**Nearest Neighbor Search using Sketches as Quantized Images of Dimension Reduction**

Naoya Higuchi$^1$, Yasunobu Imamura$^1$, Tetsuji Kuboyama$^2$, Kouichi Hirata$^1$ and Takeshi Shinohara$^1$

1*Kyushu Institute of Technology, Kawazu 680-4, Itzuka 820-8502, Japan  
2Gakushuin University, Mejiro 1-5-1, Toshima, Tokyo 171-8588, Japan

Keywords: Similarity Search, Sketches, Ball Partitioning, Hamming Distance, Dimension Reduction, Distance Lower Bound, Quantized Images.

Abstract: In this paper, we discuss sketches based on ball partitioning (BP), which are compact bit sequences representing multidimensional data. The conventional nearest search using sketches consists of two stages. The first stage selects candidates depending on the Hamming distances between sketches. Then, the second stage selects the nearest neighbor from the candidates. Since the Hamming distance cannot completely reflect the original distance, more candidates are needed to achieve higher accuracy. On the other hand, we can regard BP sketches as quantized images of a dimension reduction. Although quantization error is very large if we use only sketches to compute distances, we can partly recover distance information using query. That is, we can compute a lower bound of distance between a query and a data using only query and the sketch of the data. We propose candidate selection methods at the first stage using the lower bounds. Using the proposed method, higher level of accuracy for nearest neighbor search is shown through experimenting on multidimensional data such as images, music and colors.

1 INTRODUCTION

A similarity search is one of important and famous tasks for information retrieval in multidimensional data. The purpose of the similarity search is to find data near to a query with respect to a given distance. In order to avoid “the curse of dimensionality,” dimension reduction techniques have been developed. One of the most important properties of dimension reduction is that the distance between any two data is not expanded after transformation. In other word, dimension reduction is a Lipschitz continuous mapping. This property establishes the safety of pruning strategy that excludes any data, which is far away from a query in the lower dimensional space, from the search target without examining the distance in the original space. For example, K-L transformation (or equivalently principal component analysis (PCA)) (Fukunaga, 1990) and FastMap (Faloutsos and Lin, 1995) are known as dimension reductions for a Euclidean space. Also H-Map (Shinohara et al., 1999) and Simple-Map (S-Map, for short) (Shinohara and Ishizaka, 2002) are dimension reductions applicable to any metric space such as $L_1$ metric space and strings with edit distance (Wagner and Fischer, 1974).

As another technique for efficient similarity search in multidimensional spaces, sketch (Müller and Shinohara, 2009; Mic et al., 2016; Dong et al., 2008; Mic et al., 2015; Wang et al., 2007) has also been developed. Sketch is a compact bit sequence representing multidimensional data. Ball partitioning (Uhlmann, 1991) (BP, for short) is a method to make sketches by assigning a bit 0 or 1 to a data, such that 0 if it is in a ball and 1 otherwise. BP is also used in vantage point tree (Yianilos, 1993).

Conventionally, the similarity search using sketches consists of two stages. The first stage selects candidates depending on their Hamming distances between sketches. Here, we should note that the computational cost of the Hamming distance is very small if we use efficient bit operations such as exclusive or and bit count. The second stage selects the nearest neighbor by comparing the candidates with the query using distances in the original space. As the Hamming distance cannot preserve the information of distances between data, similarity search using sketches
is only an approximation method. One of the most important tasks for sketches is achieving high accuracy with small number of candidates obtained at the first stage.

When the width $w$ of sketches is considered as the dimensionality, the dimensionality may not be reduced. However, as the size $w$ bit is usually much smaller than the original data, we may consider mapping to sketches as a quasi-dimension reduction. Nevertheless, the Lipschitz continuity of the mapping is not guaranteed as long as the Hamming distance is used.

On the other hand, since the sketch of $w$ bits is defined by using $w$ BP’s, we can regard $w$-bits sketches as quantized images of S-Map (Ohno, 2011). That is, BP sketches are quantized images of S-Map, where each axis value is quantized to one bit depending on whether or not greater than thresholds. Note that the $L_w$ distance should be used to guarantee the Lipschitz continuity of S-Map. Any $L_w$ distance between sketches is 0 or 1, that is, the quantization error is very large. As for data in the database, we should use only sketches at the first stage because the original high dimensional data are too large. However, as for queries, we can use the original queries as well as their sketches. Hence, in this paper, we show that, for each sketch bit, a lower bound of distance between a query $q$ and data $x$ can be calculated using $q$ and the sketch of $x$ without the original $x$. If we take score$_w$ defined by the maximum of distance lower bounds as the aggregation like $L_w$ distance, the distance estimation using sketch is not expanded and a BP sketch mapping can be considered as a quasi-dimension reduction. Similar idea is also found in asymmetric distance estimation (Dong et al., 2008; Jain et al., 2011; Balu et al., 2014), where sketches are constructed by generalized hyperplane partitioning (GHP), some of them assume the Euclidean distance or cosine distance, and their estimation may expand the distance.

