will make its presence more obvious. Similarly, after applying an opening (second column), the 'cut' in X_3 will become more obvious.

Thus, given any two shapes, each opening and closing will emphasize different details of their differences, resulting in a different value of $d^*()$. The question is how to combine all these scale-dependent penalties to arrive at a single number. The solution we propose is to take the *maximum* difference. More formally,

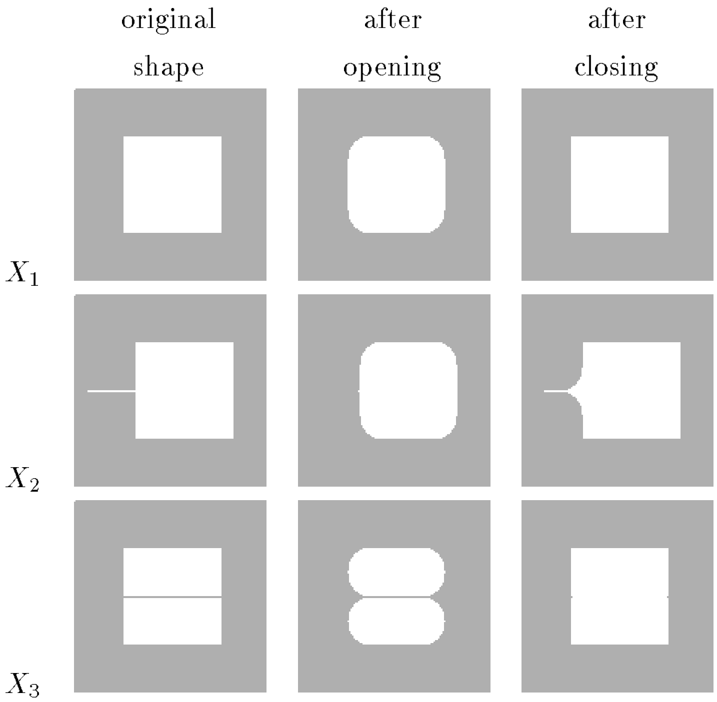


Figure 5: Three different scales of shapes X_1, X_2 and X_3

Definition 8 *Define the* **Max Morphological Distance**

$$d_\infty^H : \quad \mathcal{X} \times \mathcal{X} \mapsto \Re_+ \tag{11}$$

as

$$d_\infty^H(X_1, X_2) \stackrel{\triangle}{=} \max_{-M \le m \le M} d^* \left(f_m^H(X_1), f_m^H(X_2) \right) \tag{12}$$

with $f_m^H(X)$ defined in Eq. 8.

For the remainder we assume some fixed structural element H (e.g., the unit ball), and we drop these indices.

The intuitive meaning of the $d_\infty()$ distance function is the following:

- compute $d^*()$, that is take the two shapes X_1 and X_2, align them optimally, and compute the area of the disjoint parts
- take their closings using a disk of radius 1, 2, ... M; in each case, compute the $d^*()$ of the resulting shapes

- do the same for openings, with a disk of radius 1, 2, ... M
- pick the maximum difference, and report it as the distance of the two shapes.

Lemma 2 *The function d_∞ is indeed a distance metric between elements of $\mathcal{X}$.*

Proof: See Appendix A.

4 Problem Definition - Proposed solution

The problem we focus on is the design of fast searching methods that will operate on the tumor database to locate the most similar object to the query object. The (dis-)similarity is measured by the max-morphological distance (Eq. 12). We focus on both range queries as well as on nearest-neighbor queries. We have three obstacles to overcome:

- what features to use (i.e., how to map tumor-shapes into n-d points)
- how to prove that the above mapping is contractive, that is, it obeys the Lower-Bounding Lemma (Lemma 1).
- how to use the resulting F-index on the feature space, so that we can answer nearest-neighbor queries with respect to the actual distance (as opposed to the distance in feature space)

Next we present the proposed solutions to these three problems.

4.1 Features

Our goal is to derive features that will capture a lot of the shape information, that will be rotation and translation invariant, and that will lead to a feature-distance function that fulfills the Lower-Bounding Lemma. Given the success of the pattern spectrum as a means to capture shape information [2, 46, 45, 63], we started with its coefficients as good features. More specifically, we use the coefficients $\mathbf{y}_X$ of the size distribution (Eq. 7), which contains exactly the same information as the pattern spectrum.

Thus, we 'penalize' two shapes for differences at several scales. The question is, what is the best way to combine the penalties of each scale? A natural choice is to pick the maximum among the penalties. This is identical to the L_∞ norm of the two feature vectors, and it leads to the following distance function:

Definition 9 *Define the* **Max Granulometric Distance** $\delta_\infty()$ *of two shapes X_1, X_2 as*

$$\delta_\infty^H(X_1, X_2) = \max_{-M \leq m \leq M} |y_{X_1}(m) - y_{X_2}(m)| \tag{13}$$

4.2 Lower-Bounding

Our next challenge is to show that the distance in feature space (i.e., the max-granulometric distance $\delta_\infty()$) lower-bounds the actual distance $d_\infty()$. This is necessary to guarantee no false dismissals.

