

# Approximate Congruence in Nearly Linear Time

PIOTR INDYK \*      SURESH VENKATASUBRAMANIAN †

## Abstract

The problem of geometric point set matching has been well-studied in the domain of computational geometry, and it has many applications in areas such as computer vision, computational chemistry, and pattern recognition. One of the commonly used metrics is the *bottleneck matching* distance, which is the minimum length of the longest edge in a one-to-one mapping between two point sets. Much effort has gone into developing efficient algorithms for minimising the bottleneck distance between two point sets under groups of transformations. However, the algorithms that have thus far been developed suffer from running times that are large polynomials in the size of the input, even for approximate formulations of the problem.

In this paper we define a point set similarity measure that includes both the bottleneck distance and the *Hausdorff distance* as special cases. This measure relaxes the condition that the mapping must be one-to-one, but guarantees that only a few points are mapped to any point. Using a novel application of Hall's Theorem to reduce the geometric matching problem to a combinatorial matching problem, we present near-linear time approximation schemes for minimising this distance between two point sets under isometries; we note here that the best known algorithms for congruence under the bottleneck measure run in time  $\tilde{O}(n^{2.5})$ .

We also obtain a combinatorial bound on the *metric entropy* of certain families of geometric objects. This result yields improved algorithms for approximate congruence, and may be of independent interest.

---

\*Department of Computer Science, Stanford University, CA 94305. Email: [indyk@cs.stanford.edu](mailto:indyk@cs.stanford.edu)

†Department of Computer Science, Stanford University, CA 94305. Email: [suresh@cs.stanford.edu](mailto:suresh@cs.stanford.edu)

# 1 Introduction

Geometric point set matching in two and three dimensions is a well-studied area of computational geometry, with application to fields such as computer vision [MNL98], pattern recognition [CGH<sup>+</sup>93, HKK92] and computational chemistry [FHK<sup>+</sup>96, FKL<sup>+</sup>97, NFWN94]. Given some choice of a space  $\mathcal{G}$  of transformations (e.g. all isometries) 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:

**Problem 1.1 (Congruence)** *Given point sets  $P, Q$  and  $\epsilon > 0$ , determine the transformation  $T \in \mathcal{G}$  that brings  $P$  close to  $Q$ , i.e. such that  $d(T(P), Q) \leq \epsilon$ .*

A natural measure that has been studied is the *bottleneck matching* distance, defined as the minimum value  $r$  such that there is a one-to-one mapping  $f$  from  $P$  to  $Q$  under which  $d(p, f(p)) \leq r$  for any  $p \in P$ ; we call  $r$  the *bottleneck value*. Alt, Mehlhorn, Wagener, and Welzl [AMWW88] initiated a comprehensive study of point set matching under the bottleneck distance, proposing a suite of polynomial time algorithms. The running times of their algorithms are large; for example, for two-dimensional point-sets and transformation space restricted to isometries (which we call a *noisy congruence problem*) the running time is  $\tilde{O}(n^8)^*$ , where  $n = |P| = |Q|$  (although they do not mention it explicitly, their algorithms work also for  $|P| = k < n = |Q|$ ). Even though the running time can be reduced to  $\tilde{O}(n^7)$  [EI96], it is still quite high. Moreover, 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”. In order to obtain faster and more practical algorithms several authors proposed algorithms for restricted cases (e.g. [AKM<sup>+</sup>92]) or resorted to *approximations*, i.e. algorithms which guarantee solutions with cost at most  $r(1 + \epsilon)$ , for some  $\epsilon > 0$ . This line of research was initiated by Heffernan and Schirra [HS94] who gave an  $\tilde{O}(n^{2.5})$ -time algorithm for approximate noisy congruence problem for the case when the noise regions are “small”. The latter assumption can be removed by using the techniques of Efrat and Itai [EI96] with no change in the asymptotic complexity; to our knowledge this is the best algorithm solving the approximate noisy congruence problem.

In the meantime, many algorithms have been proposed for other measures, most notably the *Hausdorff distance*, for which  $f$  is not restricted to be one-to-one. It was observed in [EI96] and by others that this lack of restriction is unsuitable in many situations (since it might happen that many feature points of the pattern are associated with only one point from the image); on the other hand, it seems to reduce the complexity of the problem. In particular, the exact problem can be solved in  $\tilde{O}(n^5)$  time [CGH<sup>+</sup>93]. Goodrich et. al. gave an approximate algorithms running in  $\tilde{O}(n^3)$ -time. Both algorithms in fact solve the pattern matching problem (i.e. allow  $|P| < |Q|$ ); if we assume  $|P| = |Q|$  then (as in [HS94]) the running time can be reduced to  $\tilde{O}(n^2)$ . Recently, Cardoze and Schulman [CS98] and Indyk et al [IMV99] proposed a new paradigm for geometric point set matching based on algebraic convolutions. This reduced the complexity of pattern matching to  $\tilde{O}(n^2)$ ; using their techniques congruence can be solved within the (ultimate)  $\tilde{O}(n)$  bound. Unfortunately, the one-to-one restriction imposed by bottleneck matching distance seems not to fit well within the rigid framework of algebraic convolutions.