For $w$ bit sketches, we have $w$ distance lower bounds. To guarantee distance lower bounds, we have to use the aggregation by maximum score$_w$. In approximate nearest neighbor search using sketches, we propose score$_1$ and score$_2$, which are the aggregations by sum and square sum respectively, which are no longer distance lower bounds. By experimenting on images, music and colors databases, we observe that score$_1$ and score$_2$ achieve a more accurate nearest neighbor search compared to the Hamming distance and score$_w$. Naive implementation of aggregation needs more computational cost than the Hamming distance using bit operations. For each query we can precompute aggregations to construct a table function. The cost for aggregation using table function is almost comparable with the cost for the Hamming distance using bit operations. We can ignore this preprocess cost if we search large database.

We believe that our contribution lies in the following three points. First, in any metric space, based on the observation that BP sketches are quantized images of dimension reduction S-Map, we point out that the sketch mapping can be considered as a quasi-dimension reduction by aggregating distance lower bounds between the query and a sketch for each bit of the sketch in $L_w$ manner. Similar methods are used elsewhere (Charikar, 2002; Jain et al., 2011), where sketches are based generalized hyperplane partitioning (GHP) in Euclidean distance and cosine distance, and aggregated distance lower bounds is just a distance estimation but lower bound. Here, we also point out that GHP sketches are quantized images of H-Map. Thus, our approach is easily extended to applicable to any GHP based sketches. Second, we propose a low cost method to compute aggregations using precomputed table functions. This contribution is not very significant but important for practical problems. In our setting, we assume that data are made by feature extraction function and they are not very high dimension. Therefore, we cannot ignore the computational cost for the first stage. Third, we propose sum or square sum aggregations of distance lower bounds for the priority at the first stage. Such aggregations are no longer distance lower bound. Nevertheless they are more useful than maximum aggregation. Similar techniques are found elsewhere, where GHP based sketches are considered in Euclidean distance or cosine distance. Our method is applicable to BP sketches in any metric space and easily extended to GHP.

2 PRELIMINARIES

Here, we briefly introduce some necessary concepts for our discussion.

2.1 Dimension Reduction and Simple-Map

We assume two metric spaces $(U, D)$ and $(U’, D’)$, where $D$ and $D’$ are distance functions satisfying triangle inequality. Let $\dim(x)$ for a data $x$ denote the dimensionality of $x$. Then, we say that a mapping $\phi : U \rightarrow U’$ is a dimension reduction if it satisfies the following conditions for every $x, y \in U$:

$$\dim(\phi(x)) < \dim(x) \quad (1)$$

$$D’(\phi(x), \phi(y)) \leq D(x, y) \quad (2)$$

357
Condition (1) means that $\phi$ reduces the dimensionality of data. However, it is not handled very strictly in this paper. For example, the size required to represent data is regarded as the dimensionality. Condition (2) means that $D'$ provides the lower bound of a distance $D(x,y)$, which guarantees to ignore a data without computing $D$ in similarity search. For example, if $D'(\phi(q),\phi(x))$ exceeds the current search diameter of a query $q$, then $x$ can be ignored without computing $D(q,x)$.

A Simple-Map (S-Map, for short) (Shinohara and Ishizaka, 2002) is a kind of Fréchet embedding (Matoušek, 2002) that any finite metric space of $n$ points can be embedded isometrically into $n$-dimensional $L_\infty$ normed space. A similar idea has also been proposed by Hjalto and Samet (Hjalto and Samet, 2003). For a point $p \in U$ called a pivot, we define an S-Map $\phi_p$ of $x \in U$ with $p$ as follows.

$$\phi_p(x) = D(p,x).$$

By triangle inequality, the following inequality holds for every $x, y \in U$:

$$|\phi_p(x) - \phi_p(y)| \leq D(x,y).$$

Furthermore, for a set $P = \{p_1, \ldots, p_m\}$ of pivots, we define S-Map $\phi_P$ with $P$ as follows.

$$\phi_P(x) = (\phi_{p_1}(x), \ldots, \phi_{p_m}(x)).$$

Suppose that we give $D'$ as follows:

$$D'(\phi_P(x),\phi_P(y)) = \max_{i=1}^m |\phi_{p_i}(x) - \phi_{p_i}(y)|.$$

In other words, if the projected space $U'$ is considered as an $L_\infty$ metric space, then, an S-Map is a dimension reduction.

