

Geometric Matching under Noise: Combinatorial Bounds and Algorithms

PIOTR INDYK*

RAJEEV MOTWANI†

SURESH VENKATASUBRAMANIAN‡

Department of Computer Science
Stanford University
Stanford, CA 94305

{indyk,rajeev,suresh}@cs.stanford.edu

Abstract

In geometric pattern matching, we are given two sets of points P and Q in d dimensions, and the problem is to determine the rigid transformation that brings P closest to Q , under some distance measure. More generally, each point can be modelled as a ball of small radius, and we may wish to find a transformation approximating the closest distance between P and Q . This problem has many applications in domains such as computer vision and computational chemistry

In this paper we present improved algorithms for this problem, by allowing the running time of our algorithms to depend not only on n , (the number of points in the sets), but also on Δ , the diameter of the point set. The dependence on Δ also allows us to effectively process point sets that occur in practice, where diameters tend to be small ([EVW94]). Our algorithms are also simple to implement, in contrast to much of the earlier work.

To obtain the above-mentioned results, we introduce a novel discretization technique to reduce geometric pattern matching to combinatorial pattern matching. In addition, we address various generalizations of the classical problem first posed by Erdős: "Given a set of n points in the plane, how many pairs of points can be exactly a unit distance apart?". The combinatorial bounds we prove enable us to obtain improved results for geometric pattern matching and may have other applications.

1 Introduction

Geometric point set matching in two and three dimensions is a well-studied family of problems with application to areas such as computer vision [MNL98], pattern recognition [CGH⁺93, HKK92] and computational

chemistry [FHK⁺96, FKL⁺97, NFWN94]. Given some choice of a space \mathcal{G} of transformations and a distance measure $d(P, Q)$ for two point sets P and Q in d -dimensional Euclidean space, we can formulate the basic problem as follows: Determine the transformation $T \in \mathcal{G}$ that brings P closest to Q , i.e. that minimises $d(T(P), Q)$. More formally,

Problem 1.1. (PATTERN MATCHING (PM)) Given $\epsilon > 0$, and point sets P and Q , where $|P| = k$ and $|Q| = n$ with $k \leq n$, find a transformation $T \in \mathcal{G}$ for which $d(T(P), Q) \leq \epsilon$.

A generalized version of this where we require only that a portion of P is near some portion of Q , is:

Problem 1.2. (LARGEST COMMON POINT-SET (LCP)) Given $\epsilon > 0$, $K > 0$, and point sets P and Q of size k and n , find a transformation $T \in \mathcal{G}$ and a set $P' \subset P$ of size $|P'| \geq K$ such that $|d(T(P'), Q)| \leq \epsilon$.

In this paper, we will restrict ourselves to point sets in \mathbb{R}^2 and \mathbb{R}^3 , and will restrict \mathcal{G} to be the space of isometries. Note that any isometry can be represented as a composition of a rotation and a translation (and possibly a reflection, which can be ignored without loss of generality). We define the following three distance measures¹.

Exact d_E : $d_E(P, Q) = 0$ if $P \subset Q$, otherwise it is 1.

Hausdorff d_H : $d_H(P, Q) = \max_{p \in P} \min_{q \in Q} d(p, q)$, where $d(p, q)$ is the Euclidean distance.

Bottleneck d_M : $d_M(P, Q) = \min_{\pi} \max_{p \in P} d(p, \pi(p))$, where π ranges over all permutations of $\{1, \dots, k\}$.

¹Technically, none of these three measures satisfy the property of being a metric.

*Supported by a Stanford Graduate Fellowship and NSF Award CCR-9357849, with matching funds from IBM, Mitsubishi, Schlumberger Foundation, Shell Foundation, and Xerox Corporation.

†Supported by an Alfred P. Sloan Research Fellowship, an IBM Faculty Partnership Award, an ARO MURI Grant DAAH04-96-1-0007, and NSF Young Investigator Award CCR-9357849, with matching funds from IBM, Mitsubishi, Schlumberger Foundation, Shell Foundation, and Xerox Corporation.

‡Supported in part by a grant from Pfizer Central Research.

In this paper, we study PM and LCP in two and three dimensions under the Hausdorff and bottleneck measures. We also study PM and LCP under “sum” versions of the two measures, where $\max_{p \in P}$ in the definition is replaced by $\sum_{p \in P}$. These measures will be referred to as Σ -Hausdorff and Σ -bottleneck measures.

A comprehensive study of these problems was initiated by Alt, Mehlhorn, Wagoner, and Welzl [AMWW88]. In this seminal work, they proposed a suite of algorithms for PM and LCP under the exact and bottleneck measures. More recent work [dRL95, Box96, IR96, ATT97] led to improved bounds for PM and LCP under the exact measure. The problem of estimating the minimum Hausdorff distance between two point sets in two and three dimensions has been studied extensively [CGH⁺93, HKK92, Ruc92]. Computing the minimum Hausdorff distance under translations where the underlying metric is L_∞ , rather than L_2 has also been studied [CDEK95]

Unfortunately, many of the above algorithms are impractical. As noted in the survey by Alt and Guibas [AG96] (also in the paper by Goodrich et al [GMO94]), these algorithms are likely to be “difficult to implement and numerically unstable due to the necessary computation of intersections of complex algebraic surfaces”. Worse still, they have unacceptably high running times: for example, even in \mathbb{R}^2 , LCP under the bottleneck measure requires $\tilde{O}(n^7)$ time² [EI96], although under additional restrictions on noise regions the running times can be improved to $\tilde{O}(n^4)$ [AKM⁺92]. Under the Hausdorff measure, the corresponding bound is $\tilde{O}(k^3 n^2)$ [CGH⁺93]; in \mathbb{R}^3 , the bounds are significantly worse.

Significant improvements came from considering the natural relaxation: algorithms that approximate the minimum value of $d(T(P), Q)$. Heffernan and Schirra [HS94] define an approximation framework in the following manner:

Problem 1.3. (β -APPROXIMATE PATTERN MATCHING)

Given $\epsilon, \beta > 0$ and point sets P, Q in the plane.