Lemma 3 (Max-Morphological Distance Bounding) *The max-granulometric distance $\delta_\infty()$ lower-bounds the max-morphological distance $d_\infty()$:*

$$\delta_\infty(X_1, X_2) \leq d_\infty(X_1, X_2), \quad \forall X_1, X_2 \in \mathcal{X} \tag{14}$$

Proof: Observe that

$$d^*(X_1, X_2) \geq ||X_1| - |X_2|| \tag{15}$$

with equality achieved if and only if there exists some rigid motion $R \in \mathcal{R}$ which brings all points in X_2 (or X_1) in registration with points in X_1 (X_2, respectively). Then

$$d^*(f_m(X_1), f_m(X_2)) \geq ||f_m(X_1)| - |f_m(X_2)|| \tag{16}$$

and

$$\max_{-M \leq m \leq M} d^*(f_m(X_1), f_m(X_2)) \geq \max_{-M \leq m \leq M} ||f_m(X_1)| - |f_m(X_2)|| \tag{17}$$

Recall that the left-hand side is the definition of d_∞ and the right-hand side is the definition of δ_∞. Thus, the proof is complete. **QED**

The latter result guarantees *completeness* of range queries.

By keeping the dimensionality of the spectra space small (say $M = 5 \mapsto 2M + 1 = 11$ features) we can use an F-index which, as we show later, results in considerably faster access of large image databases.

4.3 Nearest-neighbor algorithm

We have just described a good set of features, namely, the $2M + 1$ entries of the cumulative pattern spectrum ($\equiv$ size distribution) of an image, as well as proved that the resulting $\delta_\infty()$ distance lower-bounds the actual distance. Thus, the resulting 'F-index' will guarantee no false dismissals upon range queries.

The next problem is to find the k-nearest neighbors of a query image, given that the images of the collection have already been mapped into n-d points and organized in a SAM. Algorithms to find the k-nearest neighbors of a given point already exist, using a branch-and-bound algorithm [22], and have been applied to R-trees recently [57].

However, the SAM search will return the k-nearest neighbors with respect to the *max-granulometric* distance $\delta_\infty()$, as opposed to the *max-morphological* distance $d_\infty()$ that we really want.

Here we describe the algorithm that finds the actual k-nearest neighbors in *any* F-index where the Lower-Bounding Lemma (Lemma 1) holds:

Algorithm 1 (k-nn)

```
/*uses an F-index to return the k-nearest neighbors according to the actual distance*/
/*input:  query object Q; # of nn k*/
/*output:  k-nearest objects X_1, X_2, ..., X_k*/
1.  Search the SAM to find the k-nn wrt the feature distance D_feature (δ_∞ in our case).
2.  Compute the actual distance D_object(Q,X) (d_∞(Q,X) in our case) for all the k
    candidates X, and return the maximum ε_max.
3.  Issue a range query with the feature vector F(Q) of the query object Q and ε_max
    on the SAM, retrieve all the actual objects, compute their actual distances
    D_object() from Q, and pick the nearest k.
```

We prove that the above algorithm will give no false dismissals:

Lemma 4 *Algorithm 1 guarantees no false dismissals for k-nn queries.*

Proof: Let X_k be the shape returned as the k-th nearest neighbor by the algorithm (step 3); let Y be a object that is the j-th nearest neighbor (with $j < k$). Then

$$D_{object}(Q,Y) \leq D_{object}(Q,X_k) \tag{18}$$

Suppose for a moment that the algorithm *fails* to return Y. We will show that this leads to a contradiction: After the range query is issued in step 3, all eliminated shapes (including Y) must have *feature* distance $D_{feature}$ greater than ϵ_{max}, that is

$$D_{feature}(Q,Y) \quad > \quad \epsilon_{max} \tag{19}$$

and, from the Lower-Bounding Lemma (Lemma 1),

$$D_{object}(Q,Y) \geq D_{feature}(Q,Y) \ > \ \epsilon_{max} \tag{20}$$

However,

$$\epsilon_{max} \geq D_{object}(Q,X_k) \tag{21}$$

(since X_k was obviously retrieved by step 3). Combining the last two inequalities, we obtain

$$D_{feature}(Q,Y) > D_{object}(Q,X_k) \tag{22}$$

which contradicts Eq. 18. **QED**

5 Experiments

To test the speed of our approach, we implemented our method and ran experiments. Next we describe the set up, as well as our results and observations, for range queries and for nearest neighbor queries.

