

# APPROXIMATE NEAREST NEIGHBORS USING SPARSE REPRESENTATIONS

Joaquin Zepeda

Ewa Kijak

Christine Guillemot

INRIA Centre Rennes -  
Bretagne Atlantique  
Rennes, France

Université de Rennes 1  
IRISA  
Rennes, France

INRIA, Centre Rennes -  
Bretagne Atlantique  
Rennes, France

## ABSTRACT

A new method is introduced that makes use of sparse image representations to search for approximate nearest neighbors (ANN) under the normalized inner-product distance. The approach relies on the construction of a new sparse vector designed to approximate the normalized inner-product between underlying signal vectors. The resulting ANN search algorithm shows significant improvement compared to querying with the original sparse vectors. The system makes use of a proposed transform that succeeds in uniformly distributing the input dataset on the unit sphere while preserving relative angular distances.

**Index Terms**— Sparse representations, indexing, data conditioning.

## 1. INTRODUCTION

Local descriptors computed on affine normalized image regions have proven successful in computer vision applications requiring image matching and recognition [1]. The selected regions are affine normalized to be invariant under common transformations such as those resulting from camera perspective or illumination changes. Different descriptors  $\underline{y}$  have been developed to describe the resulting normalized image regions. At query time, a nearest neighbors (NN) search is carried out between the query descriptor  $\underline{y}_q$  and the descriptor vectors  $\underline{y}_b$  computed on the database images. Yet since local descriptors are high-dimensional, they are subject to the *curse of dimensionality* [1], meaning that the NN search complexity is very high.

Approximate nearest neighbor (ANN) searches based on various sparse representation schemes have been recently proposed to address the high computational complexity in local descriptor query systems [2, 3, 4]. Given some sparse representation  $\underline{x}_b$  of each  $\underline{y}_b$ , the search index will be the sparse matrix with columns  $\underline{x}_b$ . This *sparse matrix index* is stored in compact row-major format by grouping all non-zero coefficients of any given row (and their column indices) to form a contiguous memory bin. This results in an implicit complexity-reducing pruning mechanism when using similarity measures based on the inner-product  $\underline{x}_q^t \underline{x}_b$ , as only the bins corresponding to the non-zero positions of  $\underline{x}_q$  need to be processed.

The work carried out in the present paper uses sparse vectors  $\underline{x}$  that are a *dictionary-based sparse representation* (DBSR) of the corresponding  $\underline{y}$ : Given an overcomplete matrix  $\mathbf{D}$  (the dictionary),  $\underline{x}$  will satisfy  $\underline{y} = \mathbf{D}\underline{x} + \underline{r}$ . The chosen  $\underline{x}$  produces an approximation  $\mathbf{D}\underline{x}$  minimizing the distortion  $|\underline{r}|$  under a constraint on the number of non-zero positions of  $\underline{x}$  (a measure of rate). Using the sparse vectors  $\underline{x}$  built following a rate-distortion criterion raises a new problem: The residual transformations following the geometrical region normalization result in descriptors  $\underline{y}$  that have DBSR  $\underline{x}$  with unstable support (positions of non-zero coefficients). This instability can

severely impact the similarity score between regions and therefore the ranking performance of the ANN search task.

In this paper, we address the problem of instabilities in the support of DBSRs  $\underline{x}$  by first modelling the instabilities and then constructing, from  $\underline{x}$ , a new vector which we call a *reduced vector*. The construction of the *reduced vector* is formulated as a minimization of the reference distance approximation error subject to a sparsity constraint as, for sparse matrix indices, sparsity is related to both memory and computational complexity. Computational complexity is further maximized for a uniform distribution of non-zero positions in  $\underline{x}$ , which is in turn favored by a uniform distribution of  $\underline{y}$  on the unit sphere. Thus we further introduce a data conditioning method which succeeds in approximately preserving the relative position of data points  $\underline{y}$  on the unit sphere while making their distribution more uniform.

One potential application of our proposed method is that of enhancing the performance/complexity tradeoff of bag-of-features (BOF) indices [3]. BOFs make use of vector quantization (VQ), which is a specific case of the more general DBSR framework. Thus VQ again uses a rate-distortion criterion that does not favor a stable support of the resulting  $\underline{x}$ , an issue addressed by our proposed method.

