Quantifying the invariance and robustness of Permutation-based Indexing schemes

Stéphane Marchand-Maillet[1], Edgar Roman-Rangel[1], Hisham Mohamed[1], and Frank Nielsen[2]

[1] Department of Computer Science, University of Geneva, Switzerland,
stephane.marchand-maillet@unige.ch
[2] LIX Polytechnique, Paris, France

Abstract. Providing a fast and accurate (exact or approximate) access to large-scale multidimensional data is a ubiquitous problem and dates back to the early days of large-scale Information Systems. Similarity search, requiring to resolve nearest neighbor (NN) searches, is a fundamental tool for structuring information space. Permutation-based Indexing (PBI) is a reference-based indexing scheme that accelerates NN search by combining the use of landmark points and ranking in place of distance calculation.
In this paper, we are interested in understanding the approximation made by the PBI scheme. The aim is to understand the robustness of the scheme created by modeling and studying by quantifying its invariance properties. After discussing the geometry of PBI, in relation to the study of ranking, from empirical evidence, we make proposals to cater for the inconsistencies of this structure.

Keywords: Permutation Based Indexing, ranking, geometry

1 Introduction

Providing a fast and accurate (exact or approximate) access to large-scale multidimensional data is a ubiquitous problem and dates back to the early days of large-scale Information Systems. The approach generally taken is that of defining a structure of the space based on information similarity and to partition the information space according to this structure for quantized or hierarchical access. The most common base for structuring the space is to assume the existence of a relevant metric in the space and to base the indexing on the properties of that metric space to resolve the Nearest Neighbor (NN) search problem. From there, a large variety of indexing techniques have been defined [36,10,37,29].

In this paper, we are interested in a finer understanding of the approximations made by the PBI scheme (and, more generally, permutation-based distance measurements). In particular, the aim is to understand the robustness of the scheme created by quantifying its invariance properties. The main contributions is the definition of a formal space partitioning model for the PBI scheme, embarking power tools from geometry modeling.

We demonstrate the validity of our proposal with extensive empirical evidence. In this paper, we are interested in understanding the approximation made by the PBI scheme. The aim is to understand the robustness of the scheme created, or conversely, quantify its invariance properties. After discussing the geometry of PBI, in relation to the study of ranking, from extensive empirical evidence, we make proposals to cater for the inconsistencies of this structure.

2 Related work

A large family of indexing techniques is that of reference-based indexing schemes, where some reference points (sometimes referred to as pivots or anchor points) are selected, based on their local or global properties and then organized for facilitating query resolution and data access. In the list of such structures, we can cite tree-based indexing that place a hierarchical structure over these pivots. These include BK-Tree [6], Vantage Point Tree [32,23] or M-Tree (Metric Tree) [11]

More recent structures such as the Fixed Query Array [9], M-Index [24] or Permutation Based Indexing [8] use pivots to partition the space and to encode the data according to the structure of the partition. These structures have a number of parameters on which their actual performance depend and their choice are generally made empirically, either based on heuristics or on the statistics of the data in question [1,3,7,4]. However, a formal modeling of the relationship between these choices and the impact on the performance, based on a sound modeling of the encoding created by the indexing scheme is still missing [2,20].

They also relate to the statistical properties of high-dimensional representation spaces within which the *curse of dimensionality* applies [34,5,13,33]. Although indexing performance decreases in such a setup and hardware advances (such as GPU computations) allow brute-force exhaustive search to be fast and robust [17,12], it is still relevant to look at indexing structures acting either within subspaces or data manifolds [35].

We have studied how PBI may be distributed over parallel architectures [19], how PBI schemes may be simplified (pruned) to scale while preserving an adequate level of approximation [20]. We have worked on large-scale data processing, including with GPU processing [22,21,14]. Here, we extend an initial modeling for the geometry of PBI [18].

3 Formal modeling of Permutation-based Indexing schemes

We follow and adapt notations from [16,8]. Given $\mathcal{U} = \{o_1, \ldots, o_N\}$ a collection (universe) of N D-dimensional objects $o_i \in \mathbb{R}^D$, and given a continuous distance function $d(.,.)$ operating on objects, typically any Minkowsky distance (including the Euclidean distance d_E) or other classical distance function (including the cosine similarity distance).

We choose from $\mathcal{U}$ a set of n $(0 < n \leq N)$ *reference objects* $R = \{r_1, \ldots, r_n\}$ where, for every k, $r_k = o_i$ for some i.