Testbed: We generated 20,000 black-and-white 128×128 pixel images of tumor shapes based on Eden's model of tumor growth. Each image contains a tumor that either (a) grows uniformly in all eight directions, (b) is biased vertically and horizontally with slower growth along the diagonals, (c) is restricted along one direction (blocked by a barrier such as a bone), or (d) is restricted along two directions (cone-shaped). Within each of these four classes of growth, we vary

- the number of iterations, which affects the size of the tumor;
- the directional bias (p_{NS}/p_{EW}), which affects the ratio of height to width.

We performed experiments for varying database sizes N, by choosing N images among the 20,000 ones.

Competing methods:

- straightforward sequential scan: This is the simple brute force algorithm. Given a query image, the algorithm goes through all images in the database and computes its max-morphological distance from the query image, keeping track of the images with the minimum distance. Because the algorithm is comparing images on a pixel-by-pixel basis, it is extremely inefficient.
- F-index with an n-d R-tree: On insertion, the size distribution, or cumulative pattern spectrum $\mathbf{y}_X$, of each image of the database has been computed and the n-dimensional vector has been inserted into an R-tree. Given a query image, its cumulative pattern spectrum is computed, and is then submitted for a range or k-nearest neighbor search in the R-tree, as we discussed before.

Measurements: We are interested in the response time, that is, the time until the last actual hit is returned to the user (*after* the system has discarded possible false alarms). For some small settings we report actual (*wall-clock*) time, from the `time` utility of $UNIX^{TM}$. However, the time t_{mm} to compute the max-morphological distance between two images is high (t_{mm} = 12.69 sec on average) and shows small variance (standard deviation of 0.036sec). Thus, to accelerate the execution of experiments on large databases, we `time` all the other steps of the algorithms involved, and simply 'charge' a delay of t_{mm} seconds for each max-morphological distance computation that we omit.

Hardware and software The methods were implemented in 'C' and *KornShell* under $UNIX^{TM}$. The experiments ran on a dedicated Sun SPARCstation 5 with 32Mb of main memory, running SunOS

4.1.3. The disk drive was a FUJITSU M2266S-512 model 'CRANEL-M2266SA' with minimum positioning time of 8.3 ms and maximum positioning time of 30ms.

We present experiments on range queries as well as nearest neighbor queries. We also give some pictures of the images that have been returned.

5.1 Range queries

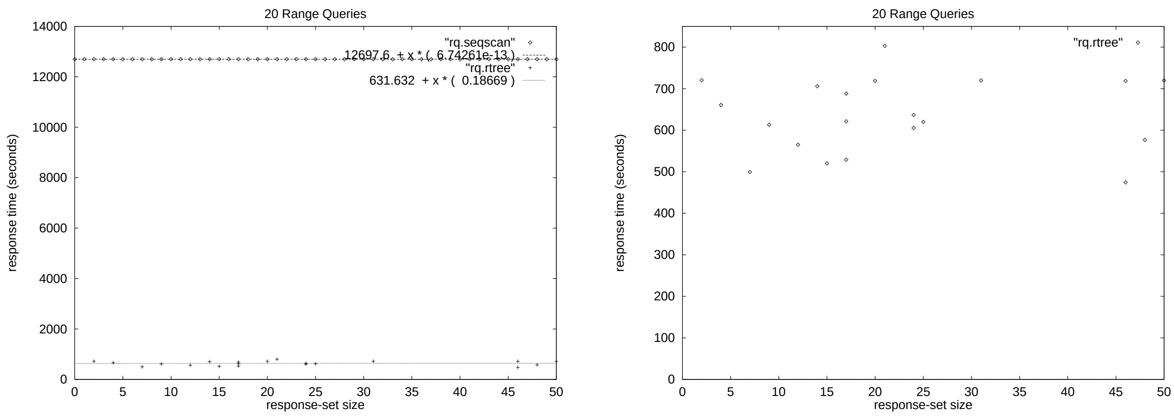


Figure 6: Response time vs. response-set size a for range queries (a) with seq. scanning (b) without seq. scanning

We asked 20 queries on a database of $N = 1,000$ images for both methods. Figure 6(a) plots the response time for the F-index as a function of the response-set size a (i.e., number of actual hits, after the false-hits have been eliminated), for several values of the tolerance. It also shows the response time for sequential scanning for comparison, which is estimated to take 12697.6 seconds. Figure 6(b) shows only the proposed F-index method, in more detail. Figure 7 displays this data in a table. The performance gap between the two methods is very large: our method achieves 15-fold to 27-fold savings.

5.2 Nearest Neighbor Queries

We ran queries with k=2,3,4, and 10 for several N. Figure 8 shows (a) the results of k-nearest neighbor queries with $k = 10$, for varying N, for the F-index method compared to the sequential scan algorithm, and (b) the results of k=2,3,4, and 10 for the F-index method only. Each data point represents the average response time (in seconds) for 100 random query images taken from the database.