A second potential application involves using  $\underline{x}$  to design a low rate image/local descriptors package for the case when the normalized image regions are used directly as descriptors  $\underline{y}$ . Transmitting the  $\underline{x}$  thus obtained yields an initial image estimate at no extra rate penalty. Since the receiver requires  $\underline{x}$  rather than  $\underline{y}$  for querying or indexing, including  $\underline{x}$  in the transmitted package further exempts the receiver from descriptor extraction and processing.

The rest of this paper is organized as follows: we present the proposed reduced vector construction strategy in section 2. Our algorithm makes use of an adaptive sparse correlation matrix, and we explain how to obtain it in section 3. Our proposed data conditioning method is then presented in section 4, and evaluated along with our main approach in section 5. We provide concluding remarks in the last section.

## 2. FORMULATING SPARSE SUPPORT SELECTION AS AN OPTIMIZATION PROBLEM

We now explain the method used to build the proposed *reduced vectors* from the sparse representations  $\underline{x}$ . *Reduced vectors* enjoy a more stable support relative to  $\underline{x}$  and are thus better suited for ANN searches based on sparse matrix indices.

Assuming all  $\underline{y} = \mathbf{D}\underline{x} + \underline{r}$  to be normalized and compressible (i.e., with negligible  $\underline{r}$ ), we first expand  $\langle \underline{y}_q, \underline{y}_b \rangle$  as  $\underline{x}_q^t \mathbf{C}_D \underline{x}_b = \underline{\chi}_q^t \underline{x}_b$ , where  $\mathbf{C}_D = \mathbf{D}^t \mathbf{D}$  and  $\underline{\chi}_q = \mathbf{C}_D \underline{x}_q$ ; the operator  $\langle \cdot, \cdot \rangle$  denotes correlation (normalized inner-product). Since computational

and memory complexity are related to sparsity, we will build the *reduced vector*  $\hat{\underline{x}}_q$  by retaining only  $l_0 = |\underline{x}_q|_0$  coefficients from  $\underline{x}_q$  at positions  $p \in \mathcal{P}$ ;  $\hat{\underline{x}}_q$  will be zero elsewhere. We choose the  $l_0$  positions that minimize the distance approximation error:

$$\begin{aligned} \mathcal{P} &= \underset{\mathcal{P}'}{\operatorname{argmin}} |(\underline{x}_q - \hat{\underline{x}}_q)^t \underline{x}_b|^2 \text{ s.t. } |\mathcal{P}'| = l_0 \iff \\ &\mathcal{P} = \underset{\mathcal{P}'}{\operatorname{argmin}} |\hat{\underline{x}}_q^t \underline{x}_b|^2 \text{ s.t. } |\mathcal{P}'| = l_0, \end{aligned} \quad (1)$$

where  $\hat{\underline{x}}_q = \underline{x}_q - \hat{\underline{x}}_q$  is the *complementary reduced vector*.

Similarly to  $\underline{x}_q = \mathbf{C}_D \underline{x}_q$ , we can write  $\hat{\underline{x}}_q = \hat{\mathbf{C}}_D \underline{x}_q$  by defining  $\hat{\mathbf{C}}_D$  with rows equal to  $\mathbf{C}_D$  at row indices  $p \in \mathcal{P}$  and zeros elsewhere. Using this, the approximate distance  $\hat{\underline{x}}_q^t \underline{x}_b = \underline{x}_q^t \hat{\mathbf{C}}_D \underline{x}_b$  is seen to have a general form  $\underline{x}_q^t \hat{\mathbf{C}}_D \underline{x}_b$  related to the reference system  $\underline{x}_q^t \underline{x}_b$ : both can be generalized by  $\underline{x}_q^t \mathbf{C}'_D \underline{x}_b$ , where  $\underline{x}_q^t \underline{x}_b$  neglects the cross-atom correlations with  $\mathbf{C}'_D = \mathbf{I}$ . Thus we expect and indeed observe in the results section that the proposed distances consistently outperform  $\underline{x}_q^t \underline{x}_b$ .