- If $\epsilon^* \leq \epsilon$, then return a transformation T such that $d(T(P), Q) \leq (1 + \beta)\epsilon$;
- If $\epsilon^* > (1 + \beta)\epsilon$, then return NONE.
- Otherwise, when $\epsilon^* \in (\epsilon, (1 + \beta)\epsilon]$, return any transformation T .

The β -approximate largest common point-set problem can be defined similarly.

²We use $\tilde{O}(f(n))$ to denote $O(f(n) \log^{O(1)} n)$ and $\delta(f(n))$ to denote $O(f(n) / \log^{\omega(1)} n)$.

Heffernan and Schirra [HS94] present an $\tilde{O}(n^{2.5} \text{poly}(\epsilon, 1/\beta))$ -time algorithm for point set congruence (PM where $k = n$) under the bottleneck measure. Efrat [Efr95] presents an $O(n(\log n + 1/\beta))$ -time algorithm for point set congruence under the Hausdorff measure in the plane under translations. Goodrich et al ([GMO94]) obtain simple algorithms that yield constant-factor approximations for PM under the Hausdorff measure in two and three dimensions under different transformation groups.

Recently and independently of our work, Cardoze and Schulman [SC98] presented a set of randomized algorithms for approximate PM and LCP under the Hausdorff measure in d dimensions under rigid transformations. For PM, the running time (omitting factors in ϵ, β) is $O(n^d \log n + \log^{O(1)} \Delta)$ for PM in d dimensions. For LCP the running times are multiplied by a factor of $(r \log n)$, where r is the number of mismatches; it is assumed that r is bounded by a constant fraction of the input size.

2 Our Results

In Tables 1 and 2, we present our approximation algorithms for the problems described above. For clarity, we omit terms that are polynomial in $1/\epsilon$ and $1/\beta$. Our algorithms are enumerative, in that they can also *enumerate* all transformations (and sets P') satisfying the approximation criteria. It should be noted that all the algorithms are deterministic.

All of our results are presented in terms of Δ , the ratio of the maximum distance to the minimum distance between points of Q ; (which is equivalent to the diameter, since we can assume that points are minimally separated). In most applications, Δ tends to be small ([EVW94]); a specific example is the case of small drug molecules, where atom locations correspond to points in three dimensions. In Figure 1, we plot the diameter as a function of n for over 127,000 small drug molecules taken from the National Cancer Institute database [Ins]. In the log-log plot, the diameters of the molecules are bounded by lines of slope less than one, implying that we can model the data as point sets of sub-linear diameter. One may also observe that for sets of points chosen uniformly at random from the unit square, with high probability $\Delta = O(n)$.

The presence of terms involving Δ renders our algorithms formally incomparable to previous work. However, for a wide range of values of Δ , we improve the existing bounds. For example, the algorithm by Goodrich et al yields a 4-approximation for PM in time $\tilde{O}(kn^2)$, which we improve for all instances where $\Delta = \delta(kn)$.

A common theme runs through all of the above re-

2D	d_H	d_M
LCP	$O(\Delta kn)$	$O(\Delta k^{2.5} n)$
PM	$O(\min(k(n^4 \Delta)^{1/3}, n(\Delta + n)))$	$O(k^{3/2}(n^4 \Delta)^{1/3})$

Table 1:

3D	d_H	d_M
LCP	$O(\Delta^3 kn)$	$O(\Delta^3 k^{2.5} n)$
PM	$O(\min(k \max(n^{2.25} \sqrt{\Delta}, n^{2.5}), n\Delta(\Delta^2 + n), n^2(n + \Delta)))$	$O(k^{1.5} \max(n^{2.25} \sqrt{\Delta}, n^{2.5}))$

Table 2:

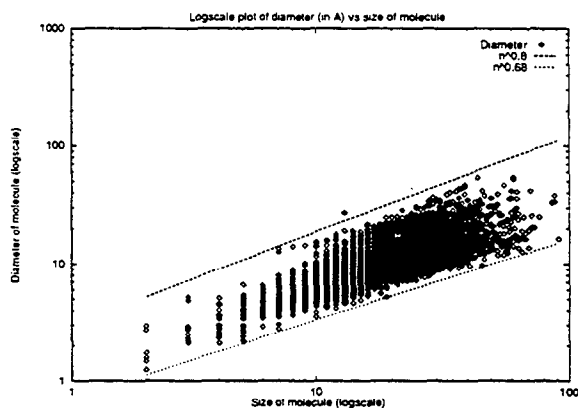


Figure 1: Diameter of NCI molecules versus number of atoms (log-scale).

sults. In each case, we first prove an upper bound on the multiplicity of a distance interval (in two dimensions) or a noisy triangle (in three dimensions). These results generalize and extend bounds first proved by Erdős, Makai, Pach and Spencer [EMPS91] and may have other applications. With these bounds, and a simple space partitioning technique, we can obtain alignment-based algorithms for PM and LCP. Secondly, we use a novel discretization technique that enables us to reduce the geometric matching problem to a combinatorial pattern matching problem. This approach, coupled with the combinatorial bounds, yields alternate algorithms for the same problems. The results above reflect this tradeoff.

The combinatorial bounds are presented in Section 3, with the resulting algorithms in Section 4. In Section 5, we present the discretization technique and the reduction to combinatorial subset matching. Section 6 discusses other algorithms that allow us tradeoffs, depending on the values of Δ , k , and n . In Section 6 we also extend our LCP results to “sum”-measure problems, including a near-linear time approximation algorithm for the *weighted k -mismatches* problem posed by Muthukrishnan [Mut95]; no sub-quadratic algorithm for

the latter problem was previously known.

3 Combinatorial Bounds

The study of combinatorial distance bounds was initiated by Paul Erdős [Erd46] in 1946, when he posed the question “Given a set of n points in the plane, how many pairs of points can be a unit distance apart?”. Erdős found an upper bound of $O(n^{3/2})$ and a lower bound of $\Omega(n^{1+c/\log \log n})$. Szemerédi and Trotter [ST83] reduced the upper bound to $O(n^{4/3})$. Later results [CEG⁺90, Szé97] reduced the constants and simplified the proof considerably. The lower bound remains unimproved and is conjectured to be tight.

