ICASSP 2018 - Streaming Binary Sketching based on Subspace Tracking and Diagonal Uniformization

Anne Morvan[1,2], Antoine Souloumiac[1], Cédric Gouy-Pailler[1] and Jamal Atif[2]

[1]CEA, LIST, [2]Université Paris-Dauphine, PSL Research University, CNRS, UMR 7243, LAMSADE

Objectives

We present a new online method for computing distance-preserving compact c-bits codes -*sketches*- of high-dimensional data stream to perform efficient similarity search. Particularities of the method:

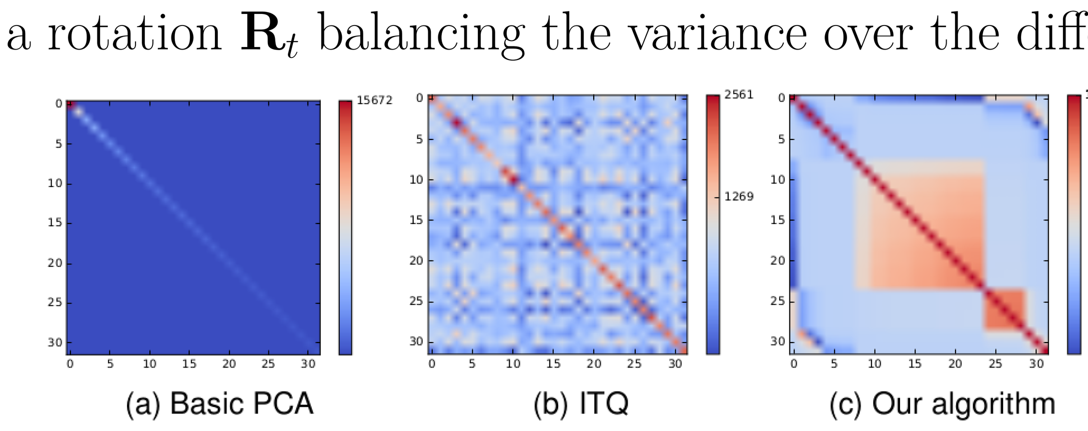
- fully adaptable to the streaming setting
- low space complexity: $O(c^2)$
- low time complexity: $O(c^2)$ per code
- convergence guarantees

Principle

- **Input:** High-dimensional streaming data $\{\mathbf{x}_t \in \mathbb{R}^d\}_{1\leq t\leq n}$
- **Goal:** find the good projection onto a lower dimensional space, i.e. define $\tilde{\mathbf{W}}_t \in \mathbb{R}^{c\times d}$ s.t. $c \ll d$ and the c-bits binary code $\mathbf{b}_t = \mathrm{sign}(\tilde{\mathbf{W}}_t\mathbf{x}_t)$
- **Proposed model (inspired by ITQ [1]):** $\tilde{\mathbf{W}}_t = \mathbf{R}_t\mathbf{W}_t$ with $\mathbf{W}_t \in \mathbb{R}^{c\times d}$ principal subspace, $\mathbf{R}_t\mathbf{R}_t^T = \mathbf{R}_t^T\mathbf{R}_t = \mathbf{I}_c$

Key challenges

- $\mathbf{W}_t$: how to estimate online the eigen subspace?
- Importance of $\mathbf{R}_t$: without, variance concentrated on the first dimensions
- How to define a rotation $\mathbf{R}_t$ balancing the variance over the different directions?



Givens rotation and notations

$$\mathbf{G}(i,j,\theta) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & -s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix}$$

where $i > j$, $c = \cos(\theta)$ and $s = \sin(\theta)$; $\forall k \neq i,\ j,\ g_{k,k} = 1$; $g_{i,i} = g_{j,j} = c$, $g_{j,i} = -s$ and $g_{i,j} = s$. All remaining coefficients are set to 0.

For $x \in \mathbb{R}$, $\mathrm{sign}(x) = 1$ if $x \geq 0$ and -1 otherwise. On vectors, applied component-wise. For any matrix $\mathbf{M}$, $\mathbf{\Sigma_M} = \mathbf{M}\mathbf{M}^T$. $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$; $\mathbf{V} = \mathbf{W}\mathbf{X} \in \mathbb{R}^{c\times n}$; $\mathbf{Y} = \mathbf{R}\mathbf{V} \in \mathbb{R}^{c\times n}$