Figure 9(a) shows response time vs. k (= 2,3,4,10) for $N = 10,000$ and $N = 20,000$ for both methods. Figure 9(b) shows response time vs. k for $N = 10,000$ and $N = 20,000$ for the proposed method only.

response-set size	seq scan (1)	F-index (2)	ratio
a	time (sec)	time(sec)	(1) : (2)
2	12697.6	720.25	17.63
4	12697.6	660.83	19.21
7	12697.6	499.15	25.44
9	12697.6	613.03	20.71
12	12697.6	564.96	22.48
14	12697.6	705.85	17.99
15	12697.6	520.10	24.41
17	12697.6	528.76	24.01
17	12697.6	621.38	20.43
17	12697.6	688.00	18.46

response-set size	seq scan	F-index	ratio
a	time (sec)	time(sec)	(1) : (2)
20	12697.6	718.83	17.66
21	12697.6	803.25	15.81
24	12697.6	605.33	20.98
24	12697.6	636.68	19.94
25	12697.6	620.13	20.48
31	12697.6	720.00	17.64
46	12697.6	474.45	26.76
46	12697.6	718.65	17.67
48	12697.6	576.88	22.01
50	12697.6	719.95	17.64

Figure 7: Table of (a) response-set size a (b) response time for seq. scan (c) response time for F-index (d) ratio of seq. scan time to F-index algorithm time

Again, each data point represents the average response time over 100 queries.

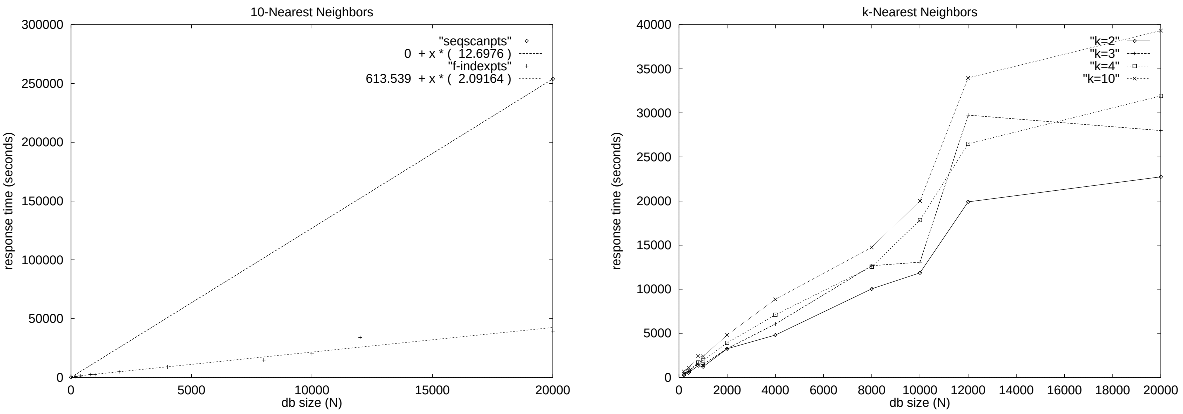
The observations are the following:

- Our proposed algorithm is 3-7 times faster than sequential scanning, even for a large value of k (e.g., 10) for the nearest neighbors;
- The savings of the proposed method compared to sequential scan seems to increase with the database size N;
- Response time grows slowly with k.

5.3 Sample output

Here we illustrate that the max-morphological distance function $d_\infty()$ seems to capture the perceptual distance between two shapes. Figure 10 shows a few query images (left column) and their corresponding 3-nearest neighbors according to the max-morphological distance. Since the query images were drawn from the database, the first nearest neighbor is identical to the query shape (which is a sanity check for our algorithms and implementations). Notice how similar the other 2 nearest neighbors are, for both query shapes.

Finally, Figure 11 illustrates the realism of Eden's model. Figure 11(a) shows the whole mamogram, highlighting the tumor shape; (b) shows the tumor magnified; (c) shows the tumor shape after it has been thresholded (and thus becomes a black-and-white image); and (d) shows the nearest neighbor that was retrieved from our testbed of 20,000 synthetic tumor shapes. The similarity of the real tumor with its synthetic nearest neighbor is striking.



(a) (b)

Figure 8: (a) Response time vs. db size(N), for $k = 10$ nn queries for both seq. scan and F-index (b) Response time vs. N for k=2,3,4,and 10.

6 Discussion

As mentioned in the introduction, we focus on whole-matching queries for 2-d shapes. In this section we discuss how to use the proposed algorithms as a stepping stone to more complicated problems, specifically (a) sub-pattern matching (b) time-evolution pattern matching and (c) matching on additional data (demographic, etc.)