H-Map (Shinohara et al., 1999), using a pair of two points $(p_1, p_2) \in U$ as a pivot, we define an H-Map $\phi_{(p_1,p_2)}$ of $x \in U$ as follows.

$$\phi_{(p_1,p_2)}(x) = \frac{D(p_1,x) - D(p_2,x)}{2}.$$  

Then, H-Map is also a dimension reduction.

### 2.2 Nearest Neighbor Search using Sketches

We assume that data in the given database are indexed by natural numbers 1 to $n$. Thus, let $db = \{x_1, \ldots, x_n\}$ be the given database of $n$ data. The dissimilarity between two data $x_i$ and $x_j$ is defined as distance $D(x_i,x_j)$. The nearest neighbor search for a query $q$ is to find $x \in db$ such that $D(q,x) \leq D(q,y)$ for all $y \in db$. Let $s$ be a function which maps data to its sketch. We can realize the $k$-nearest neighbor search using sketches as follows, where $K > k$.

1. **Preparation stage:**
   Calculate all the sketches $s(x_1), \ldots, s(x_n)$.

2. **First stage (Filtering using the Hamming distances of sketches):**
   Calculate the sketch $s(q)$ of query $q$.
   Calculate all the Hamming distances of sketches from $s(q)$.
   Select the closest $K$ sketches $s(x_i), \ldots, s(x_k)$ to $s(q)$.

3. **Second stage (Nearest neighbor search using actual distances):**
   Select the $k$ nearest neighbor data from the candidates $x_1, \ldots, x_k$.

Sketches are relatively small structures with respect to their original feature data. For example, we use sketches of 32 bits for image feature data of 64 bytes in our experiments. At the first stage of searching process, we use the Hamming distances, which can be more easily calculated using bit operations than the actual distances between features. However, sketches cannot preserve all the distance relation. Therefore, we use them as a filter. The larger $K$ of the number of candidates at the first stage achieves a more accurate but slower search. Thus, one of the most important subjects on sketch is to achieve higher accuracy with smaller $K$, or equivalently, to speed up search within acceptable error.

### 2.3 Sketches based on Ball Partitioning

In this paper, we consider sketches based on ball partitioning (BP). A pair $(p,r)$ of a point and a radius is called a pivot. A ball partitioning $BP_{(p,r)}$ is defined as follows:

$$BP_{(p,r)}(x) = \begin{cases} 0 & \text{if } D(p,x) \leq r \\ 1 & \text{otherwise} \end{cases}$$

A BP based sketch function $s_p$ of width $w$ is defined by a set of $w$ pivots $P = \{(p_1,r_1), \ldots, (p_w,r_w)\}$ as follows:

$$s_p(x) = BP_{(p_1,r_1)}(x) \ldots BP_{(p_w,r_w)}(x)$$

Consider 4 points $A, B, C, D$ on a Euclidean plane in Figure 1. Using a set of two pivots $P = \{(p_1,r_1), (p_2,r_2)\}$, their sketches are $s_p(A) = 01$, $s_p(B) = 00$, $s_p(C) = 10$, $s_p(D) = 11$.

Let $q$ be any query outside of both balls. Since $s_p(q) = 11$, Hamming distances between sketches of $q$ and $A, B, C, D$ are 1, 2, 1, 0, respectively. The order
of conventional priority at the first stage is $D < A = C < B$. Note that $A$ and $C$ cannot be distinguished by Hamming distances from $q$.

### 2.4 Pivot Selection for Sketches using Binary Quantization

The pivot selection for sketch is a very important problem but outside of our scope. Here, we briefly introduce a heuristic algorithm $\text{SELECTPivotQB}$, which is not the best search algorithm but can find a relatively good set of pivots within a small computation time.

*Binary quantization* (BQ) selects one of the $2^n$ corners of $n$ dimensional feature space, which are too many to efficiently be selected. BQ randomly chooses a data from database and quantizes it to a corner according with the median.

In our experiments, we adopt probability of collisions between sketches as the evaluation function of pivots. If two different data $x$ and $y$ have the same sketch, we say a collision occurs. From experience, sketches with less collisions provide more accurate nearest neighbor search. The algorithm $\text{SELECTPivotQB}$ in Algorithm 15 illustrates the algorithm for selecting pivots of QBP, where $D$ is the distance function and $\text{eval}$ is the evaluation function for pivot.