A *noisy* version of the above problem was posed and solved by Erdős et al. in 1991:

THEOREM 3.1. ([EMPS91]) *Given $t > 0$, and a set of n points in the plane with minimum distance at least 1, the number of pairs (i, j) such that $d(p_i, p_j)$ lies in the range $[t, t + 1]$ is at most $\lfloor n^2/4 \rfloor$, provided n is sufficiently large.*

They also show that this bound is tight for sufficiently large t . However, this requires $t = \Omega(n^2)$.

Related problems were considered by Valtr [Val96] who investigated bounds on the number of approximate incidences between families of well-separated lines and points; assuming that the point sets are dense, i.e., their diameter is $O(\sqrt{n})$, he showed that several key results developed for the exact incidence model have counterparts in the approximate setting.

In the sequel we use $B(p, r)$ to denote a 2 or 3-dimensional ball of radius r centered at p . We also use $R(p, r_1, r_2)$ to denote a 2 or 3-dimensional annulus $B(p, r_2) - B(p, r_1)$. For a point p , let $N_\epsilon^t(p) = P \cap R(p, t, t + \epsilon)$. We denote a set of points as δ -separated if the minimum inter-point distance in this set is δ .

3.1 Noisy Distances in Two Dimensions We extend and generalize the bound given by [EMPS91], obtaining the following result:

THEOREM 3.2. *Given $\epsilon > 1, t > 1$, and a set of n 1-separated points in the plane, the number of pairs (i, j) such that $d(p_i, p_j)$ lies in the range $[t, t + \epsilon]$ is $O(\epsilon(n^4 t)^{1/3})$.*

The following facts are well-known. However, in the absence of a reference, we supply proofs in appendix A.

FACT 3.1. *For any n , there exists a set of n 1-separated points in the plane such that the number of distance pairs in the interval $[t, t + 1]$ is $\Omega(n\sqrt{t})$.*

FACT 3.2. *Given $\epsilon, l > 1$ and two points p and q in the plane such that $d(p, q) = d$ and $1 \leq d < 2l$, the number of 1-separated points whose distance from p and q lies in the interval $[l, l + \epsilon]$ is $O(\epsilon^2 l/d)$ when $d \leq l$, and $O(\epsilon^2 \sqrt{l/(2l-d)})$ when $l < d \leq 2l$.*

Our main lemma bounds the number of distance intervals induced by a single point.

LEMMA 3.1. *Let P be a 1-separated set of points in the plane. Given $\epsilon > 1$ and t , there exists a point $p \in P$ such that $N_\epsilon^t(p) = O(\epsilon(nt)^{1/3} \log n)$.*

Proof. Let f be a parameter such that for all points $p \in P$, $N_\epsilon^t(p) \geq f$. We show that $f = O(\epsilon(nt)^{1/3} \log n)$. Take any point $p \in P$. As $|N_\epsilon^t(p)| \geq f$, there must be a segment S_p of $R(p, t, t + \epsilon)$ subtending an angle $\pi/3$ such that $|S_p \cap P| = u \geq f/6$. Let q_1, q_2, \dots, q_u be the members of S_p , ordered in clockwise fashion around p . Notice that the distance between any pair q_i and q_j is at most t . Partition this sequence into subsequences S_1, S_2, \dots, S_k of size l . Let q'_1, q'_2, \dots, q'_k be the subsequence formed by taking the first member of each S_i , for $1 \leq i \leq k$, and let $N_i = N_\epsilon^t(q'_i)$. Note that

$$(3.1) \quad lk \geq f$$

By Fact 3.2, two points at distance $l \leq t$ can have at most $O(\epsilon^2 t/l)$ common neighbors in the distance range $[t, t + \epsilon]$. We can approximate arc lengths of a circle by a straight line with only a constant factor error, which implies that for some c and for $1 \leq j < i \leq k$, $|N_j \cap N_i| \leq c\epsilon^3 t/(l(i-j))$. Now,

$$\sum_{j=1}^{i-1} |N_j \cap N_i| \leq \sum_{j=1}^{i-1} c \frac{\epsilon^3 t}{(i-j)l} = c \frac{\epsilon^3 t}{l} H_i \leq c \frac{\epsilon^3 t \log n}{l}$$

where H_i is the i^{th} harmonic number. Note that $i \leq n$. The total contribution of $\cup_{j < i} N_j$ to N_i is bounded by $\frac{\epsilon^3 t \log n}{l}$. If we now choose $l = 2c\epsilon^3 t \log n/f$, for some constant c , the contribution is at most $f/2$, ensuring

that for each i , $|N_i - \cup_{j=1}^{i-1} N_j| \geq \alpha f$, for some constant α . Therefore,

$$|\cup_{i=1}^k N_i| = \sum_{i=1}^k |N_i - \cup_{j=1}^{i-1} N_j| \geq \alpha f k.$$

But clearly $|\cup_{i=1}^k N_i| \leq n$. This yields

$$(3.2) \quad kf \leq n/\alpha.$$

Using Equations (3.1) and (3.2), and substituting the value of l , we obtain

$$\frac{\epsilon^3 n t \log n}{f} \geq \alpha c f^2,$$

which yields the desired bound.

Theorem 3.2 now follows by induction, as at each step, we can use Lemma 3.1 to upper bound the number of distance pairs added. The details are omitted in this abstract.

3.2 Noisy Triangles in Three Dimensions In this section, we prove analogous results for *noisy* triangles in three dimensions. For any measure $d \in \{d_H, d_M, d_E\}$ we define approximate congruence \cong_ϵ^d as follows. Let $\Delta = (p_1, p_2, p_3)$ and $\Delta' = (p'_1, p'_2, p'_3)$ be two triangles. Then $\Delta \cong_\epsilon^d \Delta'$ if and only if $d(\Delta, \Delta') \leq \epsilon$. Hereafter, we assume that for all p_i and p_j , where $1 \leq i < j \leq 3$, the noise regions $B(p_i, \epsilon)$ and $B(p_j, \epsilon)$ are disjoint. It is then easy to see that $\cong_\epsilon^{d_H}$ and $\cong_\epsilon^{d_M}$ are equivalent, so we omit the superscript d from \cong_ϵ^d . For a point set P , $|P| = n$, we define $H_P^\epsilon(\Delta)$, the *multiplicity of Δ* , as the following;

$$H_P^\epsilon(\Delta) = |\{\Delta' = (p_1, p_2, p_3) \mid \Delta' \cong_\epsilon \Delta, p_i \in P, i = 1, 2, 3\}|$$

Let $H_n^\epsilon(\Delta) = \max_{|P|=n} H_P^\epsilon(\Delta)$. In the sequel, we drop n from H_n^ϵ if its value is clear from the context.