Definition 1 (Ordered list). *Given $o_i \in \mathcal{U}$, we define the* ordered list *of object o_i as the permutation $\pi_i : [\![1, n]\!] \to [\![1, n]\!]$ such that for all $k \in [\![1, n-1]\!]$:*[3]

$$\begin{cases} d(r_{\pi_i(k)}, o_i) < d(r_{\pi_i(k+1)}, o_i) \\ \text{or } d(r_{\pi_i(k)}, o_i) = d(r_{\pi_i(k+1)}, o_i) \text{ and } \pi_i(k) < \pi_i(k+1) \end{cases}$$

We note $\boldsymbol{\pi_i} = (\pi_i(1), \ldots, \pi_i(n))$.
Given $p \in \mathbb{R}^D$, we note π_p the ordered list of any point p.

In other words, $\boldsymbol{\pi_i}$ is the list of indices of the reference objects r_k sorted in increasing distance values from o_i. To remove randomness completely from the ranking, in case of a tie on distances, the reference object of lower index appears first in the list.

Viewing the ordered list as a bijective function, we can define π_i^{-1} as its inverse function, providing the position of a reference object in the ordered list.

We also extend the notation to apply the function π_i (resp π_i^{-1}) on ordered sets. In that case, for example, $\pi_i^{-1}(J) = (\pi_i^{-1}(j_1), \ldots, \pi_i^{-1}(j_l))$, where $J = (j_1, \ldots, j_l)$.

The function π_i *encodes* the position of object o_i with respect to the list of reference objects R and it is the purpose of this paper to study further the properties of π_i.

Based on this position encoding, we can define a new distance approximation using any distance that can be computed between rankings (ordered lists). The Spearman Footrule Distance (SFD) based on set R or the Spearman Rho (ρ) are typically used:

$$\delta_R(o_i, o_j) = \sum_{k=1}^{n} \left| \pi_i^{-1}(k) - \pi_j^{-1}(k) \right| \tag{1}$$

$$\rho_R(o_i, o_j) = \sqrt{\sum_{k=1}^{n} \left(\pi_i^{-1}(k) - \pi_j^{-1}(k) \right)^2} \tag{2}$$

It has been shown that such distance functions can be used to resolve the k nearest neighbor problem (kNN) since δ_R and ρ_R approximate, in terms of ranking, continuous distances for the search of kNN [8]. In other words, for example,

$$\delta_R(o_i, o_j) \overset{\text{rank}}{\simeq} d(o_i, o_j) \qquad \forall o_i, o_j \in \mathcal{U} \tag{3}$$

Hence, *Permutation-based Indexing* (PBI) aims at facilitating and optimizing, for any query q ($q \in \mathcal{U}$ or $q \notin \mathcal{U}$) the computation of rank-based distances such as $\delta_R(q, o_i)$ for all $o_i \in \mathcal{U}$.

We will base our formal analysis on δ_R but, unless otherwise stated, any other rank-based distance function (such as ρ_R) may apply instead.

[3] We use the compact notation $[\![1, n]\!] = \{1, \cdots, n\}$ for sets of successive integers.

3.1 Invariance

Computing distances over ordered lists creates distance approximations, which in turn create equivalence relationship.

Definition 2 (Equivalence relationship). *Given $R \subset \mathcal{U}$ and $o_i, o_j \in \mathcal{U}$, we note $o_i \equiv o_j$ if and only if*

- $\delta_R(o_i, o_j) = 0$,
- *equivalently,* $\boldsymbol{\pi_i = \pi_j}$ *(since δ_R is a distance function)*

Definition 3 (Equivalence class - Invariance). *The* equivalence class *of object o_i is*

$$[o_i] = \{p \in \mathbb{R}^D \text{ such that } p \equiv o_i\}$$

The quotient space $\mathcal{U}/\equiv$ is the set of all equivalence classes of δ_R from $\mathcal{U}$.

The equivalence class is the set of all positions p an object can take in the initial space without changing its encoding in the permutation space. As an immediate consequence, the value of the δ_R distance between any pair of points of respective classes does not vary. Hence, the equivalence classes show the extent of the invariance of the π_i encoding. Similarly, the equivalence classes measure the approximation made by the distance function δ_R.

We now construct a geometric structure for analyzing the PBI scheme.

3.2 Geometry

Objects o_i are points of the $\mathbb{R}^D$ space over which some geometrical properties can be inferred. We use the Euclidean distance in $\mathbb{R}^D$ but this analysis may be extended with using other metrics.

A D-dimensional space may be partitioned by $(D-1)$-dimensional hyperplanes. In our context, perpendicular bisectors are particular such hyperplanes.

Definition 4 (Perpendicular bisector). *Given $r_k, r_l \in R$, we define Δ_{kl} as the $(D-1)$-dimensional* perpendicular bisector[4] *of the segment $[r_k, r_l]$.*

Proposition 1. *If two given objects $o_i, o_j \in \mathcal{U}$ are separated by Δ_{kl} then*

$$(\pi_i^{-1}(k) - \pi_i^{-1}(l)).(\pi_j^{-1}(k) - \pi_j^{-1}(l)) < 0$$

If Δ_{kl} is the only bisector separating o_i and o_j, then in that case, in particular, $\delta_R(o_i, o_j) = 2$.