1. Sub-pattern matching: The problem is to detect a similar $s \times s$ shape within a collection of $S \times S$ bitmaps ($s \leq S$). The idea is to use sliding, overlapping windows, as we suggested in [19] for time sequences. That is, we assume that the smallest query will be at least of size $s_{min} \times s_{min}$ and we pre-process each $S \times S$ data image by considering all the possible $s_{min} \times s_{min}$ windows in it; then we extract n features from each window, mapping each window into a point in n-d space. Thus, the problem is reduced to the whole-matching problem, which we have just solved. Algorithms to minimize the space requirements of the feature vectors, as well as to handle queries of arbitrary sizes, have been suggested in [19] for 1-d time sequences; all these techniques can be easily extended to 2-d images (and, in fact, arrays of arbitrary dimensionality).

2. Time-evolution: Currently, we considered static, 2-d images. By introducing time, we have sequences of images, which can be viewed as 3-d arrays, if we envision time as one more dimension. We could extend our 2-d algorithms to handle such 3-d shapes: The morphological concepts that we mentioned before (opening, closing, pattern spectrum) can be trivially extended to 3-d shapes. Therefore, our 2-d algorithms can be trivially extended to accommodate temporal patterns.

db size	seq scan	F-index	ratio
N	time (sec)	time(sec)	(1) : (2)
200	2539.52	675.74	3.76
400	5079.04	1084.19	4.68
800	10158.08	2422.04	4.19
1000	12697.60	2375.44	5.35
2000	25395.20	4814.59	5.27
4000	50790.40	8855.21	5.74
8000	101580.80	14743.66	6.89
10000	126976.00	20002.95	6.35
12000	152371.20	33973.27	4.49
20000	253952.00	39340.32	6.46

Table 2: Values for 10-nn queries: (a) db size N (b) response time for seq. scan (c) response time for the proposed algorithm (d) ratio of seq. scan time to proposed algorithm time

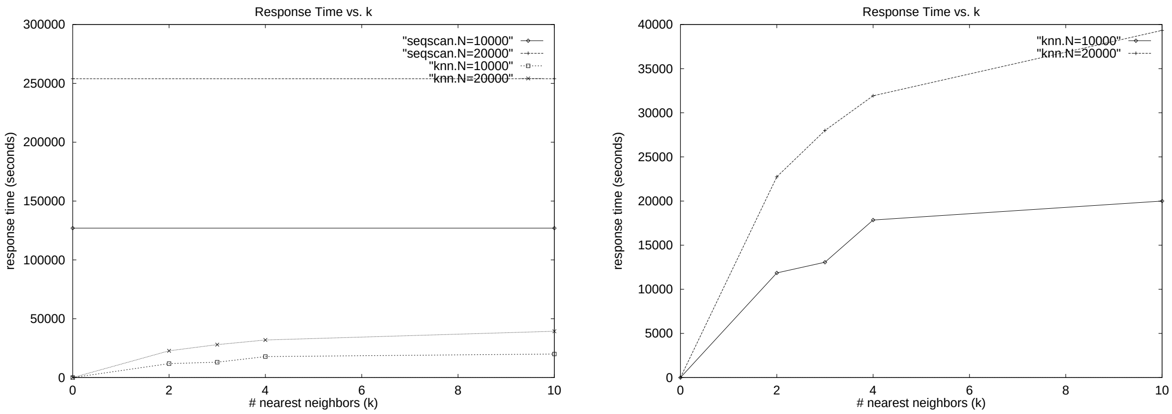
3. Inclusion of non-spatial data: The diagnosis for a tumor-like shape has to take into account not only the shape, but also additional information: e.g., whether/how close is it to an organ (lung, liver, etc.), demographic characteristics of the patient (age, gender, etc.) Our algorithm can easily incorporate such data by augmenting the n-dimensional feature vectors with a few more numerical features (age of patient, distance of tumor from liver, etc). Thus, our feature space will have n spatial features (the size-distribution coefficients of the tumor shape) plus n' non-spatial features; the rest of our algorithms will remain intact.

7 Conclusions

We have focused on fast searching for similar shapes with emphasis on tumor-like shapes. To solve the problem, we used a multi-scale distance function, the so-called 'max-morphological' distance. This distance function is based on modern signal processing methods, and specifically mathematical morphology. The distance is invariant to rotations and translations, and gives similar attention to all levels of detail ('scales'). From the database end, we used the 'Feature index' (F-index) approach [1, 19], which is the latest in multimedia indexing.

The main contribution of this work is that it manages to couple the max-morphological distance with the F-index. This is done by using the coefficients of the size distribution as features, and by showing that the L_∞ (=max) distance in the resulting feature space lower-bounds the max-morphological distance. Given the Lower-Bounding Lemma (Lemma 1), this guarantees no false dismissals for range queries.