### 2.1. Exact solution for certain $\underline{x}_b$

In searching a solution to (1) we first note that, using  $\mathcal{P}$  equal to the sparse support of  $\underline{x}_q$ , the approximation  $\hat{\underline{x}}_q^t \underline{x}_b$  will in fact equal  $\langle \underline{y}_q, \underline{y}_b \rangle$  for all  $\underline{y}_b = \mathbf{D} \underline{x}_b$  having  $\underline{x}_b$  with support contained in  $\mathcal{P}$ . Using subscript  $\langle \cdot \rangle$  to indicate vector components, we express the resulting  $\mathcal{P}$  and corresponding *reduced vector*  $\hat{\underline{x}}_q^o$  as

$$\mathcal{P} = \left\{ p : \underline{x}_{q(p)} \neq 0 \right\} \rightarrow \hat{\underline{x}}_q^o. \quad (2)$$

### 2.2. Minimizing upper bound

The  $\mathcal{P}$  given by (2) disregards the original problem formulation (1). We present now a first solution to (1) that consists of substituting the cost function  $|\hat{\underline{x}}_q^t \underline{x}_b|^2$  by the upper bound  $|\hat{\underline{x}}_q|^2 |\underline{x}_b|^2$ . Dropping the term  $|\underline{x}_b|^2$  constant with  $\mathcal{P}$ , we write the resulting cost function as  $|\hat{\underline{x}}_q|^2 = \sum_{k \notin \mathcal{P}} |\underline{x}_{q(k)}|^2$ , exposing the solution  $\mathcal{P}$  as that excluding the  $|\mathcal{P}| = l_0$  strongest terms  $|\underline{x}_{q(p)}|^2$  from the sum of squares:

$$\mathcal{P} = \left\{ p : \forall k \notin \mathcal{P}, |\underline{x}_{q(p)}|^2 \geq |\underline{x}_{q(k)}|^2 \right\} \rightarrow \hat{\underline{x}}_q^u. \quad (3)$$

### 2.3. Probabilistic approach

The solutions obtained in (2) and (3) disregard the index vectors  $\underline{x}_b$  appearing in the original problem (1). We would prefer to build *reduced vectors*  $\hat{\underline{x}}_q$  better suited to finding nearest neighbors from within a particular index set  $\{\underline{x}_b\}$ . To this end we treat  $\underline{x}_b$  as a random vector with realizations corresponding to the index set  $\{\underline{x}_b\}$ . Taking the expectation over  $\underline{x}_b$  of the target function (1) produces

$$\underset{\mathcal{P}'}{\operatorname{argmin}} \hat{\underline{x}}_q^t \mathbf{C}_B \hat{\underline{x}}_q \text{ s.t. } |\mathcal{P}'| = l_0, \quad (4)$$

with  $\mathbf{C}_B = \mathbb{E} [\underline{x}_b \underline{x}_b^t]$ . Unlike (2) and (3), a solution to (4) is not straightforward and could in general require a combinatorial approach wherein all  $\binom{l_0}{N_A}$  possible values for  $\mathcal{P}$  are tried.

#### 2.3.1. Hybrid upper bound using first singular vector

We propose here a simplification of (4) that aims to adapt the simplicity of  $\mathcal{P}$  in (3) while considering the statistics of the index set. To this end we consider substituting the cost function  $|\hat{\underline{x}}_q|^2$  used to obtain (3) by  $|\operatorname{diag}(\underline{v}_1) \hat{\underline{x}}_q|^2$ , where  $\operatorname{diag}(\underline{v}_1)$  is the diagonal matrix with  $\underline{v}_1$  along its main diagonal. The resulting solution thus takes a simple form similar to (3):

$$\mathcal{P} = \left\{ p : \forall k \notin \mathcal{P}, |\underline{v}_{1(p)} \underline{x}_{q(p)}|^2 \geq |\underline{v}_{1(k)} \underline{x}_{q(k)}|^2 \right\} \rightarrow \hat{\underline{x}}_q^1. \quad (5)$$

The vector  $\underline{v}_1$  needs to be a model for vectors  $\underline{x}_b$  that are good responses to  $\underline{x}_q$ . We note in particular that nulls  $\underline{v}_{1(k)}$  will force the corresponding positions  $k$  to be excluded from  $\mathcal{P}$ . We propose here using  $\underline{v}_1$  corresponding to the first singular vector of  $\mathbf{C}_B$ . Later we will show how to build an estimate  $\mathbf{C}_B$  that adapts to  $\underline{x}_q$  and thus yields a  $\underline{v}_1$  that is a good model of correct responses  $\underline{x}_b$ .