In this paper, we define a point set distance measure, the *generalized bottleneck distance*, that includes as special cases both the bottleneck distance and the Hausdorff distance. More specifically, we allow transformations for which the corresponding function  $f : P \rightarrow Q$  maps some small fraction

---

\* $\tilde{O}(f(n))$  denotes  $f(n) \log^{O(1)} n$ .

of points in  $P$  to non-unique elements of  $Q$ ; the number of points from  $P$  mapped to any point from  $Q$  is bounded by a small constant. In fact, we can solve a more general bicriterion  $(a, b)$ -approximation problem of the following form: if there is a transformation  $T$  and a function  $f$  which is  $p$ -to-one and with bottleneck value  $r$ , find  $T'$  and  $f'$  which is  $(a \cdot p)$ -to-one and has bottleneck value  $b \cdot r$ . If we moreover require that the fraction of points from  $P$  that are not mapped to a unique point is at most  $\delta$ , we call it a  $(a, b, \delta)$ -approximate solution.

The above formulation generalizes both the Hausdorff distance (for  $p = \infty$  and  $a$  arbitrary) and bottleneck matching distance (for  $p = 1$  and  $a = 1$ ). Moreover, it allows us to achieve a tradeoff between the quality of match (not too many points can be contracted to one) and the complexity of the matching procedures, (which in many cases is almost linear). We present a summary of results in Table 1.

Approximation	Running Time <sup>2</sup>
$(3, 5 + \epsilon)$	$O(\sqrt{kn})$
$(6, 5 + \epsilon)$	$\tilde{O}(n)$
$(4, 5 + \epsilon)^1$	$\tilde{O}(n)$
$(3, 1 + \epsilon)$	$\tilde{O}(2^{\log(1/\epsilon)/\epsilon^{1.25}} \sqrt{kn})$
$(6, 1 + \epsilon)$	$\tilde{O}(2^{\log(1/\epsilon)/\epsilon^{1.25}} n)$
$(4, 1 + \epsilon)^1$	$\tilde{O}(2^{\log(1/\epsilon)/\epsilon^{1.25}} n)$

Table 1: Our Results

All our results are based on the reduction of geometric matching to combinatorial pattern matching; in each case the output is a bit vector  $o$  of translations such that  $o[t] = 1$  iff  $t(P)$  and  $Q$  have a match with the desired properties. All running times that we report are for algorithms that output such a bit vector. The algorithms are Monte Carlo; the probability of error in each case is  $O(1/n^c)$ ,  $c \geq 1$ .

## 1.1 Our Techniques

The existence of a bottleneck matching between two sets of points is a global criterion. In order to apply techniques from combinatorial pattern matching, we need to express this in terms of local criteria. In order to do this, we invoke Hall's theorem, which relates a global property (the size of a matching) to a local property (expansion of subsets).

The expansion of subsets can be captured by the problem of *less-than* matching [AF95] which is a combinatorial pattern matching problem where the pattern and text elements are numbers, and a pattern element is said to *matches* a text element if its value is less than the value of the text element. We show that this problem can be solved approximately in  $\tilde{O}(n)$  time; this result may be of independent interest.

In order to reduce the complexity of our algorithms we need to estimate a certain combinatorial parameter of independent interest. Specifically, for a certain natural family of subsets of the unit square (call it  $F$ ), we need to find a small subset  $F'$  of  $F$  such that for any element from  $F$  there

<sup>1</sup>Although these two schemes appear to strictly dominate the algorithms given above them, they require more time to *construct* an explicit matching. We explain this in more detail in Section 3.2.

<sup>2</sup>The exact dependence on  $\epsilon$  is not shown. For pattern matching under translation, multiply the times by  $1/\epsilon^2$ ; for congruence under translation and rotation, multiply the bounds by  $1/\epsilon^4$ .

exists a member of  $F'$  within distance  $\epsilon > 0$  to it (with respect to Hausdorff metric). For example, for a family of convex polygons, it is known that a sub-family of size roughly  $2^{1/\epsilon^{0.5}}$  exists [Dud74]; the exponent of  $1/\epsilon$  (in this case 0.5) is called the *exponent of entropy* of  $F$  [Dud74]. Our proof uses the result for convex polygons as well as some additional ideas to show that the exponent of entropy for our family is at most 1.25.