### 3 DISTANCE LOWER BOUNDS BETWEEN QUERIES AND SKETCHES

In this section, we introduce a method to compute distance lower bounds between queries and sketches.

First, let us explain the idea using an example. Consider four points $A, B, C, D$ and a query $q$ in Figure 2.

The sketch $s^p(q)$ has 1 at the leftmost bit while $s^p(A)$ has 0. Therefore,

$$D(p_1, A) \leq r_1.$$  

By triangle inequality,

$$D(q, A) \geq D(p_1, q) - D(p_1, A).$$

From these two inequalities,

$$D(q, A) \geq D(p_1, q) - r_1.$$  

The lower bound of distance between $q$ and any point whose leftmost sketch bit is 0 is given by (as $e_1$ in Figure 2),

$$e_1(q, 0^*) = D(p_1, q) - r_1.$$  

Similarly, the lower bound of distance between $q$ and $C$ is

$$e_2(q, 0^*) = D(p_2, q) - r_2.$$  

Here, note that the above lower bounds can be calculated only from $D(p_1, q), D(p_2, q), r_1$ and $r_2$ without $D(q, A)$ and $D(q, C)$.

More formally, we can compute the lower bounds as explained below. Let $p_i$ and $r_i$ be the center and radius of $i$-th pivot. Assume $i$-th sketch bits for a query $q$ and data $x$ are different from each other, that is, $D(p_i, q) > r_i$ and $D(p_i, x) \leq r_i$ (or $D(p_i, q) \leq r_i$ and $D(p_i, x) > r_i$). This difference is extremely quantized as 1 in their Hamming distance. Calculating the distances $D(p_i, x)$ for all $x$ in database require more computational costs. On the other hand, the distance $D(q, x)$ is partly recovered using $D(p, q)$ only.

When

$$D(p_i, q) > r_i \quad \text{and} \quad D(p_i, x) \leq r_i,$$
In this section, we report experiments using several data, which are images, music and colors, as follows:

On the other hand, the Hamming distance is the sum of the differences. We also examine the sum and the square sum of the distance lower bounds

\[
\text{score}_1(q, s(x)) = \sum_{i=1}^{w} e_i(q, s(x)) \\
\text{score}_2(q, s(x)) = \sum_{i=1}^{w} (e_i(q, s(x)))^2
\]

as the priorities. Although they are no longer theoretical lower bounds, experiments show that they bring a more accurate nearest neighbor search.

In the conventional sketch search, the Hamming distance is used as the priority at the first stage, and it is advantageous that high speed calculation is possible by bit operations. In the proposed method, since the distance lower bounds are obtained for a large number of data for each question, the cost increase can be reduced by preparing a table function. The cost increase can almost be ignored.

4 EXPERIMENTS

In this section, we report experiments using several data, which are images, music and colors, as follows:

Algorithm 1: SELECTPivotQBP.

by triangle inequality, as shown in the left of Figure 3,

\[ D(q, x) \geq D(p_i, q) - D(p_i, x) \geq D(p_i, q) - r_i. \]

Similarly, when

\[ D(p_i, q) \leq r_i \quad \text{and} \quad D(p_i, x) > r_i, \]

we have, as shown in the right of Figure 3,

\[ D(q, x) \geq D(p_i, x) - D(p_i, q) \geq r_i - D(p_i, q). \]

Thus, we have the lower bound \( e_1(q, s(x)) \) of \( D(q, x) \) by

\[
e_1(q, s(x)) = \begin{cases} 0 & \text{if } s_i(q) = s_i(x) \\
|D(p_i, q) - r_i| & \text{if } s_i(q) \neq s_i(x) \end{cases}
\]

We propose priorities using the distance lower bound \( e_1(q, x) \) as the criteria to select candidates at the first stage. When we use as the priority

\[
\text{score}_w(q, s(x)) = \max_{i=1}^{w} e_i(q, s(x))
\]

which is the maximum lower bound, we can safely prune some of candidates because it is really a distance lower bound.
• images: about 70 million 2D frequency spectrums of 64 dimension data extracted from 2,900 videos.
• music: about 70 million mel-frequency spectrums of 96 dimension data extracted from 1,400 music CD.
• colors: about 110,000 data of 112 dimension from SISAP database.

We adopted 32 bit as the width of sketches. 32 pivots are selected by SELECTPIVOTQBP with TRIAL = 1000. For each database, we selected 10 different sets of pivots. The experimental results of precision show the average of these 10 sets of pivots.