#### 2.3.2. Approximate solution

The solution in (5) does not address the formulation (4) directly. Thus, here we present an iterative approximation of (4) that consists in selecting, at iteration  $l$ , the single position  $p_l$  yielding the greatest decrease in magnitude of the cost function when all previous positions  $\mathcal{P}_{l-1} = \{p_1, \dots, p_{l-1}\}$  are nulled out. Letting  $\hat{\underline{x}}_{q\mathcal{P}'}$  denote the *complementary reduced vector* with nulls at positions  $\mathcal{P}'$ , and further letting  $\mathcal{P}_0$  be the empty set, we write this as

$$\mathcal{P} = \left\{ p_l : \forall k \notin \mathcal{P}_{l-1}, \left| \hat{\underline{x}}_{q\{\mathcal{P}_{l-1}, p_l\}}^t \mathbf{C}_B \right|^2 \leq \left| \hat{\underline{x}}_{q\{\mathcal{P}_{l-1}, k\}}^t \mathbf{C}_B \right|^2 \right\} \rightarrow \hat{\underline{x}}_q^h. \quad (6)$$

### 2.4. Query-side versus index-side reduced vectors

Up to now we have considered query-side *reduced vectors*  $\hat{\underline{x}}_q$  built from the query vectors  $\underline{x}_q$  at query time, yet we can also consider building index-side *reduced vectors*  $\hat{\underline{x}}_b$  at index construction time. The two approaches trade query overhead related to the construction of the *reduced vector* for index construction overhead. The index-side vectors  $\hat{\underline{x}}_b$  are built in the same manner as  $\hat{\underline{x}}_q$  by treating each individual  $\underline{x}_b$  in the same manner as the query vector  $\underline{x}_q$  appearing (implicitly) in expressions (2), (3), (5) and (6). Query-side *reduced vectors*  $\hat{\underline{x}}_q$  use a sparse matrix index with columns  $\underline{x}_b$ ; the corresponding approximate distance is  $\hat{\underline{x}}_q^t \underline{x}_b$ . For the case of index-side *reduced vectors*, the  $\hat{\underline{x}}_b$  will instead comprise the columns of the sparse matrix index with  $\underline{x}_q$  as the query vector and resulting approximate distance  $\underline{x}_q^t \hat{\underline{x}}_b$ . For brevity, we focus our discussion on the case of query-side *reduced vectors*  $\hat{\underline{x}}_q$ .

## 3. CONSTRUCTION OF $\mathbf{C}_B$

The sparse correlation matrix  $\mathbf{C}_B$  is needed in (5) and (6) to build the *reduced vector*  $\hat{\underline{x}}_q$  and we now consider its construction. We will use an unbiased estimate  $\frac{1}{|\mathcal{C}|-1} \sum_{\underline{x}_b \in \mathcal{C}} \underline{x}_b \underline{x}_b^t$  of  $\mathbf{C}_B$ , where the realizations  $\underline{x}_b \in \mathcal{C}$  used are chosen as a function of  $\underline{y}_q = \mathbf{D} \underline{x}_q$  in one of two manners specified shortly. This adaptive approach is meant to obtain better distance approximations  $\hat{\underline{x}}_q^t \underline{x}_b$  for stored vectors  $\underline{y}_b = \mathbf{D} \underline{x}_b$  that are more similar to  $\underline{y}_q$ . The estimate of  $\mathbf{C}_B$  thus built can be seen as a model for the instabilities in the sparse support  $\mathcal{P}$  of  $\underline{x}_q$ .

We propose two methods to obtain the realizations  $\underline{x}_b \in \mathcal{C}$ , a two-stage method and an episode method. The two-stage approach consists of first carrying out an initialization query using a sparse matrix index with columns  $\underline{x}_b$  and a query vector  $\hat{\underline{x}}_q^o$  built as in (3) (note that  $\underline{x}_q$  and  $\hat{\underline{x}}_q^u$  could likewise be used as query vectors since neither requires  $\mathbf{C}_B$ ). The first several realizations  $\underline{x}_b$  thus obtained will comprise  $\mathcal{C}$ , and the resulting estimate of  $\mathbf{C}_B$  will be used to build a *reduced vector* using (6) ((5) could likewise be used). The second method of obtaining  $\mathcal{C}$  relies on an assumed prior distribution of the data vectors  $\underline{y}_q$  that can be used to obtain a set of similar vectors  $\underline{y}_q^c$  referred to as the *episode* of  $\underline{y}_q$ . The related sparse decompositions of the episode provide the realizations  $\mathcal{C}$ . An example of a possible assumed prior distribution would be the case where the  $\underline{y}_q$  are noisy measurements; the episode would be obtained by applying multiple realizations of the noise model to  $\underline{y}_q$ .