We outline the basic method in Section 2. In Section 3, we present improvements to this scheme that yield running times of  $\tilde{O}(n/\epsilon^2)$  by using approximate less-than matching and a result on matching in expander graphs. In Section 4 we utilize the aforementioned combinatorial bounds to improve the dependence on  $\epsilon$ . In Section 5 we sketch a strongly polynomial algorithm that reduces the dependency on  $1/\epsilon$  to polynomial, but can only achieve a constant factor approximation to the bottleneck distance.

## 2 A Reduction To Pattern Matching Based On Hall's Theorem

In this section, we describe a method based on a geometric version of Hall's Theorem that yields an  $(3, 1 + \epsilon)$ -approximation for generalized bottleneck matching under translations. We will focus on the case  $p = 1$  (i.e. assume the existence of a one-to-one mapping), since the generalization to  $p > 1$  is easy. In addition, we present a scheme for the corresponding tricriterion problem, yielding a  $(3, 1 + \epsilon, \delta)$ -approximation.

We start with some definitions. For a point  $p$ , let  $B_r(p)$  denote the ball of radius  $r$  centered at  $p$  i.e  $B_r(p) = \{q \mid \|q - p\| \leq r\}$ . We define  $B(p) = B_1(p)$ . For a set of points  $S$ ,  $B_r(S) = \cup_{s \in S} B_r(s)$ . Further, we represent a translation  $t = (t_x, t_y)$  as a mapping from  $p = (x, y)$  to  $t(p) = (x + t_x, y + t_y)$ . For a set of points  $P$ ,  $t(P) = \cup_{p \in P} t(p)$ .

The idea of the algorithm is as follows. Let the optimal translation be  $t$  and assume that the cost of the matching is 1. Align  $t(P)$  on  $Q$  and impose a brick-wall grid structure on  $R^2$  as shown on Figure 1.

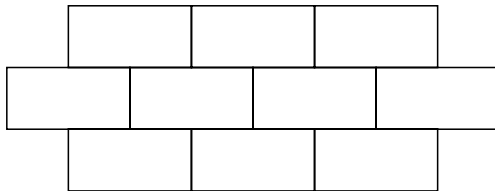


Figure 1: The brick-wall

The width of each grid cell is 4 and the height is 2. Consider any cell (say  $C$ ) and let  $C' = B(C)$  (see Figure 2).

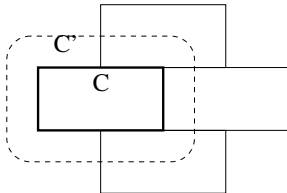
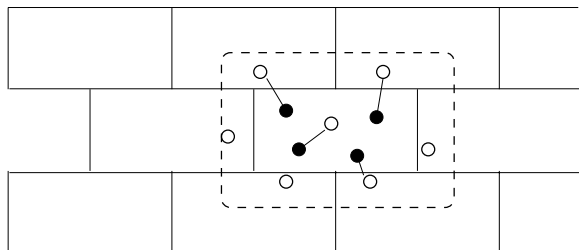


Figure 2: The brick-wall - one cell

A key property of the grid (used later to prove that the solution given by our algorithm is a 3-to-1 matching) is the following:

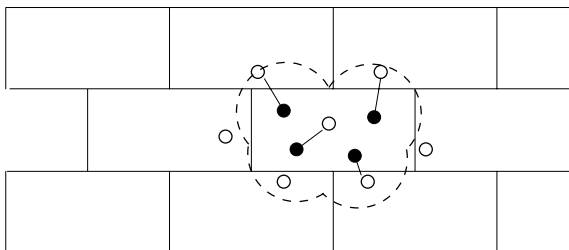
**Claim 2.1** For any point  $p \in R^2$ , the number of cells within distance 1 from  $p$  is at most 3.

Denote  $P_C = t(P) \cap C$  and  $Q_C = Q \cap C'$ . Since the function  $f$  is one-to-one, we know that  $|P_C| \leq |Q_C|$  (since any point from  $P_C$  has to have a corresponding unique point in  $Q_C$ ). We refer to this formula as the *expansion inequality* (See Figure 3).



● Points in P    ○ Points in Q

Figure 3: The expansion inequality



● Points in P    ○ Points in Q

Figure 4: Strong expansion

The expansion inequality guarantees that at the optimal translation,  $|P_C| \leq |Q_C|$  for any cell  $C$ . However, we know that for the optimal matching, a stronger fact is true:  $|P_C| \leq |B(P_C) \cap Q|$  (see Figure 4). We will employ Hall's theorem to exploit this fact.

**Theorem 2.1 *Hall's Matching Theorem*** [Hal35] Let  $G = (A \cup B, E)$  be a bipartite graph. Then  $G$  has a matching of  $A$  into  $B$  iff  $|B(X)| \geq |X|$ , for all  $X \subseteq A$ .