Proof. Traversing Δ_{kl} flips the ranking of r_k and r_l in the ordered list, while leaving other values of $\pi_i^{-1}(m)$ and $\pi_j^{-1}(m)$ unchanged for all $m \neq k, l$.

Definition 5 (Local flip). *We call the fact of traversing a bisector Δ_{kl} a* local flip, *($|\pi_i^{-1}(k) - \pi_i^{-1}(l)| = 1$).*

[4] We initially restrict ourselves to $\mathbb{R}^D$ spaces. The generalisation of these notions to generic metric spaces is left for future work.

There is therefore a direct relationship between the geometrical organization of the points and the organization of the ordered list. More generally, neighboring relationships between objects relate to Voronoi diagrams, themselves formed out of bisectors Δ_{kl}. We define the base element of $\mathcal{V}(R)$, the classical Voronoi diagram of R, as follows.

Definition 6 (Voronoi cell). *Given $r_k \in R$, we define $V_R(r_k) \subset \mathbb{R}^D$ as the* Voronoi cell *of r_k with respect to R. $V_R(r_k)$ is the subset:*

$$V_R(r_k) = \left\{p \in \mathbb{R}^D \text{ such that } d(p, r_k) \leq d(p, r_l) \quad \forall r_l \in R\right\}$$

$V_R(r_k)$ is a D-dimensional simplex bounded by bisectors Δ_{kl}. r_k is then said to be a generator *of $V_R(r_k)$.*
$\mathcal{V}(R) = \{V_R(r_k)\ \forall r_k \in R\}$ is the Voronoi diagram *of R.*

Remark 1. We assume that, considering the randomness of the positions of objects in $\mathcal{U}$ (and therefore in R):

- The Voronoi diagram of R is not degenerate, ie, no more than $D+1$ reference objects lie on the same D-dimensional hypershpere;
- no object o_i lies exactly on the boundary of two or more Voronoi cells.

A number of properties of the Voronoi diagrams help us understanding the structure of PBI. We recall the definition of the Delaunay graph.

Definition 7 (Delaunay graph). *Given R and $\mathcal{V}(R)$, we define $G = (R, E)$ the* Delaunay graph *with vertices $r_k \in R$ and edges E such that:*

$$(r_k, r_l) \in E \text{ if and only if } V_R(r_k) \text{ and } V_R(r_l) \text{ share a common facet.}$$

Definition 6 considers a unique object as generator for each Voronoi cell. Hence, by definition, for all objects $o_i \in V_R(r_k)$, we have $\pi_i^{-1}(k) = 1$.

Consider now r_l and $r_m \in R$, neighbors of r_k in G. Δ_{kl} and Δ_{km} support facets of $V_R(r_k)$. Suppose we extend Δ_{lm} within $V_R(r_k)$. Δ_{lm} separates objects o_i for which $\pi_i^{-1}(l) > \pi_i^{-1}(m)$ from objects o_i for which $\pi_i^{-1}(l) < \pi_i^{-1}(m)$. In particular, because r_l and $r_m \in R$ are neighbors of r_k, one may isolate a portion of $V_R(r_k)$ bounded by Δ_{lm} where, for each object o_i in that region $\pi_i^{-1}((k, l, m)) = (1, 2, 3)$. Repeating that process, leads to the construction of the *ordered order-2 Voronoi diagram*, where the generators of the cells at the ordered pairs of reference objects (Figure 1).

Generalizing this construction, we obtain the *ordered order-k Voronoi diagram* (OOkVD).

Definition 8 (Ordered order-k Voronoi diagram). *Given $R_k = (r_{j_1}, \ldots, r_{j_k})$ an ordered subset of R, we define $V_R^k(R_k) \subset \mathbb{R}^D$ as the* OOkVD cell *of R_k with respect to R. $V_R^k(R_k)$ is such that:*

$$o_i \in V_R^k(R_k) \quad \Leftrightarrow \quad \pi_i^{-1}((j_1, \ldots, j_k)) = [\![1, k]\!]$$

Proposition 2. *The equivalent classes of the δ_R distance ($\mathcal{U}/\equiv$) are cells of the ordered order-$(n-1)$ Voronoi diagram of R: if $o_i \in V_R^k(R_k)$ then $[o_i] = V_R^k(R_k)$.*

Proof. By construction. Knowing that $p \in V_R^k(R_k)$ for all $k < n$ is sufficient to determine the ordered list $\boldsymbol{\pi}_p$.