UnifDiag Model

- **Role of R:** balancing variance over the c directions
- **Equivalence:** equalizing the diagonal coefficients of $\mathbf{\Sigma_Y}$ to the same value
$$\tau = \mathrm{Tr}(\mathbf{\Sigma_V})/c$$
How to proceed? In the sequel, the subscript t is dropped for readability.
- $\mathbf{\Sigma_V}$ dynamically computed as new data is seen during update of W with OPAST [2]
- $\mathbf{R}$ defined as a product of $c-1$ Givens rotations $\{\mathbf{G}(i_r, j_r, \theta_r)\}_{1\leq r\leq c-1}$ iteratively applied left and right to $\mathbf{\Sigma_V}$:
- For $r \in \{1, \ldots, c-1\}$, given i_r, j_r, θ_r,
$$(\mathbf{\Sigma_Y})_r \leftarrow \mathbf{G}(i_r, j_r, \theta_r)\ (\mathbf{\Sigma_Y})_{r-1}\ \mathbf{G}(i_r, j_r, \theta_r)^T$$
$$\mathbf{R}_r \leftarrow \mathbf{R}_{r-1}\ \mathbf{G}(i_r, j_r, \theta_r)^T,$$
where $(\mathbf{\Sigma_Y})_0 = \mathbf{\Sigma_V}$, $\mathbf{R}_0 = \mathbf{I}_c$.
- At each step r, i_r and j_r are chosen to be the indices of the current smallest and largest diagonal coefficients of $(\mathbf{\Sigma_Y})_{r-1}$.
- θ_r is computed accordingly (cf. Th.3.1 in the paper)
- **Result:** At the end of step r, r diagonal coefficients of $(\mathbf{\Sigma_Y})_r$ are equal to τ.

Experimental results

- Datasets CIFAR-10 and GIST1M: 60000 960-D GIST descriptors each
- Quality of hashing assessed on the nearest neighbor (NN) search task with the Mean Average Precision (mAP): 1000 queries randomly sampled and the remaining data as training set
- Euclidean ground truth built with a nominal threshold of the average distance to the 50th nearest neighbor
- Comparison with 3 online baselines with hashing scheme $\Phi(\mathbf{x}_t) = \mathrm{sign}(\tilde{\mathbf{W}}_t\mathbf{x}_t)$:
  1. **OSH** [3].
  2. **RandRot-OPAST**: $\mathbf{W}_t$ obtained with OPAST, $\mathbf{R}_t$ a constant random rotation
  3. **IsoHash-OPAST** [4]: $\mathbf{R}_t$ obtained with IsoHash
  4. **UnifDiag-OPAST**: $\mathbf{R}_t$ obtained with UnifDiag
- Code on GitHub: `annemorvan/UnifDiagStreamBinSketching/`

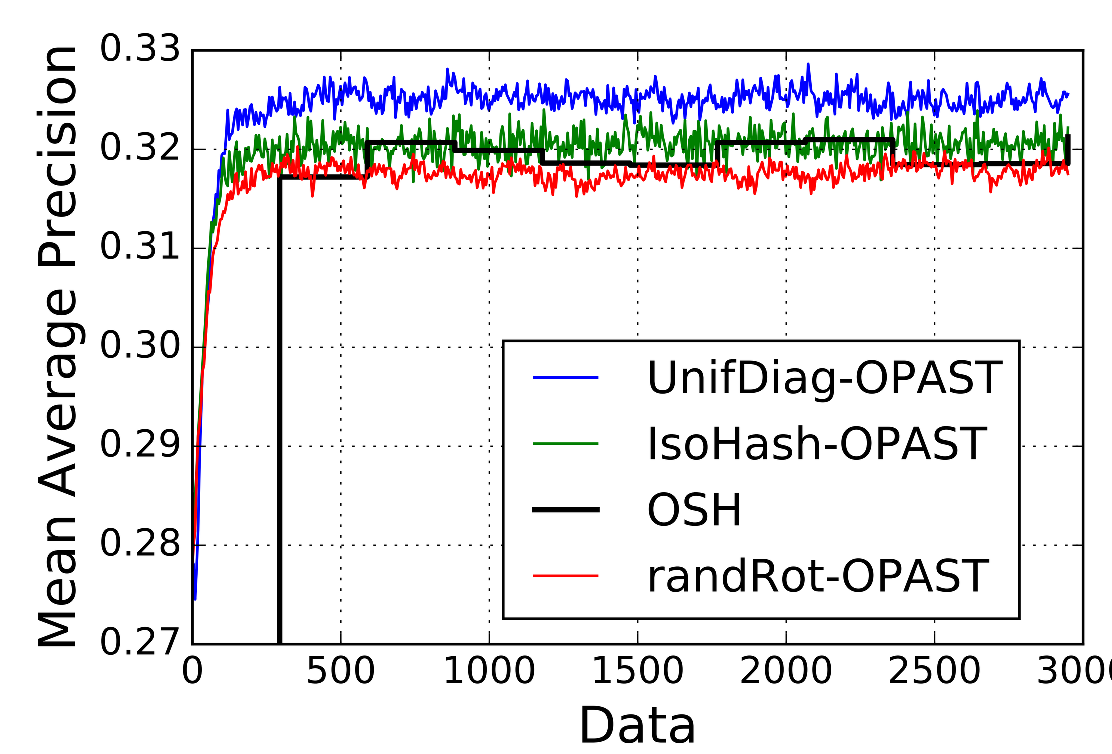


Figure 1: mAP for $c = 32$ on CIFAR-10 (avg. over 5 training/test partitions)