Our main theorem in this section is:

THEOREM 3.3. *For any triangle Δ with largest side length l , $H^\epsilon(\Delta) = O(\min(\epsilon^{2.5} n^{2.25} \sqrt{l}, n^3))$.*

Proof: (WHAT DO WE SAY HERE) ◁

A corresponding lower bound is:

FACT 3.3. *For any $l > 1$ such that $l = O(n^2)$, there exists a point set P and a triangle Δ with the largest side length l such that $H_P^1(\Delta) = \Omega(n^2 \sqrt{l})$.*

4 Alignment-based Schemes

We present alignment-based schemes for PM and LCP that utilize the combinatorial bounds of the previous section.

Let p_1, p_2, q_1, q_2 be any points. Then for any $\epsilon > 0$, we define

$$T_\epsilon(p_1, p_2, q_1, q_2) = \{T \mid d_H(\{T(p_1), T(p_2)\}, \{q_1, q_2\}) \leq \epsilon\}$$

$$S_\epsilon(p_1, p_2, q_1, q_2) = \{(T(p_1), T(p_2)) \mid T \in T_\epsilon(p_1, p_2, q_1, q_2)\}$$

Both T_ϵ and S_ϵ are infinite subsets of the transformation space. The following proposition (in the spirit of [HS94]) shows that this space can be approximated by a small set of transformations.

PROPOSITION 4.1. *For any $S = S_\epsilon(p_1, p_2, q_1, q_2)$ and $\gamma > 0$, there exists $S^* \subset S$ of cardinality $O(1/\gamma^3)$ such that for any $(s_1, s_2) \in S$, there exists $(s_1^*, s_2^*) \in S^*$ with $d_H(\{s_1, s_2\}, \{s_1^*, s_2^*\}) \leq \gamma\epsilon$. Such a set S^* is denoted by $S_{\gamma, \epsilon}^*(p_1, p_2, q_1, q_2)$.*

Let $d = d_H$, and let $\gamma = \beta/4$. The algorithm is as follows:

Algorithm 2D Alignment:

1. Compute a diameter pair p_1, p_2 of P .
2. Find the set A of all pairs (q_1, q_2) of points in Q such that $\|p_1 - p_2\| - 2\epsilon \leq \|q_1 - q_2\| \leq \|p_1 - p_2\| + 2\epsilon$.
3. Construct $\mathcal{T} = \cup_{(q_1, q_2) \in A} \{T_0(p_1, p_2, s_1, s_2) \mid (s_1, s_2) \in S_{\gamma, \epsilon}^*(p_1, p_2, q_1, q_2)\}$.
4. Search for $T^* \in \mathcal{T}$ such that $d_H(T^*(P), Q) \leq (1 + 3\gamma)\epsilon$.

LEMMA 4.1. *Algorithm 2D Alignment is a β -approximation algorithm for PM.*

Proof. Assume that there exists a transformation T such that $d_H(T(P), Q) \leq \epsilon$. Let p_1, p_2 be the diameter points found by the algorithm. Let $s_1 = T(p_1)$ and $s_2 = T(p_2)$ and let $q_1, q_2 \in Q$ be points such that $d(s_i, q_i) \leq \epsilon$ for $i = 1, 2$. By the definition of $S^* = S_{\gamma, \epsilon}^*(p_1, p_2, q_1, q_2)$ we know that there exists $(s_1^*, s_2^*) \in S^*$ such that $d_H(\{s_1^*, s_2^*\}, \{q_1, q_2\}) \leq \gamma\epsilon$. Let T^* be the transformation which maps (p_1, p_2) to (q_1, q_2) . Consider any point $p \in P$. We need to estimate $d(T(p), T^*(p))$. We represent T^* as $R \circ I \circ T$, where I is a translation which moves s_1 to s_1^* , and R is a rotation centered at s_1^* which moves $I(s_2)$ to s_2^* . Then

$$\begin{aligned} d(T(p), T^*(p)) &\leq d(T(p), (I \circ T)(p)) + \\ &\quad d((I \circ T)(p), (R \circ I \circ T)(p)) \\ &\leq d(s_1, I(s_1)) + d(I(s_2), s_2^*) \\ &\leq d(s_1, s_1^*) + (d(I(s_2), s_2) + d(s_2, s_2^*)) \\ &\leq 3\gamma\epsilon \leq \beta\epsilon. \end{aligned}$$

Therefore $d(T^*(p), Q) \leq d(T(p), Q) + d(T(p), T^*(p)) \leq \epsilon + \beta\epsilon = (1 + \beta)\epsilon$, thus T^* gives the desired approximate mapping. It is straightforward to check that our algorithm finds such T^* .

LEMMA 4.2. *Algorithm 2D Alignment can be implemented to run in time*

$$O\left(k \frac{\min(\epsilon n^{4/3} \Delta^{1/3} \log n, n^2)}{\beta^3} \cdot \min\left(\left(\frac{\epsilon}{\beta}\right)^2, \log n\right)\right)$$

for any $\epsilon \geq 1$ and $\beta \leq 1$, where Δ is the diameter of P . For $\epsilon < 1$ the running time is as for $\epsilon = 1$; similarly for $\beta > 1$.

Proof. Consider the complexity of each step of the algorithm:

Step 1: Can be performed in time $O(n \log n)$ using the standard algorithm [PS85]

Step 2: Let $r = d(p_1, p_2)$. We need to find $A = \cup_{q_1 \in Q} \{(q_1, q_2) \mid q_2 \in Q \cap R(q_1, r - 2\epsilon, 4\epsilon)\}$. If $\Delta \leq n^2$ then we employ the following bucketing scheme. Let G denote a planar grid consisting of rectangular cells of width 4ϵ and height \sqrt{r} . For each $i = 0, \dots, \sqrt{r}$ we define G_i to be the grid G rotated by an angle $\frac{i\pi}{2\sqrt{r}}$. Notice, that for any point q the ring $R(q, r - 2\epsilon, 4\epsilon)$ can be covered by $O(\sqrt{r})$ grid cells from G_0, \dots, G_r (this can be proved similarly to Fact 3.2).