Additional contributions are the following:



(a) (b)

Figure 9: Response time vs. k for $N = 10,000$ and $N = 20,000$ (a) with seq. scanning (b) without seq. scanning

- The design and implementation of a nearest neighbor algorithm on an F-index, which provably guarantees no false dismissals
- The implementation of the proposed method and the experimentation on a synthetic, but realistic, database of tumor-like shapes. There, the proposed method achieved dramatic speed-ups (up to 27-fold) over the straightforward sequential scanning.
- A survey of the literature on tumor-growth modeling. Given the administrative and patient-privacy problems in obtaining large collections of real X-rays, we believe that the above citations and algorithms will provide an excellent starting point for database researchers who want to work in this area and need a realistic testbed for their algorithms.
- The introduction of the basic morphological concepts (dilation, erosion, pattern spectrum, etc.) in an intuitive way, so that these powerful tools will become more accessible to database researchers.

Future research should focus on applications and extensions of the proposed method for several modalities including Computed Radiography, CT, MRI, Ultrasound, and Nuclear Medicine, as well as non-radiologic images in areas such as dermatology and pathology. The algorithm could be incorporated for general use in a large-scale PACS and serve as a powerful tool for both diagnostic and research purposes.

References

[1] Rakesh Agrawal, Christos Faloutsos, and Arun Swami. Efficient similarity search in se-

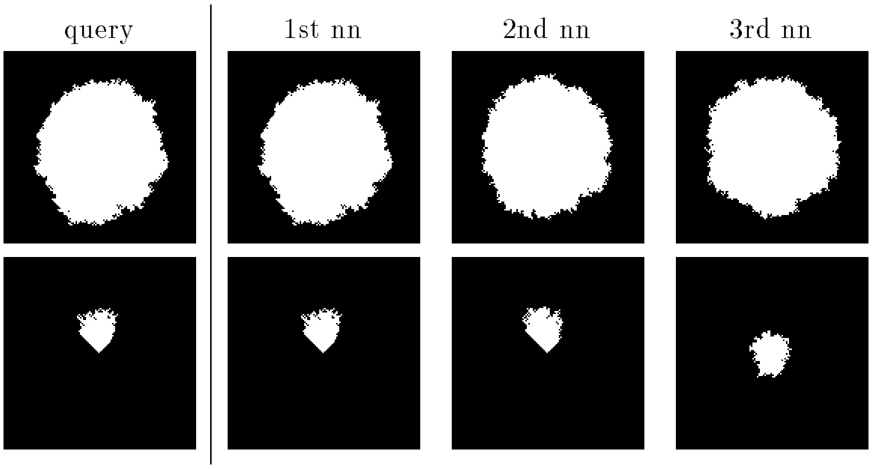


Figure 10: Query images (left column) and their nearest neighbors, according to the max-morphological distance.

quence databases. In *Foundations of Data Organization and Algorithms (FODO) Conference*, Evanston, Illinois, October 1993. also available through anonymous ftp, from olympos.cs.umd.edu: ftp/pub/TechReports/fodo.ps.

[2] V. Anastassopoulos and A.N. Venetsanopoulos. Classification properties of the pecstrum and its use for pattern identification. *Circuits, Systems and Signal Processing*, 10(3), 1991.

[3] Walid G. Aref and Hanan Samet. Optimization strategies for spatial query processing. *Proc. of VLDB (Very Large Data Bases)*, pages 81–90, September 1991.

[4] Jeffrey R. Bach, Santanu Paul, and Ramesh Jain. A visual information management system for the interactive retrieval of faces. *IEEE Trans. on Knowledge and Data Engineering (TKDE)*, 5(4):619–628, August 1993.

[5] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger. The r*-tree: an efficient and robust access method for points and rectangles. *ACM SIGMOD*, pages 322–331, May 1990.

[6] J.L. Bentley. Multidimensional binary search trees used for associative searching. *CACM*, 18(9):509–517, September 1975.

[7] S. Beucher. Digital skeletons in Euclidean and geodesic spaces. *Signal Processing*, 38:127–141, 1994.

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

(a) full mamogram (b) magnification of tumor (c) thresholded tumor (d) synthetic nn

Figure 11: (a) a real tumor within a mamogram (b) magnification of the tumor (c) its black-and-white (thresholded) version (d) the most similar synthetic tumor

[9] Christina J. Burdett, Harold G. Longbotham, Mita Desai, Walter B. Richardson, and John F. Stoll. Nonlinear indicators of malignancy. *Proc. SPIE 1993 - Biomedical Image Processing and Biomedical Visualization*, 1905 (part two of two):853–860, February 1993.

[10] N. Cressie and F.L. Hulting. A spatial statistical analysis of tumor growth. Preprint 89-1, Iowa State University, Ames, IA, 1989.

[11] E. R. Dougherty, Y. Chen, J. Hornack, and S. Totterman. Detection of osteoporosis by morphological granulometries. In *Visual Communications and Image Processing '92*, volume 1660 of *SPIE Proceedings*, San Jose, 1992. February.

[12] Edward R. Dougherty. *An Introduction to Morphological Image Processing*, volume TT9. A publication of SPIE - the Int. Society for Optical Engineering (SPIE Press), 1992.