We divide each cell  $C$  of  $G$  into a grid of cell size  $\epsilon$ ; we will assume that  $1/\epsilon = E$  is an integer. Let the  $(i, j)^{\text{th}}$  grid cell be denoted by  $g_{ij}^C$ . A *canonical subset*  $\sigma$  of  $C$  is a subset of  $\{(i, j) | i, j \leq E\}$ . This canonical subset induces a subset  $\sigma_C = \{g_{ij}^C | (i, j) \in \sigma\}$  of  $C$ . The canonical subset corresponding to a subset  $S \subseteq C$  is  $\sigma(S) = \{(i, j) | g_{ij}^C \cap S \neq \emptyset\}$ .

Assume that  $P$  and  $Q$  are aligned according to the optimal translation  $t$ . Consider  $S \subseteq C$ . From Theorem 2.1, it follows that  $(\forall S \subseteq C) |B(S) \cap Q_C| \geq |S \cap P_C|$ . This implies a similar (although slightly weaker) property that can be stated in terms of canonical subsets. Consider a canonical subset  $\sigma$ . Let  $B_E(\sigma) = \cup_{(i,j) \in \sigma} \{(u, v) | |u - i|, |v - j| \leq E\}$ . Then we have<sup>2</sup>

$$(\forall \sigma \subseteq \sigma(C)) |B_{E+1}(\sigma_C) \cap Q_C| \geq |\sigma_C \cap P_C| \quad (1)$$

Our algorithm will find a translation  $t'$  for which equation 1 is true for every cell  $C$ . By Hall's theorem, this implies the existence of a bottleneck matching from  $t'(P) \cap C$  to  $Q$  with value  $1 + O(\epsilon)$ , for every cell  $C$ .

We first define a two-dimensional instance of the “less-than-matching” problem [AF95]:

**Definition 2.1** Given two-dimensional matrices  $x, y$ , find a pair  $(i, j)$  such that for all  $k, l$  we have  $x[k, l] \leq y[i + k, j + l]$ .

A solution to this problem will be used to check the validity of equation 1. Thus, our algorithm for generalized bottleneck matching is as follows:

1. For all canonical subsets  $\sigma$ ,

---

<sup>2</sup>The subscript of  $E + 1$  (instead of  $E$ ) comes from the rounding error of coordinates of  $t$  to nearest multiples of  $\epsilon$ .

2. Let  $x[i, j] = |\sigma_{C_{ij}} \cap P_{C_{ij}}|$ ,  $y[i, j] = |B_{E+1}(\sigma_{C_{ij}}) \cap Q_{C_{ij}}|$ .
3. Solve this instance of **Less-Than Matching**, yielding solution vector  $o_\sigma$ .
4. Compute  $o = \bigwedge_\sigma o_\sigma$ .

The less-than matching problem can be solved by the following method. First, we reduce it to a one-dimensional problem (i.e.  $x$  and  $y$  are now vectors of length  $[0, \dots, N]$ , for some integer  $N$ ) using the random projection technique of [CS98]; this step does not incur any errors with high probability. Then, we reduce the value of  $N$  to  $O(n)$ ; note that since there are  $O(n)$  non-zero entries in the  $x$  and  $y$  matrices,  $O(\log n)$  repetitions of this step are sufficient in order to ensure a high probability of correctness. Each such instance of one-dimensional less-than-matching is then solved in time  $\tilde{O}(\sqrt{kn}/\epsilon^2)$  by using the algorithm of [AF95]. Thus the total running time is  $\tilde{O}(\sqrt{kn}/\epsilon^2)$ .

A cell  $C$  of  $G$  is decomposed into a grid having  $E^2$  cells. Thus, there are  $2^{E^2}$  invocations of a less-than matching algorithm, yielding a running time of  $\tilde{O}(2^{1/\epsilon^2} \sqrt{kn}/\epsilon^2)$ .

All translations  $(i, j)$  corresponding to  $o[i, j] = 1$  yield bottleneck matches. The mapping  $f$  induced by such a match will not be one-to-one in general. However, consider any point  $q \in Q$ . We know that any point  $p \in P$  mapped to  $q$  has to belong to a cell within distance 1 from  $q$ . By Claim 2.1 there are at most 3 such cells, and therefore  $f$  is 3-to-one.

We thus have:

**Theorem 2.2** *There is a  $(3, 1+\epsilon)$ -approximate algorithm for generalized bottleneck matching under translations running in time  $\tilde{O}(2^{1/\epsilon^2} \sqrt{kn}/\epsilon^2)$ .*

**Tricriterion approximation bounds.** In order to limit (say to  $\delta$ ) the fraction of points which are assigned to non-unique image points, we can enlarge the cells by a factor of  $1/\delta$  and translate the grid by a random vector. It is easy to see that the fraction of matching edges close to the boundary of cells is at most  $O(\delta)$  for any fixed translation  $t$ . The remaining edges will be assigned one-to-one. In this way we obtain  $(a, O(1/\delta), \delta)$ -approximation algorithms for constants  $a$  as before.