#### 4. DATA CONDITIONING

We now present a data conditioning method that better adapts the vectors  $\underline{y}$  to the calculation of approximate distance using the sparse matrix index. The method aims to more uniformly distribute data on the unit sphere while preserving the relative positions of the original data vectors. A uniform distribution is desirable because it maximizes the average minimum pairwise correlation  $\min_{\underline{y}, \underline{y}'} \langle \underline{y}, \underline{y}' \rangle$ , thus allowing for a larger distance approximation error without affecting data ranking. Furthermore, a uniform distribution better distributes the coefficients of all  $\underline{x}_b$  amongst sparse matrix index row bins. Query complexity is related to bin size and thus reducing the largest bin size is a good complexity reduction strategy.

We express the desired uniform distribution and distance preserving characteristics of the conditioned vectors  $\underline{y}^c$  as follows:

$$\underline{y}^c \sim \begin{cases} 1/V_1 & |\underline{y}^c| = 1 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad (7a)$$

$$\langle \underline{y}, \underline{y}_1 \rangle \leq \langle \underline{y}, \underline{y}_2 \rangle \iff \langle \underline{y}^c, \underline{y}_1^c \rangle \leq \langle \underline{y}^c, \underline{y}_2^c \rangle \quad (7b)$$

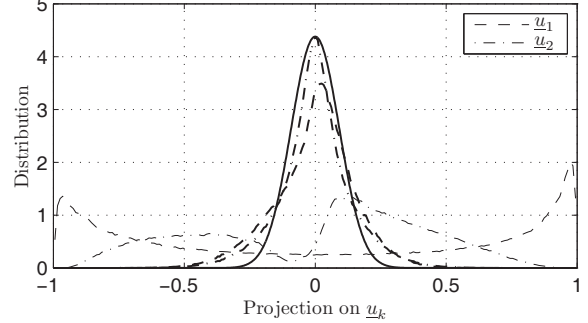
where  $V_1$  denotes the surface area of the unit sphere. We propose an approximation of the conditions (7) based on a singular value decomposition  $\mathbf{U}_y^t \mathbf{D}_y \mathbf{U}_y$  of  $\mathbb{E}[\underline{y} \underline{y}^t]$ :

$$\underline{y}^c = \frac{\mathbf{D}_y^{-1/2} \mathbf{U}_y \underline{y}}{|\mathbf{D}_y^{-1/2} \mathbf{U}_y \underline{y}|}. \quad (8)$$

The proposed approach is justified by noting that, since the  $\underline{y}$  are normalized, the singular vectors in  $\mathbf{U}_y$  are determined by concentrations of points on the unit sphere, with singular values measuring the corresponding point density. Hence the vectors  $\mathbf{D}_y^{-1/2} \mathbf{U}_y \underline{y}$  have undergone anisotropic scaling along principal directions in a manner inversely proportional to the point density; renormalizing subsequently spreads high density concentrations, thus addressing condition (7a). Regarding the second condition (7b), we consider the 2-D case, where it is easy to see that anisotropic scaling and renormalization preserves the relative neighbors of a given point along *either of two* angular directions.