During the preprocessing for all $G_0, \dots, G_{\sqrt{r}}$ and all $q \in Q$ we store q at the grid cell of all G_i containing q (using hashing). Then, in order to compute $Q \cap R(q, r - 2\epsilon, 4\epsilon)$ we retrieve the contents of the $O(\sqrt{r})$ buckets covering $R(q, r - 2\epsilon, 4\epsilon)$. The total complexity of this step can be bounded by the total number of the points retrieved (which is $O(\min(\epsilon n^{4/3} \Delta^{1/3} \log n, n^2))$ by Theorem 3.2) plus the total number of buckets accessed (which is $O(n\sqrt{\Delta})$) plus the cost of the preprocessing (which is again $O(n\sqrt{\Delta})$).

If $\Delta > n^2$, then we can compute A by computing all n^2 pairwise distances in time $O(n^2)$. Therefore, Step 2 can be implemented to run in $O(\min(n^2, \epsilon n^{4/3} \Delta^{1/3} \log n + n\sqrt{\Delta}))$ time, which is $O(\min(n^2, \epsilon n^{4/3} \Delta^{1/3} \log n))$.

Steps 3/4: In these steps, we perform at most $O(\min(\epsilon n^{4/3} \Delta^{1/3} \log n, n^2)/\beta^3)$ computations of $d_H(T(p), Q)$. Each computation requires k comparisons of $d_H(T(p), Q)$ to $(1 + 3\gamma)\epsilon$. It is easy to perform this comparison in $O(\log n)$ using a Voronoi diagram. One can however achieve constant time per query as follows. Impose a uniform grid of side $\gamma\epsilon$. For each $q \in Q$ we store it in all grid cells intersecting $B(q, (1 + 3\gamma)\epsilon)$. This takes $O(n/\gamma^2)$ time which is subsumed by the complexity of searching \mathcal{T} . In order to verify if $d_H(p, Q) \leq$

$(1 + \beta)\epsilon$, it is sufficient if the grid containing p contains any q .

We can prove similar bounds for the case of PM/LCP in three dimensions, using our results from Section 3.2. We present the result here, deferring the details to an extended version of the paper.

THEOREM 4.1. *β -approximate PM in three dimensions can be solved, for all $\epsilon > 1, \beta \leq 1$, in time $O(k \max(n^{2.25}\sqrt{\Delta}, n^{2.5}) \cdot (\frac{\epsilon}{\beta})^3 / \beta^6)$.*

5 Concentric Pattern Matching

The subset matching problem is an important problem in combinatorial pattern matching and is defined as follows:

Subset Matching: Let Σ be a finite alphabet. Given a text string $t[0, \dots, n - 1]$ and a pattern string $p[0, \dots, m - 1]$ such that for all i and j , $t[i], p[j] \in \Sigma$, return a binary array $o[0, \dots, n - 1]$ such that $o[i] = 1$ if and only if $p[j] \subset t[i+j \bmod n]$ for all $j \in \{0, \dots, m-1\}$.

Recent work of Cole and Hariharan [CH97] showed that this problem can be solved in (randomized) $O(n \log^3 m)$ time. Cole, Hariharan, and Indyk [CHI99] have shown that the above algorithm can be derandomized to run in $O(n \log^3 m)$ time.

We present Algorithm 2D Concentric Pattern Matching which solves the approximate noisy pattern matching problem in almost-quadratic time by reducing it to multiple instances of the subset matching problem.

The basic idea of the algorithm is as follows. From earlier remarks we can assume that the rigid transformation to be computed can be expressed as a composition of a translation and a rotation.

1. Choose an arbitrary point p in P and translate it to all points in $B(q, \epsilon)$, for all $q \in Q$. Using the same technique as in Section 4, we can replace this infinite space of translations by a discrete set.
2. For each such alignment, subdivide the plane into concentric rings around p , of increasing radius. Each ring will yield one instance of a subset matching problem.
3. Further subdivide each ring concentrically and radially into cells of fixed size. Since the point set is 1-separated, this implies that each cell contains at most one point, for appropriate choice of constants.
4. For any ring, all the cells at a fixed distance from the center have the same identifier. Construct a text string made up of identifiers of cells that contain points of Q , where each position in the

string corresponds to the set of all such cells that lie along the same radial line. Similarly construct a pattern string using points of P .

Each *match* of text and pattern corresponds to a rotational shift of the corresponding ring. It is now easy to see that there exists a rotation matching P approximately into Q if there exists a single rotational shift that corresponds to a pattern match for *each* ring.

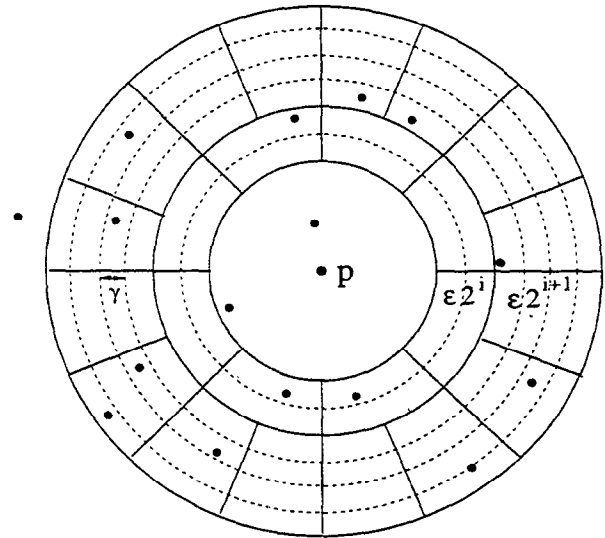


Figure 2: The grid decomposition

We now present the algorithm in more detail.

(1) Choose an arbitrary point (say p) from P . For each $q \in Q$ align p to a set of $O(1/\beta^2)$ points spaced uniformly inside the disk of radius ϵ centered at q ; this is done similarly as in Algorithm 2d Alignment. For each such translation T , move the points P according to T ; for simplicity we still refer to $T(P)$ as P .