**Approximate congruence.** Here we discuss how to extend our result for translations to isometries. To this end, we first translate  $P$  such that the centroids of  $P$  and  $Q$  align (say at point  $s$ ). Then, we find a rotation of  $P$  around  $s$  which minimizes its distance to  $Q$ ; we perform it by reducing it to  $O(\log \Delta)$  instances of pattern matching under translations, as in [CS98, IMV99]. By the argument of [ABB91] this approach results in a constant factor approximation algorithm; this can be further reduced to  $1 + \epsilon$  by exploring  $O(1/\epsilon^2)$  points  $s$  [ABB91]. In this way we obtain an algorithm for approximate congruence with essentially the same performance guarantees as the algorithm for translations and with running time larger by a factor of  $1/\epsilon^2$ .

### 3 Near-Linear Matching Schemes

This section (and the next one) are devoted to improving the running time of the above algorithm. In Section 3.1, we will outline a scheme that outputs a  $(6, 1 + \epsilon)$ -approximate solution and has running time  $\tilde{O}(n/\epsilon^2)$ . In Section 3.2 we then show how to improve it to a  $(4, 1 + \epsilon)$ -approximation with the same running time. Although this result appears to strictly dominate the previous one, this is in fact not the case; we explain this in more detail in Section 3.2.

#### 3.1 Approximate Less-Than Matching

The key routine in the previous algorithm was an algorithm to solve the *less-than matching* problem. Here, we define an *approximate* version of this problem, which is allowed to report a match even if

$p[i] > t[i]$  but  $p[i] \leq (1 + \gamma)t[i + t]$  for all  $i$ , where  $\gamma$  is the approximation parameter. Replacing the algorithm for less-than matching by one for approximate less-than problem yields all translation where for each cell  $C$ ,  $|P_C| \leq (1 + \gamma)|Q_C|$ . Unfortunately, this *approximate* expansion property does not yield the required matching; some points of  $P$  may remain unmatched. However, suppose we replace each point in  $Q$  by two points arbitrarily close to each other. Call the resulting set  $Q'$ . We now observe that the expansion property is preserved with respect to  $Q'$ ; specifically, for  $\gamma \leq 1$ ,  $|P_C| \leq (1 + \gamma)|Q_C| \leq (1 + \gamma)/2|Q'_C| \leq |Q_C|$ .

Running the algorithm from Section 2 on  $P, Q'$  yields a  $(3, 1 + \epsilon)$ -approximation algorithm for  $P, Q'$  which can be easily converted into a  $(6, 1 + \epsilon)$ -approximation algorithm for  $P, Q$  by merging the two points in  $Q'$  corresponding to each point in  $Q$ .

It remains to prove that we can solve approximate less-than matching in near-linear time. Recall that the instance consists of two vectors  $p[i], t[i]$ , where  $0 \leq i \leq O(n)$ . We construct  $O(\log n/\gamma)$  binary vectors  $p^j[i], t^j[i]$ , where  $p^j[i] = 1$  if  $(1 + \gamma)^j \leq p[i] < (1 + \gamma)^{j+1}$ , and  $t^j[i]$  is defined similarly. Notice that  $p[i] \leq (1 + \gamma)t[i + t]$  for all  $i$ , if, for all  $i, j, j' \geq j$ ,  $p^j[i + t] \Rightarrow t^{j'}[i]$ . For a fixed  $j$  and  $j'$ , this problem can be solved deterministically in  $O(n \log n)$  time using the convolution-based scheme of Fischer and Paterson [FP74]. There are  $O((\log n/\gamma)^2)$  such instances, yielding a running time of  $O(n \log^3 n/\gamma^2)$  for solving approximate less-than matching.

### 3.2 An Expander-based Matching Algorithm

We now show how the above ideas can be used to obtain a  $(4, 1 + \epsilon)$ -approximate matching in near-linear time. As before, we employ the algorithm for approximate less-than matching on the sets  $P, Q$  to determine all translations where for each cell  $C$ ,  $|P_C| \leq (1 + \gamma)|Q_C|$ . We show that each such translation can be mapped to a  $(4, 5)$ -approximate matching in near-linear time. To do this, we replicate each point in  $Q$  four times. Let the resulting set be  $Q'$ . For a set  $S \subseteq P$ , let  $\mathcal{C}_S$  be the set of cells such that  $S \cap P_C \neq \emptyset$  for  $C \in \mathcal{C}_S$ .