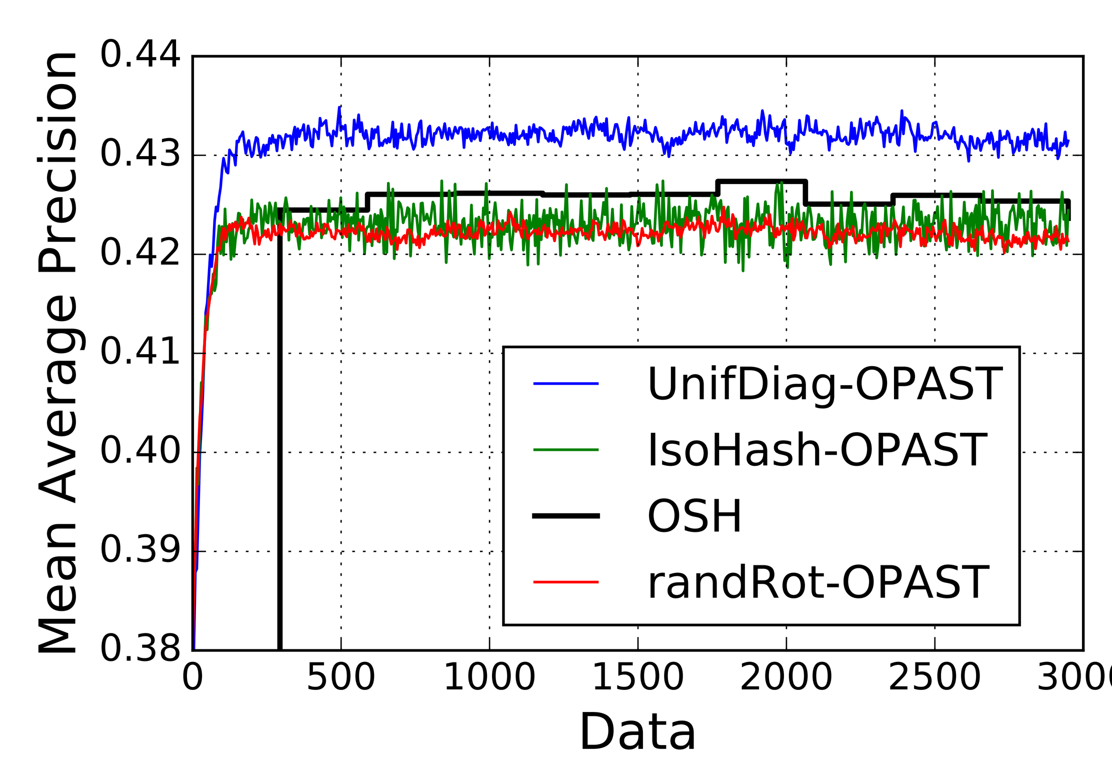


Figure 2: mAP for $c = 32$ on GIST-1M (avg. over 5 training/test partitions)

Conclusion and perspectives

- We introduced a novel method for learning distance-preserving binary embeddings of high-dimensional data streams with convergence guarantees.
- Our algorithm does not need to store the whole dataset.
- Binary codes can be obtained without delay as a new data point is seen.
- Our approach achieves better accuracy than state-of-the-art online unsupervised methods.
- We showed how Givens rotations can be used for uniformizing the diagonal of a symmetric matrix.
- Could another rotation be more optimal?

To whom correspondence should be adressed: anne.morvan@cea.fr. Anne is partly supported by the *Direction Générale de l'Armement* (French Ministry of Defense).

[1] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin.
Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval.
IEEE Transactions on Pattern Analysis and Machine Intelligence, (12):2916–2929, 2013.

[2] K. Abed-Meraim, A. Chkeif, and Y. Hua.
Fast orthonormal past algorithm.
IEEE Signal Processing Letters, (3):60 – 62, 2000.

[3] C. Leng, J. Wu, J.and Cheng, X. Bai, and H. Lu.
Online sketching hashing.
In *CVPR*, pages 2503–2511, 2015.

[4] W. Kong and W. Li.
Isotropic hashing.
In *NIPS*, pages 1646–1654. 2012.