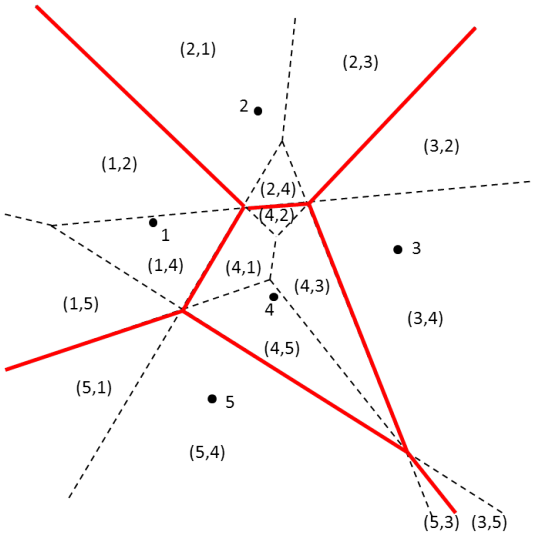

Fig. 1. Original Voronoi diagram (*red bold lines*) for 5 points in the 2D plane. The union with the order-2 Voronoi diagram (*dashed lines*) forms the ordered order-2 Voronoi partition. Cell centers act as reference points. The label for every cell is given as the permutation of the 2 closest reference points from points in the cell. Every original cell is repartitioned by the order-2 neighboring relationships (adapted from [26])

As noted in [2], equivalent classes are the vertices of the permutahedron of order n, the polytope whose edges are connecting all permutations differing from a local flip.

Proposition 3. *The edges of the order-n permutahedron form a equivalent "order-*$(n-1)$ *Delaunay graph" for the ordered order-*$(n-1)$ *Voronoi diagram. In other words, permutations differing from one local flip (connected vertices of the order-n permutahedron) relate to neighboring cells of the ordered order-*$(n-1)$ *Voronoi diagram.*

Proof. Direct from Definition 5 and Proposition 2.

Propositions 2 and 3 provide us with powerful geometric tools to study the performance of the δ_R distance and therefore the permutation-based encoding. For example, it is easily seen that local flips between positions k and $k+1$ in the list relate to crossing edges of the order-k Voronoi diagram. Similarly, relationships between Voronoi cells, Delaunay simplices and enclosing spheres help us understanding which of the $n!$ possible permutations will actually exist in the permutation-based encoding defined by a given choice of R. Upperbounds and D-dimensional constructs that achieve these bounds are presented in [30,31]. An empirical study on the number of Pivot Permutations prefixes is proposed in [25].

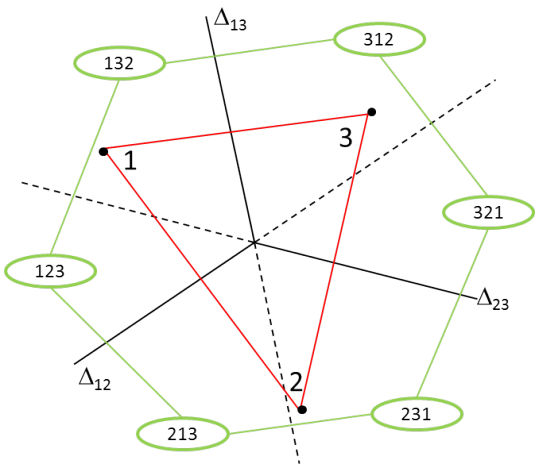

Fig. 2. Ordered order-2 Voronoi diagram of 3 points (*black lines*). Edges of the Delaunay graph (*red lines*). Edges of the order-n permutahedron mapped on the same plane (*green lines*)

3.3 Invariance and robustness

In this paper, we wish to investigate empirically the factors that emerged from the above modeling. Namely, we wish to obtain an empirical understanding of the properties of invariance and robustness of the scheme against perturbations. The related literature focused on the capabilities of the encoding to retrieve all and only the k-NN of a query point p. This provides insight on how much balls centered on p grow similarly according to increasing distances d_E and δ_R, which we use as prototypical metric in the original and permutations spaces, respectively.

Here, we rather aim at going to a finer understanding by giving insights on the questions:

- How much unique is the correspondence between the values of d_E and δ?
- How much position information does each reference point r_k carry in the encoding of object o_i?

We think that such information will advance the understanding of the limitations of PBI and help formally optimizing its parameters such as the number and position of reference points, and whether using partial ordered lists is useful.

4 Experiments

We base our experiments on dense sets of objects drawn uniformly from the unit $\mathbb{R}^D$ cube. We chose n reference points according to the greedy global locality approach [18].

4.1 Original versus permutation-based distances

We first investigate the match between distance values in the original space and the permutation space. Ideally, for every original distance value, we should find a corresponding permutation-based distance value. However, due to rank approximation and invariance, this is not the case. To measure this invariance, for every value of the permutation-based distance[5], we gather the corresponding histogram of the original distance values. The less peaked the histogram, the more invariance, and the more confusion in discriminating objects.