From Claim 2.1 we know that each point in  $Q$  has at most three cells that are adjacent to it. Therefore,  $|\cup_{C \in \mathcal{C}_S} Q_C| \geq \sum_{C \in \mathcal{C}_S} |Q_C|/3$ . The guarantee provided by the less-than matching algorithm implies that

$$\begin{aligned} \sum_{C \in \mathcal{C}_S} |P_C| &\leq \sum_{C \in \mathcal{C}_S} (1 + \gamma)|Q_C| \\ &\leq 3(1 + \gamma)|\cup_{C \in \mathcal{C}_S} Q_C| \\ &\leq 3/4(1 + \gamma)|\cup_{C \in \mathcal{C}_S} Q'_C| \end{aligned}$$

Since  $|S| \leq \sum_{C \in \mathcal{C}_S} |P_C|$ , we have that  $B(S) \geq 4/(3(1 + \gamma))|S|$  for any  $S \subseteq P$ ,  $|S| \leq 3(1 + \gamma)/4|Q|$ . In other words, the bipartite graph defined by  $P$  and  $Q'$  is an expander. In [Mot94], Motwani proved that a perfect matching on a bipartite expander can be computed in time  $O(m \log n/\log \Delta)$ , where  $\Delta$  is the expansion factor and  $m$  is the number of edges in the graph. This algorithm can be modified using the techniques of [EI96] to run in time  $O(n \log n/\log \Delta)$  for bipartite graphs defined on points in the plane.

Thus we obtain a procedure running in time  $O(n \log n/\gamma)$  that returns a matching between  $P, Q'$ . As before, we then merge the four points in  $Q'$  that correspond to a single point in  $Q$ , yielding a  $(4, 1 + \epsilon)$ -approximate matching. Notice that the the process of computing the actual match takes  $O(n \log n/\gamma)$  time, compared to the linear-time simple nearest-neighbour procedures which suffice for the earlier schemes.

## 4 A Combinatorial Lemma

In this section we improve the dependence of the running time on  $\epsilon$  from  $2^{O(1/\epsilon^2)}$  to  $2^{O(\frac{\log 1/\epsilon}{\epsilon^{1.25}})}$ . The basic idea is to reduce the number of sets for which we verify the expansion by using geometry. More specifically, assume that  $\sigma^1 \dots \sigma^s$  are canonical sets such that the Hausdorff distance  $d(B_{E+1}(\sigma^i), (B_{E+1}(\sigma^j))) = O(1)$ .

Then we replace all tests

$$|B_{E+1}(\sigma^k)_C \cap Q_C| \geq |\sigma_C^k \cap P_C|$$

by one test

$$|B_{E+O(1)}(\sigma)_C \cap Q_C| \geq |\sigma_C \cap P_C|$$

where  $\sigma = \cup_k \sigma^k$ . Notice that the second test is implied by the first one. Moreover, if the second test holds for all cells for a translation  $t'$ , then by Hall's theorem we are guaranteed a matching with bottleneck value  $1 + O(\epsilon)$ . Therefore, if we prove that the number of such tests is small, we improve the running time of the algorithm. This number can be bounded from above by using the concept of a cover defined as follows.

**Definition 4.1 (*t*-cover)** *Let  $(X, d)$  be a metric space. A set  $S \subset X$  is called a  $t$ -cover for  $Y \subset X$  ( $t \geq 0$ ) if for any  $p \in Y$  there exists  $q \in S$  such that  $d(p, q) \leq t$ .*

Let  $R = E + 1$  and  $R' = c \cdot R$  for some constant  $c$ . Let the set  $(X, d)$  be the Hausdorff space consisting of subsets of  $[0, R']^2$ . Consider the set  $Y \subset X$  defined as  $Y = \{B_R(S) | S \subset \{0, \dots, R'\}^2\}$ . Observe that the number of tests can be bounded by the size of a 1-cover for  $Y$ . Moreover, for any  $B$  from the cover, we can easily find a union of all  $S$  such that  $d(B_R(S), B) \leq 1$  (corresponding to  $\sigma$  above) simply by taking all  $p \in \{0, \dots, R'\}^2$  such that  $B_R(p) \subset B_1(B)$ . Therefore, it remains to construct a small 1-cover for  $Y$ .

**Theorem 4.1** *There exists a 1-cover for  $Y$  of size  $2^{O(R^{1.25} \log R)}$ .*

**Proof:** Note that (by a scaling argument) it is sufficient to show the existence of an  $O(1)$ -cover. Let  $r = \sqrt{R}$ . Impose a square grid on  $[0, R']^2$  of side  $r$ . We will show that for each grid cell  $C$  the family  $Y_C = \{B \cap C | B \in Y\}$  has an  $O(1)$ -cover of size  $C = 2^{O(\sqrt{r} \log r)}$ . Since there are only  $O(R)$  cells, this implies a  $O(1)$ -cover for  $Y$  of size  $C^{O(R)}$ , which is what we need.