(2) In the next step, split the plane into $l = O(\log \Delta)$ concentric rings R_1, \dots, R_l centered at p (see Figure 2); note that R_1 is a full disk. The inner radius of the i -th ring (for $i \geq 2$) is equal to $\epsilon 2^{i-1}$, the outer radius (for $i \geq 1$) is $r_i = \epsilon 2^i$. For each ring R_i , let I_i be the (infinite) set of rotations which brings $P \cap R_i$ within Hausdorff distance ϵ of $Q \cap (R_{i-1} \cup R_i \cup R_{i+1})$. Clearly, P is within Hausdorff distance ϵ of Q under rotation iff $I_1 \cap \dots \cap I_l \neq \emptyset$; this follows from the fact that a disk of radius ϵ intersects at most two rings. Therefore it is sufficient to compute the sets I_1, \dots, I_l and find their intersection.

Each I_i can be represented as a finite union of intervals. In order to solve the approximate matching problem, we approximate the I_i as follows: the endpoints of the intervals in each I_i restricted to multiples of

$\alpha_i = \gamma/2^i$, for $\gamma > 0$ specified later. Denote the approximate intervals by \tilde{I}_i . Include an interval $[j\alpha_i, (j+1)\alpha_i]$ in \tilde{I}_i if $I_i \cap [j\alpha_i, (j+1)\alpha_i] \neq \emptyset$; we call such an interval *important*. Notice that rotation of $P \cap R_i$ by angle α_i changes the position of each point in $P \cap R_i$ by at most $2^i\alpha_i = \gamma$. Therefore, we can ensure that all important intervals are contained in \tilde{I}_i by including all intervals $[j\alpha_i, (j+1)\alpha_i]$ such that rotating $P_i = P \cap R_i$ by both angles $j\alpha_i$ and $(j+1)\alpha_i$ results in a set which is within Hausdorff distance $\epsilon + \gamma$ from $Q_i = Q \cap (R_{i-1} \cup R_i \cup R_{i+1})$. This admits some false matches, but as we explain below, the error they induce is bounded.

(3) To find all such intervals, check for each angle $j\alpha_i$ if $P'_i \subset S_i$, where the set P'_i is obtained by rotating P_i by $j\alpha_i$ and $S_i = \cup_{q' \in Q} B(q', \epsilon + \gamma)$. Partition each R_i into $2\pi/\alpha_i$ sectors, the sectors being partitioned further by $2^i/\gamma$ uniformly placed concentric circles (Figure 2). We denote the set of grid cells obtained from the ring R_i by G_i ; the union of all G_i 's (i.e., the whole partition) is denoted by G . For any point x , let $G(x)$ be the cell of G to which x belongs (ties are broken arbitrarily); the function G can be extended to sets of points in a natural way. Each grid cell has diameter $c\gamma$, for $c > 0$ with value near $\sqrt{2}$.

(4) Now, for each angle $j\alpha_i$, check if $G(P'_i) \subset G(S_i)$ for the set P'_i obtained by rotating P_i by an angle $j\alpha_i$. This condition clearly implies that $P'_i \subset S_i$; although it also introduces false matches, the error can be bounded as claimed. Notice however, that rotation of P_i by α_i results in “shifting” the set $G(P_i)$ by one position in G_i . Therefore, it is sufficient to check (for all shifts) if the shifted set $G(P_i)$ belongs to $G(S_i)$. This can be done using the subset matching algorithm as follows. Define a *signature* of a grid cell to be its distance to the origin point p ; note that all grids from one sector of a ring R_i have different signatures, while the signatures of cells from different sectors can be equal. Define the pattern p to be a sequence of sets $p[0], p[1], \dots$ such that each set $p[j]$ contains signatures of grid cells from $G(P_i)$ belonging to the j th sector of R_i (the first sector is chosen arbitrarily). The text t is constructed analogously to $G(S_i)$. It is easy to check that the subset matching algorithm finds the desired shift.

By the above discussion it follows that whenever there is an angle δ such that rotating P by δ results in a set P' within distance ϵ to Q , the intersection $\tilde{I}_1 \dots \tilde{I}_l$ will be non-empty. On the other hand, we know that for any $q' \in Q$ the distance from q' to any point in $G(B(q', \epsilon + \gamma))$ is at most $\epsilon + (1+c)\gamma$. Therefore, a false match rotation occurs only if it brings P with distance $\epsilon + (1+c)\gamma$ to Q ; for sufficient γ , this distance can be made less than $\frac{1}{2}\epsilon$.

This scheme constructs $l = \log \Delta$ instances of the

subset matching problem. Each instance is drawn from an alphabet of size $O(\Delta/\epsilon\beta)$. The pattern (and text) lengths are $O(2^i/\gamma) = O(\Delta/\epsilon\beta)$. From this, and the above algorithm, we obtain the following theorem:

THEOREM 5.1. *2D Concentric Pattern Matching is a β -approximation algorithm for the PM problem, for suitable $\gamma = \Theta(\beta)$. It can be implemented in $O(n(\frac{\Delta}{\epsilon\beta} + \frac{n}{\beta^2}) \log^3 n \cdot \log \Delta)$ time, for any $\beta \leq 1$. For $\beta > 1$ the running time is as for $\beta = 1$.*

6 Other Results

In this section we briefly describe the other results presented in Tables 1 and 2. Consider first the two-dimensional case. The bounds for pattern matching under the Hausdorff measure follow from the previous section. The algorithm for pattern matching under the bottleneck measure is obtained via alignment; the $k^{1.5}$ factor is the time needed for match verification of a transformation T obtained from alignment [E196].

In order to solve LCP we employ a voting scheme similar to the Hough transform [Bal81]. First, we discretize the rotation space, obtaining $O(\Delta)$ different angles. Then, for each angle we enumerate all pairs $p \in P$ and $q \in Q$. Each such pair “votes” for a translation moving p to q . At the end we count the number of votes for each (rotation, translation) pair and output all pairs with more than K votes. For the bottleneck measure, we again employ the bipartite matching algorithm.

In three dimensions, the bounds are obtained in a similar way. The only difference is that we have additional freedom in designing algorithms for PM under the Hausdorff measure. Let $p_1, p_2 \in P$. We can either align p_1 to all $q \in Q$, enumerate one of the angular coordinates, and apply two-dimensional subset matching; or align p_1, p_2 to all pairs $q_1, q_2 \in Q$ and apply 1-dimensional subset matching; or align all matching triangles. The corresponding bounds follow.