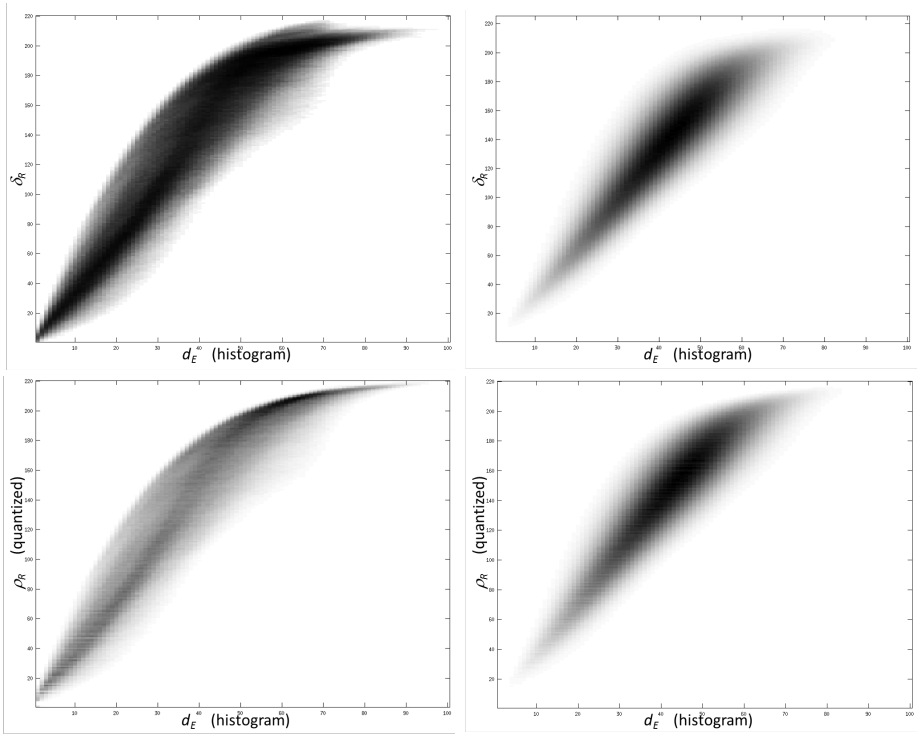

Fig. 3. Collection of histograms (horizontal lines of the images - the darker the higher the value) of Euclidean distance values for every value of the permutation-based distance (vertical value). From left to right, top to bottom: (a) Uniform distribution of 2D objects with δ_R based on 30 reference points. (b) Uniform distribution of 4D objects with δ_R based on 30 reference points. (c) Uniform distribution of 2D objects with ρ_R (values quantized) based on 30 reference points. (d) Uniform distribution of 4D objects with ρ_R (values quantized) based on 30 reference points.

[5] We use $\delta_R(o_i, o_j) = \frac{1}{2}\sum_{k=1}^{n} \left|\pi_i^{-1}(k) - \pi_j^{-1}(k)\right|$, to avoid systematically empty odd bins.

As can be seen from Figure 3, both original and permutation-based distance functions show a decent correlation (dark diagonal corresponding to the peak value of the histograms). δ_R and ρ_R behave similarly. However, the higher the value of the permutation-based distance, the more spread the original corresponding distance values are. This can be interpreted as the fact that the ball of the permutation distance will grow more and more with irregular borders. In other words, there is more and more uncertainty in the match between original distance values and permutation-based distance values.

4.2 Local invariance properties

We now wish to get a more detailed understanding of how permutation-based distance work. From their definition, these distance functions (eg δ_R or ρ_R) essentially count the discrepancy between the ordered list, without accounting for the position in the lists at which this difference arises. For example, if $\delta_R(o_i, o_j) = 1$, the corresponding ordered lists differs from only one local flip. However this local flip may indifferently be between elements at the beginning of the list (eg changing cell of the order-1 Voronoi diagramà) or at the end of the list (crossing Δ_{kl} where r_k and r_l are far from o_i and o_j).

Definition 9 (Activation). *We say that a reference object r_k is* activated *in the computation of $\delta_R(o_i, o_j)$ if $\pi_i^{-1}(k) \neq \pi_j^{-1}(k)$*

Ideally, we would like the position of an object be encoded mostly by its local reference objects. This corresponds to making the position encoding independent of far structures. As a result, this would support the use of local criteria for the choice of reference objects.

In that case, when computing permutation-based distance values for neighboring objects, local reference objects would be activated. Conversely, low values of permutation-based distance should be due to the activation of local reference objects. This would for example justify formally that ordered list pruning is a sound operation.

We plot in Figure 4 the statistics of activation of reference objects ($n = 30$) for each value of the permutation-based distance.