Randomly generated data are not appropriate for experiments of nearest neighbor search, because in higher dimensional spaces it is rare to find near data. Therefore, we prepare five types of queries: very-near, near, middle, far, very-far which are generated from randomly selected pairs from database with mixing noise ratio of 5 – 10%, 15 – 20%, 25 – 30%, 35 – 40%, 45 – 50%, respectively. For example, a very-near query \( q \) is a weighted sum of randomly selected data \( x \) and \( y \) from database with weight 10% and 90%, respectively. For each noise level, we prepare 100 queries. The average of nearest neighbor distances for queries are shown in Figure 4.

First, we compare run times using priorities of the Hamming distance, \( \text{score}_\infty \), \( \text{score}_1 \), and \( \text{score}_2 \). Table 1 illustrates the computer environment.

<table>
<thead>
<tr>
<th>CPU</th>
<th>Intel(R) Xeon(R) CPU E5-2640 2.5 GHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>memory</td>
<td>64GBytes</td>
</tr>
</tbody>
</table>

Table 2 shows the average run time per query in millisecond. From this table, no significant difference between methods is observed. Thus, computational costs for distance lower bounds and their aggregation are almost the same as that for the Hamming distance. In this table, only the results for images and music databases are shown, because the run time for colors is very small.

<table>
<thead>
<tr>
<th>DB</th>
<th>images</th>
<th>music</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K )</td>
<td>0.10%</td>
<td>1.0%</td>
</tr>
<tr>
<td>Hamming</td>
<td>19.5</td>
<td>21.1</td>
</tr>
<tr>
<td>( \text{score}_1 )</td>
<td>16.8</td>
<td>23.0</td>
</tr>
<tr>
<td>( \text{score}_2 )</td>
<td>22.5</td>
<td>23.4</td>
</tr>
<tr>
<td>( \text{score}_\infty )</td>
<td>15.2</td>
<td>16.9</td>
</tr>
</tbody>
</table>

The precision of the search is defined as the probability that the top \( K \) candidates of the first stage include the exact nearest neighbor. We report the results for \( K \) of three sizes 0.01%, 0.1% and 1.0% relative to database size.

Figure 5, 6 and 7 summarize the precisions of search for databases images, music and colors, respectively, where \( \text{score}_2 \) is omitted, because it is very similar to \( \text{score}_1 \).

5 CONCLUDING REMARKS

From the experiments, we confirmed that the precision is improved by the proposed methods using the
distance lower bounds, their maximum score\(_\infty\), their total sum score\(_1\), and their total square sum score\(_2\), compared to using the Hamming distance. One of the most important tasks for the future work of this research may be to examine why score\(_1\) and score\(_2\) give better priorities than score\(_\infty\) and Hamming distance. Since all priorities in these experiments have a similar tendency to any of the three databases of image, music and colors, we think that the dependency of database is unlikely. We should consider the influence of the width \(w\) of sketches, because score\(_\infty\) will improve by the larger \(w\), but score\(_1\) and score\(_2\) may get worse. There is a possibility that the same phenomenon may also occur in S-Map, where \(L_\infty\) is used in projected space. This is also a future issue.

In the experiments we reported, we optimized sketches by random trials using binary quantization. We can apply optimization techniques such as local search and simulated annealing. We run annealing by increasing resampling (AIR) introduced by Imamura et.al (Imamura et al., 2017) on pivot selection for sketches to minimize collisions. Although sketches with smaller collision probability are found by using them, the search precision is not improved. Therefore, we have to consider other evaluation function for sketches rather than the probability of collision. For example, it is conceivable to use the correlation coefficient of distances before and after projection between sample pairs and the distance preservation ratio of the distance lower bounds as the evaluation function.

In this paper, we consider sketches based on ball partitioning, which can be considered as quantized images of S-Map. Generalized hyperplane partitioning (Uhlmann, 1991) (GHP) can also be used to make sketches. GHP based sketches can be considered as quantized images of H-Map. We can calculate
distance lower bounds between queries and sketches even for GHP based sketches. We should also investigate the effectiveness of proposed methods for GHP.

We compare aggregation methods $\text{score}_1$ and $\text{score}_2$ in addition to $\text{score}_\infty$. In our experiments, their performances are better than the traditional method using the Hamming distances. The $\text{score}_\infty$ gives a theoretical distance lower bound. Therefore, for $K$ objects selected at the first stage based on $\text{score}_1$ or $\text{score}_2$, safe pruning based on $\text{score}_\infty$ is applicable. At the second stage, any candidate with $\text{score}_\infty$ larger than the actual distance of the provisional nearest neighbor can be removed without computing the actual distance. We also have to consider $\text{score}_p$ aggregation for $p$ other than 1 and 2, such as 1.5.

REFERENCES