Sum Measures. We briefly describe how to extend our PM and LCP results to the Σ -Hausdorff measure; again, the Σ -bottleneck measure can be computed using the bipartite matching algorithm. Consider any transformation T . Suppose that for any ϵ we know the number $f(\epsilon)$ of points from $T(P)$ which are within distance ϵ of Q . Assume for simplicity that all distances from $p \in P$ to Q are distinct. Define $f^{-1}(i)$ to be the “inverse” of f , i.e., the function which when given a number $i \in \{0, \dots, n\}$ returns the smallest ϵ such that $f(\epsilon) = i$. Then it is easy to see that the Σ -Hausdorff distance between $T(P)$ and Q is equal to $\sum_{i=1}^k f^{-1}(i)$. Similarly, we can compute an approximation of that distance if we know an approximation of f , i.e., a function \tilde{f} such that $\tilde{f}(\epsilon) \in \{f(\epsilon), \dots, f((1+\beta)\epsilon)\}$, for some

$\beta > 0$. Such an approximation can be obtained by solving a logarithmic number of instances of approximate LCP between $T(P)$ and Q under Hausdorff measure for $\epsilon = 1, (1 + \gamma), (1 + \gamma)^2, \dots, \Delta$ and $\gamma = \Theta(\epsilon)$. Thus, we can obtain an approximation of the Σ -Hausdorff for a fixed transformation T . However, our Hough-transform-based algorithm computes the LCP for the whole (properly discretized) space of transformations. Therefore we can choose the one which minimizes the distance value, obtaining an approximate solution for PM under Σ -Hausdorff measure. An algorithm for LCP under the same measure can be similarly obtained.

The technique of using the values of “max-measures” to approximate the values of sum measures turns out to be useful in a different context. Consider the following *weighted k -mismatches problem* posed by Muthukrishnan [Mut95]: given a text $t = t[1 \dots n]$ and a pattern $p = p[1 \dots m]$ over an alphabet Σ , a real parameter Δ , and a weight function $f : \Sigma \times \Sigma \rightarrow \mathbb{R}$, determine all positions i in the text where $\sum_{j=1}^m f(t[i + j - 1], p[j]) \leq \Delta$. This problem has many applications in computational biology [Mut95]. Muthukrishnan asked if the problem can be solved in $o(nm)$ time even for the case where Σ is the set of natural numbers and $f(\sigma, \tau) = |\sigma - \tau|$. Using our techniques, we can solve the *approximate* version of this problem (formulated like the approximate point matching problems) in $\tilde{O}(n)$ time. Observe that when in the definition of the problem we replace \sum by “max”, the resulting problem can be solved in $\tilde{O}(n)$ time using the subset matching algorithm as described earlier. In fact, within the same time bounds we can solve the counting version of the “max” problem, where the goal is to determine for each position i the approximate number of j 's such that $|t[i + j - 1] - p[j]| \leq \Delta$; this can be done by resorting to the counting version of the subset matching algorithm due to Indyk [Ind97]. We can thus obtain an approximate solution for the weighted k -mismatches problem in near-linear time.

References

- [AG96] H. Alt and L. Guibas. Discrete geometric shapes: Matching, interpolation, and approximation. a survey. Technical Report B96-11, Freie Universität Berlin, December 1996.
- [AKM⁺92] E.M. Arkin, K. Kedem, J.S.B. Mitchell, J. Srinivasan, and M. Werman. Matching points into pairwise-disjoint noise regions: Combinatorial bounds and algorithms. *ORSA Journal on Computing*, 4(4):375–386, 1992.
- [AMWW88] H. Alt, K. Melhorn, H. Wagener, and E. Welzl. Congruence, similarity, and symmetries of geometric objects. *Discrete Computational Geometry*, 3:237–256, 1988.
- [ATT97] T. Akutsu, H. Tamaki, and T. Tokuyama. Distribution of distances and triangles in a point set and algorithms for computing the largest common point set. In *Proceedings of the Thirteenth Annual ACM Symposium on Computational Geometry*, 1997.
- [Bal81] D.H. Ballard. Generalizing the Hough Transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.
- [Box96] L. Boxer. Point set pattern matching in 3-d. *Pattern Recognition Letters*, 17(12):1293–1297, 1996.
- [CDEK95] L. P. Chew, D. Dor, A. Efrat, and K. Kedem. Geometric pattern matching in d -dimensional space. In *Proc. 2nd Annu. European Sympos. Algorithms*, volume 979 of *Lecture Notes Comput. Sci.*, pages 264–279. Springer-Verlag, 1995.
- [CEG⁺90] K. Clarkson, H. Edelsbrunner, L. Guibas, M. Sharir, and E. Welzl. Combinatorial complexity bounds for arrangements of curves and spheres. *Discrete Computational Geometry*, 5:99–160, 1990.
- [CGH⁺93] L. Chew, M.T. Goodrich, D.P. Huttenlocher, K. Kedem, J.M. Kleinberg, and D. Kravets. Geometric pattern matching under euclidean motion. In *Proceedings of the Fifth Canadian Conference on Computational Geometry*, pages 151–156, 1993.
- [CH97] R. Cole and R. Hariharan. Tree pattern matching and subset matching in randomized $O(n \log^3 m)$ time. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, 1997.
- [CHI99] R. Cole, R. Hariharan, and P. Indyk. Tree pattern matching and subset matching in deterministic $O(n \log^3 m)$ Time. In *This proceedings*, 1999.
- [dRL95] P.J. de Rezende and D.T. Lee. Point set pattern matching in d -dimensions. *Algorithmica*, 13:387–404, 1995.
- [Efr95] A. Efrat. Finding approximate matching of points and segments under translation. Unpublished manuscript, 1995.
- [EI96] A. Efrat and A. Itai. Improvements on bottleneck matching and related problems using geometry. In *Proceedings of the Twelfth Annual ACM Symposium on Computational Geometry*, 1996.
- [EMPS91] P. Erdős, E. Makai, J. Pach, and J. Spencer. Gaps in difference sets, and the graph of nearly equal distances. *Applied Geometry and Discrete Mathematics: The Victor Klee Festschrift (P. Gritzmann and B. Sturmfels, eds.)*, 4:265–273, 1991.
- [Erd46] P. Erdős. On sets of distances of n points. *American Mathematical Monthly*, 53:248–250, 1946.
- [EVW94] H. Edelsbrunner, P. Valtr, and E. Welzl. Cutting dense point sets in half. In *Proceedings of the Tenth Annual ACM Symposium on Computational Geometry*, pages 203–210, 1994.
- [FHK⁺96] P. Finn, D. Halperin, L. E. Kavradi, J. C. Latombe, R. Motwani, C. Shelton, and S. Venkatasubramanian. Geometric manipulation of flexible ligands. *LNCS Series - 1996 ACM Workshop on Applied Computational Geometry*, 1148:67–78, 1996.
- [FKL⁺97] P. Finn, L. E. Kavradi, J. C. Latombe, R. Mot-