We read a rather uniform distribution of activation, which counters to the idea of local encoding. This may be understood by looking at Figure 1. One can see that bisectors resulting from the high order Voronoi partition splits cells into a fine grain partition. Hence, pairs of distance reference objects do participate in the determination of the fine sensitivity of the encoding. This is rather undesirable and motivates the use of weighted permutation-based distance functions such as that proposed in [15] to enforce a local penalty on distance measurements.

4.3 Real data

We now study a real use case where indexing invariance is desirable and should be adapted to the data. We study Maya hieroglyphs images. A part of Maya

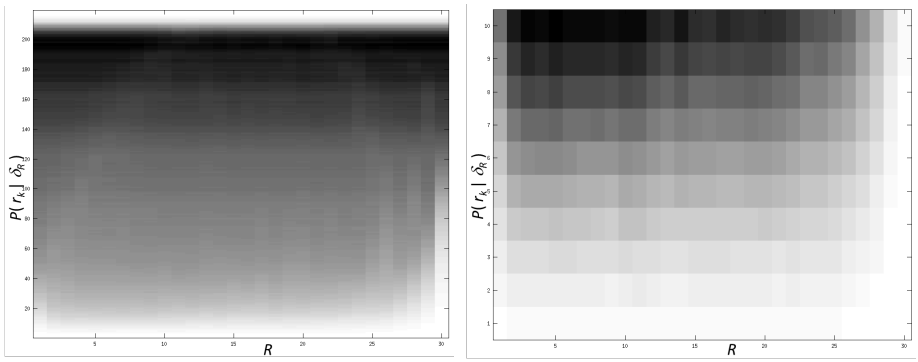

Fig. 4. Collection of histograms (horizontal lines of the images - the darker the higher the value) for the activation (see text) of each reference object depending of the value of δ_R ($D = 2$,$n = 30$). (left) full statistic. (right) zoom on low values of δ_R.

writing consists into *glyphs* (base signs) combined into *glyph blocks*. Each glyph can be referenced by a Thompson code (T-code, eg T0168) and glyphblocks can be therefore described by the combination of the T-codes of the glyphs that compose the block, which we call a T-string (see Figure 5).

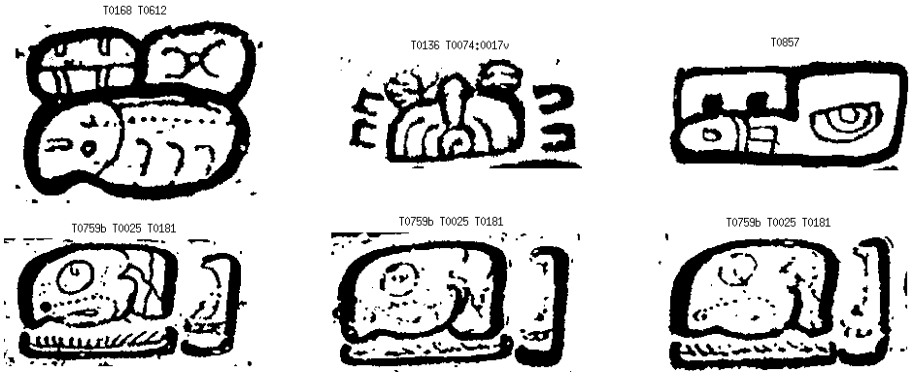

Fig. 5. Maya glyphblocks annotated with their corresponding T-strings. (1st row) Glyphblocks with different number of composing glyphs. (2nd row) Glyphblocks of the same class, illustrating the visual variability of the classes

It is interesting to study visual similarity of Maya hieroglyphs [28]. Figure 5 (2nd row) motivates the fact that the indexing scheme should absorb the visual variation of the symbols. Integrating this with our earlier discussion, the question is how to map similarity-based classification onto the notion of invariance of

the indexing scheme. Here, "invariance" is understood as "invariance to writing style". In other words the challenge is to tune the parameters of the indexing scheme to align with the semantics of the data.

Here, we use a set of 15'500 annotated glyphblocks in 155 classes (same Tstring) of 100 individuals. We extract features from an autoencoder. We use the values of the L most activated neurons on the encoding layer (joining the encoder and the decoder architectures). Our initial experiments show that this encoding captures relevant visual features [6]. We extract the $L = 30$ values of the most activated neurons of the encoding layer as features and use the Euclidean distance to measure similarity. Here, we adapt the number n of reference objects using the greedy global locality approach.

Table 1. Average equivalence class population and precision with respect to the number of reference objects. Values in brackets indicate the standard deviation

n	6	7	8	9	10	20
Pop.	233.56 (187.76)	99.31 (99.14)	43.77 (57.46)	19.74 (35.58)	7.20 (11.87)	1.06 (1.44)
Prec.	0.05 (0.09)	0.10 (0.17)	0.19 (0.25)	0.36 (0.34)	0.54 (0.37)	99.84 (0.03)

The above numbers in Table 1 illustrate the reduction in size of the partition cells, leading to a reduction of the size of their population. In that particular application, standard deviation figure on the cell population show that the choice of reference objects is not adapted to the data since there is a large variation in the number of items per cell. A higher number of reference objects creates a finer partition. As a result, the precision inside the equivalence class mechanically increases. However, here again, the figures show the need for an adapted choice of reference objects to align the equivalence classes (cells of the partition) with the semantic classes of the data. It is therefore a critical challenge to formulate the optimisation of the choice of reference objects according to the semantic value of the data.