[13] R. Durrett. *Lecture Notes on Particle Systems and Percolation.* Wadsworth, Pacific Groove, CA, 1988.

[14] J. Serra Ed. *Image Analysis and Mathematical Morphology, vol. 2, Theoretical Advances.* Academic, San Diego, 1988.

[15] M. Eden. A two-dimensional growth process. In *Proc. of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, CA, 1961. J. Neyman (ed.).

[16] Georgios Evangelidis, David Lomet, and Betty Salzberg. The hbp-tree: a modified hb-tree supporting concurrency, recovery and node consolidation. *VLDB*, pages 551–561, September 1995.

[17] C. Faloutsos and S. Roseman. Fractals for secondary key retrieval. *Eighth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 247–252, March 1989. also available as UMIACS-TR-89-47 and CS-TR-2242.

[18] Christos Faloutsos, William Equitz, Myron Flickner, Wayne Niblack, Dragutin Petkovic, and Ron Barber. Efficient and effective querying by image content. *J. of Intelligent Information Systems*, 3(3/4):231–262, July 1994.

[19] Christos Faloutsos, M. Ranganathan, and Yannis Manolopoulos. Fast subsequence matching in time-series databases. *Proc. ACM SIGMOD*, pages 419–429, May 1994. 'Best Paper' award; also available as CS-TR-3190, UMIACS-TR-93-131.

[20] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jon Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, and Peter Yanker. Query by image and video content: the qbic system. *IEEE Computer*, 28(9):23–32, September 1995.

[21] W.F. Forbes and R.W. Gibberd. Mathematical models of carcinogenesis: A review. *The Mathematical Scientist*, 9:95–110, 1984.

[22] Keinosuke Fukunaga and Patrenahalli M. Narendra. A branch and bound algorithm for computing k-nearest neighbors. *IEEE Trans. on Computers (TOC)*, C-24(7):750–753, July 1975.

[23] I. Gargantini. An effective way to represent quadtrees. *Comm. of ACM (CACM)*, 25(12):905–910, December 1982.

[24] Gary and Mehrotra. Shape similarity-based retrieval in image database systems. *SPIE 92*, 1662:2–8, 1992.

[25] D. Greene. An implementation and performance analysis of spatial data access methods. *Proc. of Data Engineering*, pages 606–615, 1989.

[26] O. Gunther. The cell tree: an index for geometric data. Memorandum No. UCB/ERL M86/89, Univ. of California, Berkeley, December 1986.

[27] A. Guttman. R-trees: a dynamic index structure for spatial searching. *Proc. ACM SIGMOD*, pages 47–57, June 1984.

[28] A. Haas, G. Matheron, and J. Serra. Morphologie mathématique et granulométries en place. 1e Partie. *Annales des Mines*, XI:735–753, 1967.

[29] A. Haas, G. Matheron, and J. Serra. Morphologie mathématique et granulométries en place. 2e Partie. *Annales des Mines*, XII:767–782, 1967.

[30] F. Hanson and C. Tier. A stochastic model of tumor growth. *Mathematical Biosciences*, 61:73–100, 1982.

[31] T.E. Harris. Contact interactions on a lattice. *Annals of Probability*, 2:969–988, 1974.

[32] K. Hinrichs and J. Nievergelt. The grid file: a data structure to support proximity queries on spatial objects. *Proc. of the WG'83 (Intern. Workshop on Graph Theoretic Concepts in Computer Science)*, pages 100–113, 1983.

[33] Berthold Horn. *Robot Vision*. The MIT electrical engineering and computer science series. MIT Press, Cambridge, Mass., 1986.

[34] H. V. Jagadish. Spatial search with polyhedra. *Proc. Sixth IEEE Int'l Conf. on Data Engineering*, February 1990.

[35] H.V. Jagadish. Linear clustering of objects with multiple attributes. *ACM SIGMOD Conf.*, pages 332–342, May 1990.

[36] H.V. Jagadish. A retrieval technique for similar shapes. *Proc. ACM SIGMOD Conf.*, pages 208–217, May 1991.

[37] T. Ji, M. Sundareshan, and H. Roehrig. Adaptive image contrast enhancement based on human visual properties. *IEEE Transactions on Medical Imaging*, 13(4), December 1994.

[38] Ibrahim Kamel and Christos Faloutsos. Hilbert r-tree: an improved r-tree using fractals. In *Proc. of VLDB Conference,*, pages 500–509, Santiago, Chile, September 1994.

[39] Y. Kurozumi and W.A. Davis. Polygonal approximation by the minimax method. *Computer Graphics Image Processing*, 19:248–264, 1982.

[40] T.M. Liggett. *Interacting Particle Systems*. Springer, New York, NY, 1985.

[41] David B. Lomet and Betty Salzberg. The hb-tree: a multiattribute indexing method with good guaranteed performance. *ACM TODS*, 15(4):625–658, December 1990.