- wani, C. Shelton, S. Venkatasubramanian, and A. Yao. Rapid: Randomized pharmacophore identification for drug design. In *Proceedings of the Thirteenth Annual ACM Symposium on Computational Geometry*, 1997.
- [GMO94] M.T. Goodrich, J.B. Mitchell, and M.W. Orletsky. Practical methods for approximate geometric pattern matching under rigid motions. In *Proceedings of the Tenth Annual ACM Symposium on Computational Geometry*, pages 103–113, 1994.
- [HKK92] D. P. Huttenlocher, K. Kedem, and J. M. Kleinberg. On dynamic Voronoi diagrams and the minimum Hausdorff distance for points sets under Euclidean motion in the plane. In *Proceedings of the Eighth Annual ACM Symposium on Computational Geometry*, pages 110–120, 1992.
- [HS94] P. J. Heffernan and S. Schirra. Approximate decision algorithms for point set congruence. *Computational Geometry: Theory and Applications*, 4(3):137–156, 1994.
- [Ind97] P. Indyk. Deterministic superimposed coding with applications to matching. In *Proceedings of the Thirty Eight IEEE Symposium on Foundations of Computer Science*, pages 127–136, 1997.
- [Ins] National Cancer Institute. 2D and 3D Structural data from the Developmental Therapeutics Program, DCTDC, NCI. <http://epwvst1.ncifcrf.gov:2345/dis3d/3ddatabase/pubstruc.html>.
- [IR96] S. Irani and P. Raghavan. Combinatorial and experimental results for randomized point matching algorithms. In *Proceedings of the Twelfth Annual ACM Symposium on Computational Geometry*, pages 68–77, 1996.
- [MNL98] D. Mount, N. Netanyahu, and J. LeMoigne. Improved algorithms for robust point pattern matching and applications to image registration. In *Fourteenth ACM Symposium on Computational Geometry*, June 1998.
- [Mut95] S. Muthukrishnan. New results and open problems related to non-standard stringology. In *Proceedings of the Symposium on Combinatorial Pattern Matching*, pages 298–317, 1995.
- [NFWN94] R. Norel, D. Fischer, H. J. Wolfson, and R. Nussinov. Molecular surface recognition by a computer vision-based technique. *Protein Engineering*, 7(1):39–46, 1994.
- [PS85] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York., 1985.
- [Ruc92] W.T. Rucklidge. Lower bounds for the complexity of the Hausdorff distance. In *Proceedings of the Fifth Canadian Conference on Computational Geometry*, pages 145–150, 1992.
- [SC98] L. Schulman and D. Cardoze. Pattern matching for spatial point sets. In *Thirty-ninth Annual Symposium on the Foundations of Computer Science*. IEEE, November 1998.
- [ST83] E. Szemerédi and W. T. Trotter. Extremal problems in discrete geometry. *Combinatorica*, 3:381–392, 1983.
- [Szé97] L. Székely. Crossing numbers and hard Erdős problems in discrete geometry. *Combinatorics, Probability and Computing (to appear)*, 1997.
- [Val96] P. Valtr. Lines, line-point incidences and crossing families in dense sets. *Combinatorica*, 16:269–294, 1996.

Appendix

A Proofs of Facts 3.1, 3.2

Proof. (Fact 3.1) Consider two rows of $n/2$ points that are t units apart. In each row, the points are spaced at unit intervals. If we consider the annulus of inner radius t and outer radius $t + 1$ swept out by any point in the first row, it will intersect a line segment on the second row of length \sqrt{n} , which implies that it intersects \sqrt{t} points. Hence, the bound follows.

Proof. (Fact 3.2) Let us scale all distances by ϵ , setting $l' = l/\epsilon$ and $d' = d/\epsilon$. The minimum inter-point separation is now $1/\epsilon$. Construct around each point concentric circles of radius l' and $l' + 1$. We need to estimate the diameter of the common area. To this end, we observe that the four circles intersect in exactly eight points (in two symmetric parts).

Consider the top four intersection points A, B, C , and D (see Figure 3). We wish to estimate the number of points lying in the region $ABCD$. It is easy to see that this can be bounded by $\epsilon^2 \max(d(A, C), (B, D))$. If we set the origin at p , and let q lie at $(d', 0)$, the coordinates of A, B, C, D are as follows:

$$\begin{aligned} A &: (d'/2 - (2l' + 1)/2d', \sqrt{l'^2 - (d'/2 - (2l' + 1)/2d')^2}), \\ B &: (d'/2, \sqrt{(l' + 1)^2 - (d'/2)^2}), \\ C &: (d'/2 + (2l' + 1)/2d', \sqrt{l'^2 - (d'/2 + (2l' + 1)/2d')^2}), \\ D &: (d'/2, \sqrt{l'^2 - (d'/2)^2}), \end{aligned}$$

which yields $\max(d(A, C), d(B, D)) = \max((2l' + 1)/d', \sqrt{(l' + 1)^2 - (d'/2)^2} - \sqrt{l'^2 - (d'/2)^2})$. It is easy to verify that for $d \leq l$, this gives a bound of $O(l/d)$; for $l < d \leq 2l$, the bound is $O(\sqrt{l/(2l - d)})$.

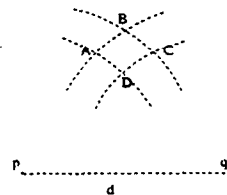


Figure 3: The four points of intersection of the two annuli