#### 5. RESULTS

We construct our query and index vectors  $\underline{y}_q$  and  $\underline{y}_b$  by extracting up to 150 MSER [5, 6, 7] regions from images of the *Holidays* dataset [2]. This results in 60,909 MSER regions of arbitrary size that are affine normalized to size  $11 \times 11$  and vectorized to obtain the query



**Fig. 1.** Distribution of projections of  $\underline{y}_b$  (light lines) and  $\underline{y}_b^c$  (dark lines) on their first two principal vectors  $\underline{u}_1$  and  $\underline{u}_2$  versus ideal distribution (solid line).

vectors  $\underline{y}_q \in \mathbb{R}^{121}$ . To obtain the index vectors  $\underline{y}_b$ , we need to model residual normalization errors as they cause instabilities in the sparse support of  $\underline{x}$ . We do this by applying random affine transformations of small magnitude to the original normalized query patches  $\underline{y}_q$ . The transformations are defined as mapping the equilateral triangle with vertices  $\underline{v}$  on the unit circle to one with vertices  $\underline{v} + \underline{n}$ , where  $\underline{n}$  is a random vector taken uniformly over the circle of radius 0.3. For each  $\underline{y}_q$ , we build 10 such realizations, for a total of 609,090 index vectors  $\underline{y}_b$ . Since we employ inner-product distances, both  $\underline{y}_q$  and  $\underline{y}_b$  are recentered using the mean vector calculated on the  $\underline{y}_b$  set. Given the  $\underline{y}_q$  and  $\underline{y}_b$  sets thus constructed, we further build vectors  $\underline{y}_q^c$  and  $\underline{y}_b^c$  using (8) trained on the  $\underline{y}_b$ . All sparse decompositions  $\underline{x}$  are built using basis pursuit and the DCT dictionary of size  $121 \times 1024$  [8].

We use  $|\mathcal{C}| = 200$  when building the two-stage *reduced vector*, denoted  $\hat{\underline{x}}^i$  to differentiate it from other  $\hat{\underline{x}}$  built using the episode method; episodes are obtained assuming a prior distribution given by the residual normalization model described above.

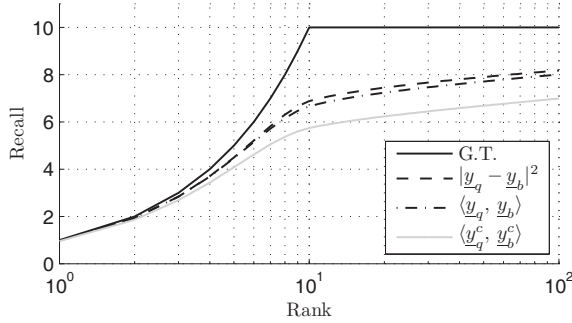
System performance will be measured by recall, i.e., the number of correct realizations (up to 10)  $\underline{y}_b$  retrieved. Complexity will be measured by the total number of non-zero coefficients in all index bins accessed by the query. This measure neglects pre-processing overhead (eg., sparse decomposition of  $\underline{y}_q$  and query-side *reduced vector* construction) yet this is valid since 1) query complexity is much larger than pre-processing overhead for sufficiently large databases; 2) the application (discussed in the introduction) involving a compact image/local descriptors package provides pre-computed sparse representations; and 3) for compressible signals,  $\underline{\chi}_q$  (cf. (1)) can be obtained directly from  $\underline{y}_q$  with no need of  $\underline{x}_q$  since  $\underline{\chi}_q = \mathbf{D}^t \mathbf{D} \underline{x}_q = \mathbf{D}^t \underline{y}_q$ . We will compare our proposed systems relative to the reference distances

$$d_1(\underline{x}_q, \underline{x}_b) = \underline{x}_q^t \underline{x}_b \quad \text{and} \quad (9a)$$

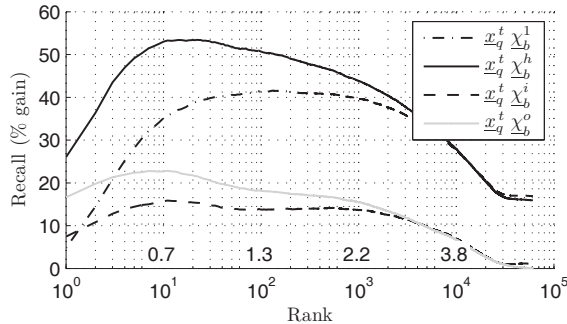
$$d_2(\underline{x}_q, \underline{x}_b) = \langle \underline{x}_q, \underline{x}_b \rangle. \quad (9b)$$

##### 5.1. Evaluation of data conditioning

In Fig. 1 we evaluate how well (8) satisfies the uniform distribution condition (7a) by comparing the distribution of projections of  $\underline{y}_b$  or  $\underline{y}_b^c$  unto their respective first two principal vectors  $\underline{u}_1$  and  $\underline{u}_2$ . Projections along the first two principal vectors deviated the most from the ideal distribution, given by  $\frac{1}{\alpha}(\sqrt{1-p^2})^{119}$  with projection  $p \in [-1, 1]$  and  $\alpha$  a normalization constant. It is evident from the graph that the conditioning scheme succeeds in better distributing the data.