[42] D.C. MacEnany and J.S. Baras. Scale-space polygonalization of target silhouettes and applications to model-based ATR. In *Proc. Second ATR Systems and Technology Conf., Center for Night Vision and Electro-Optics, Ft. Belvoir, VA*, pages 223–247, Mar. 1992. Vol. II.

[43] S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. PAMI*, 11(7):2091–2110, 1989.

[44] P. Maragos. Morphological skeleton representation and coding of binary images. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34:1228–1244, 1986.

[45] P. Maragos. Morphology-based symbolic image modeling, multi-scale nonlinear smoothing, and pattern spectrum. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Ann Arbor*, page ??, june 1988.

[46] P. Maragos. Pattern spectrum and multiscale shape representation. *IEEE Transactions on Patt. Anal. Mach. Intell.*, 11(7):701–716, July 1989.

[47] P. Maragos and R.W. Schafer. Morphological skeleton representation and coding of binary images. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 34:1228–1244, 1986.

[48] G. Matheron. *Eléments pour une Théorie des Milieux Poreux.* Masson, Paris, 1967.

[49] G. Matheron. *Random Sets and Integral Geometry.* Wiley, New York, 1975.

[50] U. Montanari. A note on minimal length polygonal approximation to a digitized contour. *Commun. ACM*, 13:41–47, Jan. 1970.

[51] J. Orenstein. Spatial query processing in an object-oriented database system. *Proc. ACM SIGMOD*, pages 326–336, May 1986.

[52] T. Pavlidis. Algorithms for shape analysis of contours and waveforms. *IEEE T. PAMI*, PAMI-2:301–312, 1980.

[53] S. Pong and A.N. Venetsanopoulos. Rotationally invariant pecstrum: An object recognition descriptor based on mathematical morphology. *Circuits, Systems and Signal Processing*, 11(4):455–492, 1992.

[54] U.E. Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics Image Processing*, 1:244–256, 1972.

[55] R. Ransom. A computer model of cell clone growth. *Simulation*, 28:189–192, 1977.

[56] J.T. Robinson. The k-d-b-tree: a search structure for large multidimensional dynamic indexes. *Proc. ACM SIGMOD*, pages 10–18, 1981.

[57] Nick Roussopoulos, Steve Kelley, and F. Vincent. Nearest Neighbor Queries. *Proc. of ACM-SIGMOD*, pages 71–79, May 1995.

[58] T. Sellis, N. Roussopoulos, and C. Faloutsos. The r+ tree: a dynamic index for multi-dimensional objects. In *Proc. 13th International Conference on VLDB*, pages 507–518, England,, September 1987. also available as SRC-TR-87-32, UMIACS-TR-87-3, CS-TR-1795.

[59] J. Serra. *Image Analysis and Mathematical Morphology.* Academic, New York, 1982.

[60] J. Slansky and V. Gonzalez. Fast polygonal approximation of digitized curves. *Pattern Recognition*, 12:327–331, 1980.

[61] R. van den Boomgaard and A. W. M. Smeulders. Towards a morphological scale-space theory. In Y-L. O, A. Toet, D. Foster, H.J.A.M. Heijmans, and P. Meer, editors, *Shape in Picture: Mathematical Description of Shape in Grey-level Images*, pages 631–640, 1994.

[62] T. Williams and R. Bjerknes. Stochastic model for abnormal clone spread through epithelial basal layer. *Nature*, 236:19–21, 1972.

[63] Z. Zhou and A. N. Venetsanopoulos. Morphological skeleton representation and shape recognition. In *Proc. of the IEEE second Int. Conf. on ASSP, New York*, pages 948–951, 1988.

A Proof that d_∞ is a distance metric

Lemma: The function d_∞ is indeed a distance metric between elements of $\mathcal{X}$
Proof: We start with d_∞. We have to verify the following properties:

- $d_\infty(X_1, X_2) \geq 0$
- $d_\infty(X_1, X_2) = 0 \; iff \; X_1 = X_2$: trivial because m can equal 0.
- $d_\infty(X_1, X_2) = d_\infty(X_2, X_1)$: follows from symmetry of $\mathcal{R}$, and $d(\cdot, \cdot)$.
- Triangle Inequality: $d_\infty(X_1, X_2) \leq d_\infty(X_1, X) + d_\infty(X, X_2)$.

$$d_\infty(X_1, X_2) \leq \max_{-M \leq m \leq M} \left\{ d^*(f_m^H(X_1), f_m^H(X)) + d^*(f_m^H(X), f_m^H(X_2)) \right\} \tag{23}$$

$$\leq \max_{-M \leq m \leq M} d^*(f_m^H(X_1), f_m^H(X)) + \max_{-M \leq m \leq M} d^*(f_m^H(X), f_m^H(X_2)) \tag{24}$$

$$= d_\infty(X_1, X) + d_\infty(X, X_2) \tag{25}$$

Thus, the proof is complete. **QED**

Contents