Consider a fixed cell  $C$  and take any  $S \subset \{-2R, \dots, 2R\}$ . We assume that all points  $p \in S$  are *significant*, i.e.  $B(S) \cap C \neq B(S - \{p\}) \cap C$  (otherwise we can remove insignificant points and apply the same arguments). We also assume that  $C$  is not completely included in  $B(S)$ . Notice that in this case for any significant point  $p$  there is a part of a boundary (an arc) of  $B(S)$  which belongs to  $C$ . For each  $p \in S$ , let  $H_p$  be any halfspace containing  $B_R(p)$  and with boundary tangent to the arc of  $B(p)$  as above. We will approximate  $B_R(p) \cap C$  by  $H_p \cap C$  using the following claim.

**Claim 4.1**  $d(B(p) \cap C, H_p \cap C) = O(1)$ .

**Proof:** Follows from the fact that the length of the arc is  $O(\sqrt{R})$  while its radius is  $R$ . ■  
As a corollary we obtain that

$$d(\cup_{p \in S} B_R(p) \cap C, \cup_{p \in S} H_p \cap C) = O(1)$$

Therefore, it is sufficient to construct an  $O(1)$ -cover for the latter set. In fact, we will construct such cover for sets  $\overline{\cup_{p \in S} H_p} \cap C = \cap_{p \in S} \overline{H_p} \cap C$  (where  $\overline{A}$  denotes the complement of  $A$ ). Notice

that this set is a subset of the set  $CP$  of convex polygons contained in  $C$ . Thus it is sufficient to find a  $O(1)$ -cover for  $CP$ . However, it follows from the theorem by Dudley [Dud74] that for any  $P \in CP$  there exists  $P' \in CP$  with only  $O(\sqrt{R})$  vertices such that  $d(P, P') \leq 1$ . Each vertex can be rounded to the nearest point of  $\{-R, \dots, R\}$  and thus can be represented only  $O(\log R)$  bits. This implies that we can represent  $P'$  using only  $O(\sqrt{R} \log R)$  bits. Since the set of all  $P'$ 's is an  $O(1)$ -cover for  $CP$ , the theorem follows.  $\blacksquare$

We mention that the exponent of 1.25 is probably not tight. One can observe however that a 1-cover must have size at least  $2^{\Omega(\sqrt{R})}$ . To see this, consider  $S$  containing  $\sqrt{R}/C$  points (for  $C > 1$ ) uniformly spaced on a horizontal line with  $y = -R$ . One can observe that for sufficiently large  $C$  we can independently move each point up or down by 2 units while always ensuring that the topmost point of a ball centered at any point is not covered by any other ball. Therefore, we can obtain  $2^{\sqrt{R}/C}$  sets such that the distance between any pair of distinct sets is at least 2.

## 5 A simple $O(1)$ -approximation algorithm

We now present a simple algorithm for generalized bottleneck matching under translation, whose running time dependence on  $1/\epsilon$  is polynomial (not exponential). More specifically, we present a  $(3, 5 + \epsilon)$ -approximation that runs in time  $\tilde{O}(n\sqrt{k})$ . Using the techniques of Section 3, we can improve the running time to  $\tilde{O}(n)$ , yielding  $(6, 5 + \epsilon)$ - and  $(4, 5 + \epsilon)$ -approximations.

As in Section 2, let the optimal translation be  $t$  and assume that the cost of the matching is 1. Align  $t(P)$  on  $Q$  and impose a brick-wall grid structure on  $R^2$  as shown on Figure 1.

Now assume that we are given another translation (say  $t'$ ) such that the expansion inequality is ensured for every cell  $C$ . We show that in this case one can find a function  $f'$  which is 3-to-one with bottleneck value equal to the diameter  $\Delta$  of  $C$  (which is  $2\sqrt{5}$ ) plus 1. The construction of  $f'$  is as follows. For every cell  $C$ , we construct a one-to-one mapping  $f_C$  from  $P_C$  to  $Q_C$  (since  $|P_C| \leq |Q_C|$ , such a mapping always exists). The mapping  $f'$  is a union of all  $f_C$ . The bottleneck value of  $f'$  is clearly bounded by  $\Delta + 1$ , and by the arguments in Section 2 we know that  $f'$  is a three-to-one mapping.