5 Conclusion

Permutation-based indexing schemes have shown to be effective to support the resolution of kNN queries. Their main parameters are the number and location of reference objects and the permutation-based distance used.

In this paper, the main contribution is a formal modeling of the mechanics of PBI schemes, helped by ranking theory and computational geometry. This base model provides insights and powerful tools for the fine study of properties of permutation-based geometry. Here, we focus on invariance, which relates to robustness to data variation (eg due to noise). We motivate such a study by the use of PBI in applications where items may be grouped by classes with internal

[6] The details of this study may be found in [27]

variation. In that case, kNN queries may be resolved directly using the space partition thus created.

Our initial experiments following our model reveal an adequate transfer of neighboring information from the original feature space onto the permutation-based representation space. However, the analysis also demonstrates that permutation-based distances such as δ_R or ρ_R do not localize the measurements, as it would be desirable. The use of adapted permutation-based distance functions (such as weighted by rank position [15]) may be beneficial here.

This paper opens many avenues for deeper studies on PBI. We plan to extend our formal model in the direction of a better understanding of the geometry of PBI and the design or choice of adapted parameters such as permutation-based distance incorporating pruning or weighting. Getting deeper insights on the geometry of the partition will also be a way to optimize the use of reference objects and therefore their location and number.

6 Acknowledgments

This work has been partly supported by the Swiss National Science Foundation under project MAAYA (SNF Grant number 144238).
Dr Hisham Mohamed is now with Sensirion AG, Staefa, Switzerland.

References

1. Amato, G., Esuli, A., Falchi, F.: Pivot selection strategies for permutation-based similarity search. In: Brisaboa, N., Pedreira, O., Zezula, P. (eds.) Similarity Search and Applications, Lecture Notes in Computer Science, vol. 8199, pp. 91–102. Springer Berlin Heidelberg (2013)
2. Amato, G., Falchi, F., Rabitti, F., Vadicamo, L.: Some theoretical and experimental observations on permutation spaces and similarity search. In: Traina, A.J.M., Traina, Caetano, J., Cordeiro, R.L.F. (eds.) Similarity Search and Applications, Lecture Notes in Computer Science, vol. 8821, pp. 37–49. Springer International Publishing (2014)
3. Amato, G., Rabitti, F., Savino, P., Zezula, P.: Region proximity in metric spaces and its use for approximate similarity search. ACM Trans. Inf. Syst. 21(2), 192–227 (2003)
4. Ares, L.G., Brisaboa, N.R., Esteller, M.F., Pedreira, O., Places, A.S.: Optimal pivots to minimize the index size for metric access methods. In: Proceedings of the 2009 Second International Workshop on Similarity Search and Applications. pp. 74–80. SISAP '09, IEEE Computer Society, Washington, DC, USA (2009)
5. Beyer, K., Goldstein, J., Ramakrishnan, R., Shaft, U.: When is "nearest neighbor" meaningful? In: In Int. Conf. on Database Theory. pp. 217–235 (1999)
6. Burkhard, W.A., Keller, R.M.: Some approaches to best-match file searching. Commun. ACM 16(4), 230–236 (Apr 1973)
7. Bustos, B., Navarro, G., Chávez, E.: Pivot selection techniques for proximity searching in metric spaces. Pattern Recognition Letters 24(14), 2357–2366 (2003)

8. chavez, E., Figueroa, K., Navarro, G.: Effective proximity retrieval by ordering permutations. Pattern Analysis and Machine Intelligence, IEEE Transactions on 30(9), 1647–1658 (Sept 2008)
9. Chávez, E., Marroquín, J.L., Navarro, G.: Fixed queries array: A fast and economical data structure for proximity searching. Multimedia Tools Appl. 14(2), 113–135 (Jun 2001)
10. Chávez, E., Navarro, G., Baeza-Yates, R.A., Marroquín, J.L.: Searching in metric spaces. ACM Computer Surveys 33(3), 273–321 (2001)
11. Ciaccia, P., Patella, M., Zezula, P.: M-tree: An efficient access method for similarity search in metric spaces. In: Proceedings of the 23rd International Conference on Very Large Data Bases. pp. 426–435. VLDB '97, San Francisco, CA, USA (1997)
12. Garcia, V., Debreuve, E., Nielsen, F., Barlaud, M.: K-nearest neighbor search: Fast gpu-based implementations and application to high-dimensional feature matching. In: Image Processing (ICIP), 2010 17th IEEE International Conference on. pp. 3757–3760. IEEE (2010)
13. Hinneburg, A., Aggarwal, C.C., Keim, D.A.: What is the nearest neighbor in high dimensional spaces? In: Proceedings of the 26th International Conference on Very Large Data Bases. pp. 506–515. VLDB '00, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2000)
14. Krulis, M., Osipyan, H., Marchand-Maillet, S.: Optimizing sorting and top-k selection steps in permutation based indexing on gpus. In: New Trends in Databases and Information Systems - ADBIS 2015 Short Papers and Workshops, Poitiers, France, September 8-11, 2015. Proceedings. pp. 305–317 (2015)
15. Kumar, R., Vassilvitskii, S.: Generalized distances between rankings. In: Proceedings of the 19th International Conference on World Wide Web. pp. 571–580. WWW '10, New York, NY, USA (2010)
16. Lebanon, G., Lafferty, J.D.: Cranking: Combining rankings using conditional probability models on permutations. In: Proceedings of the Nineteenth International Conference on Machine Learning. pp. 363–370. ICML '02, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2002)
17. Li, S., Amenta, N.: Brute-force k-nearest neighbors search on the GPU. In: 8th International Conference on Similarity Search and Applications, SISAP 2015. vol. LNCS 9371. Glasgow, UK (2015)
18. Mohamed, H.: Scalable approximate k-NN in multidimensional Big Data. Ph.D. thesis, Viper group, CS Department, University of Geneva (Aug 2014), (in particular, Chapter 3)
19. Mohamed, H., Marchand-Maillet, S.: Distributed media indexing based on MPI and mapreduce. Multimedia Tools and Applications 69(2) (2014)
20. Mohamed, H., Marchand-Maillet, S.: Quantized ranking for permutation-based indexing. Information Systems (2015)
21. Mohamed, H., Osipyan, H., Marchand-Maillet, S.: Multi-core (CPU and GPU) for permutation-based indexing. In: Proceedings of the 7th Internation Conference on Similarity Search and Applications (SISAP2014). Los Cabos, Mexico (2014)
22. Mohammed, H., Marchand-Maillet, S.: Big Data: Algorithms, Analytics, and Applications, chap. Scalable Indexing for Big Data Processing. Chapman & Hall (2015)
23. Nielsen, F., Piro, P., Barlaud, M.: Bregman vantage point trees for efficient nearest neighbor queries. In: Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on. pp. 878–881. IEEE (2009)
24. Novak, D., Batko, M., Zezula, P.: Metric index: An efficient and scalable solution for precise and approximate similarity search. Inf. Syst. 36(4), 721–733 (2011)

25. Novak, D., Zezula, P.: Performance study of independent anchor spaces for similarity searching. The Computer Journal 57(11), 1741–1755 (2014)
26. Okabe, A., Boots, B., Sugihara, K., Chui, S.N.: Spatial Tessellations: Concepts and Applications of Voronoi Diagrams. John Wiley & Sons, Inc., New York, NY, USA, 2nd edn. (2000)
27. Roman-Rangel, E., Marchand-Maillet, S.: Indexing Mayan hieroglyphs with neural codes. In: International Conference on Pattern Recognition (ICPR'16). Cancun, Mexico (2016)
28. Roman-Rangel, E., Wang, C., Marchand-Maillet, S.: Simmap: Similarity maps for scale invariant local shape descriptors. Neurocomputing 175, Part B, 888 – 898 (2016)
29. Samet, H.: Foundations of multidimensional and metric data structures. The Morgan Kaufmann series in computer graphics and geometric modeling, Elsevier/Morgan Kaufmann (2006)
30. Skala, M.: Counting distance permutations. In: Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference on. pp. 362–369 (April 2008)
31. Skala, M.: Aspects of Metric Spaces in Computation. Ph.D. thesis, University of Waterloo (2008)
32. Uhlmann, J.K.: Satisfying general proximity / similarity queries with metric trees. Information Processing Letters 40(4), 175 – 179 (1991)
33. Volnyansky, I., Pestov, V.: Curse of dimensionality in pivot based indexes. In: Similarity Search and Applications, 2009. SISAP '09. Second International Workshop on. pp. 39–46 (Aug 2009)
34. Weber, R., Schek, H.J., Blott, S.: A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In: Proceedings of the 24rd International Conference on Very Large Data Bases. pp. 194–205. VLDB '98, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
35. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. J. Mach. Learn. Res. 10, 207–244 (2009)
36. Yianilos, P.N.: Data structures and algorithms for nearest neighbor search in general metric spaces. In: Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 311–321. SODA '93, Philadelphia, PA, USA (1993)
37. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity Search: The Metric Space Approach, Advances in Database Systems, vol. 32. Springer (2006)