It remains to show how to find a translation  $t'$  such that the expansion inequality holds. For each cell  $C$  let  $p_C = (x_C, y_C)$  denotes its center and let  $S$  denote the set of all centers. Let  $h_P : R^2 \rightarrow R$  be a function such that  $h(p_C) = |P_C|$  for any  $p_C \in S$  and is 0 otherwise. Moreover construct a function  $h_Q : R^2 \rightarrow R$  which assigns (to any point  $q$ ) the value of  $|Q \cap B(C)|$ , where  $C$  is a brick cell centered at  $q$ . The problem can be now stated as follows:

$$\text{find a translation } t \text{ such that for all } p \in R^2 \text{ we have } h_{t(P)}(p) \leq h_Q(p)$$

In order to make this problem tractable, we proceed as in [CS98, IMV99]. Since the techniques are similar, we only sketch the argument here. Firstly, we approximate the domain of  $h_P$  and  $h_Q$  by setting  $Z_\epsilon = \{\epsilon \cdot (i, j) | i, j \text{ integers}\}$  (call the resulting functions  $h'_P$  and  $h'_Q$ ). In order to make sure that the expansion inequality still holds, it is sufficient to relax the bottleneck value by at most a factor of  $(1 + O(\epsilon))$ . Notice that the number of function arguments with non-zero values is at most  $O(k)$  for  $h'_P$  and at most  $O(n/\epsilon^2)$  for  $h'_Q$ . By scaling the lattice by a factor  $1/\epsilon$  we can represent  $h'_P$  (resp.  $h'_Q$ ) as (sparse) two-dimensional matrices  $x[i, j]$  (resp.  $y[i, j]$ ) with  $0 \leq i, j \leq M$ , for some integer  $M$ .

Now we invoke the less-than matching algorithm of Section 2, thus obtaining a  $(3, 2\sqrt{5} + 1 + \epsilon)$ -approximation that runs in time  $\tilde{O}(n\sqrt{k})$ .

**A Better Tiling.** We can obtain a better approximation bound by considering other tilings. Consider the hexagonal tiling (with side length 2) depicted on Figure 5.

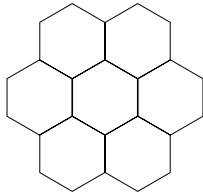


Figure 5: A hexagonal tiling

One can observe that Claim 2.1 holds for this tiling as well. Since the diameter of a hexagonal cell is 4, we obtain the following:

**Theorem 5.1** *There is a  $(3, 5 + \epsilon)$ -approximation for generalized bottleneck matching under translation that runs in time  $\tilde{O}(n\sqrt{k})$ .*

## 6 Conclusions

It seems likely that the 1.25 exponent of  $1/\epsilon$  (from Section 4) can be improved. We believe that the right value is 0.5 (as for convex polygons). Unfortunately, the proof appears complex since the (complement) of a union of balls is not convex; in fact, the shape even does not have to be connected.

It might be possible to reduce the dependence on  $\epsilon > 0$  from exponential to polynomial, and still maintain the  $(1 + \epsilon)$ -approximation of the matching cost. One possibility is to encode the RNC algorithm for bipartite matching testing using algebraic convolutions (in fact, one can view our method as a constant time parallel algorithm with exponential number of processors). However, at present we do not know if this is possible at all.

## References

- [ABB91] H. Alt, B. Behrends, and J. Blömer. Approximate matching of polygonal shapes. In *Proc. 7th Annu. ACM Sympos. Comput. Geom.*, pages 186–193, 1991.
- [AF95] A. Amir and M. Farach. Efficient 2-dimensional approximate matching of half-rectangular figures. *Information and Computation*, 118(1):1–11, 1995.
- [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<sup>+</sup>92] E.M. Arkin, K. Kedem, J.S.B. Mitchell, J. Sprinzak, 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. Wagnen, and E. Welzl. Congruence, similarity, and symmetries of geometric objects. *Discrete Computational Geometry*, 3:237–256, 1988.
- [CGH<sup>+</sup>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.
- [CS98] D. Cardoze and L. Schulman. Pattern matching for spatial point sets. In *Thirty-ninth Annual Symposium on the Foundations of Computer Science*. IEEE, November 1998.

- [Dud74] R. M. Dudley. Metric entropy of some classes of sets with differentiable boundaries. *Approximation Theory*, 10:227–236, 1974.
- [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.
- [FHK<sup>+</sup>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<sup>+</sup>97] P. Finn, L. E. Kavradi, J. C. Latombe, R. Motwani, 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.
- [FP74] M. J. Fischer and M. S. Paterson. *Complexity of Computation*, chapter String Matching and Other Products, pages 113–125. SIAM-AMS, 1974.
- [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.
- [Hal35] P. Hall. On representatives of subsets. *Journal of London Mathematics Society*, 10:26–30, 1935.
- [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.
- [IMV99] P. Indyk, R. Motwani, and S. Venkatasubramanian. Geometric matching under noise: Combinatorial bounds and algorithms. In *8th ACM-SIAM Symposium on Discrete Algorithms*, pages 457–465, 1999.
- [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.
- [Mot94] R. Motwani. Average-case analysis of algorithms for matchings and related problems. *Journal of the ACM*, 41:1329–1356, 1994.
- [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.