**Fig. 2.** Recall of  $\langle y_q, y_b \rangle$ ,  $\langle y_q^c, y_b^c \rangle$  and  $l$ -2 distance on unnormalized vectors  $y$ . The solid line is the ideal response.



**Fig. 3.** Percent gain of  $\hat{\chi}_b^h$  in (6),  $\hat{\chi}_b^1$  in (5),  $\hat{\chi}_b^o$  in (2) and the two-stage  $\hat{\chi}_b^i$ . The reference recall values are the best recall per rank of (9) (given above the abscissa).

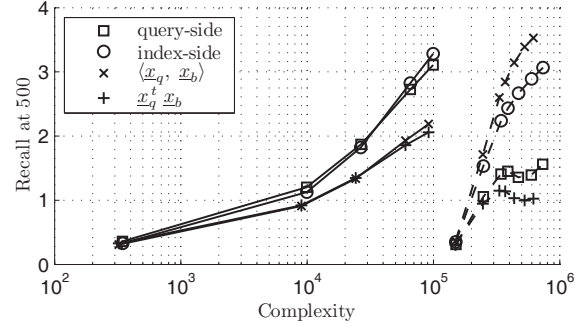
To evaluate the distance preserving condition (7b) we compare average recall under  $\langle y_q, y_b \rangle$  and  $\langle y_q^c, y_b^c \rangle$  to the ground truth recall. Results are shown in figure Fig. 2, along with the recall under the euclidean distance (with unnormalized  $y_q$  and  $y_b$ ) used elsewhere [6]. Data conditioning degrades the response relative to the unconditioned data, but the response is still of sufficient quality to yield an advantage for  $y^c$  in a subsequent test.

## 5.2. Improvement over reference systems

We compare the various methods used to construct index-side *reduced vectors*  $\hat{\chi}_b$  from conditioned data vectors  $y^c$  in Fig. 3. As a reference we take the maximum recall (at each rank) obtained under either reference system in (9) and plot the percent gain over this reference. Note that all systems display a performance advantage.

In Fig. 4,  $x_q^t \hat{\chi}_b^h$  (the best system of Fig. 3) and its query-side version  $\hat{\chi}_q^{ht} x_b$  are compared against the reference systems (9). Both unconditioned data  $y$  and conditioned data  $y^c$  are used to obtain the sparse decompositions  $x$ . We first note that the  $y$  data results in a wide spread in performance between both reference systems. The systems based on  $\hat{\chi}$  consistently outperform the related reference system (9a), yet the reference system (9b) outperforms all.

The unconditioned dataset  $y^c$  results in comparable performance for both reference systems, indicating a more uniform sparse norm  $|x|$ . The  $\hat{\chi}^h$  systems continue to improve upon (9a) and thus also (9b). At a complexity of  $10^5$ ,  $x_q^t \hat{\chi}_b^h$  performs at a recall of 3.3 that is 50% better than that of the best reference system, 2.2. Our data conditioning scheme offers a clear advantage: for a fixed recall,  $\hat{\chi}^h$  systems built using  $y^c$  display a complexity improvement of 0.5 to 3 orders of magnitude relative to any system built using  $y$ .



**Fig. 4.** Recall at rank 500 versus query complexity when using the proposed distances  $\hat{\chi}_q^{ht} x_b$  and  $x_q^t \hat{\chi}_b^h$  based respectively on query-side and index-side *reduced vectors* (built as in (6)) and the reference distances in (9). Solid (respectively dashed) lines are obtained using the conditioned (unconditioned) datasets  $y$  ( $y^c$ ).

## 6. CONCLUSION

We introduced a method that succeeded in improving the performance / complexity tradeoff when approximating the normalized inner-product distance between compressible signals relative to approximations that use the sparse representations directly. The proposed approach included a method for data conditioning that succeeds in preserving the relative positions of the original data points while making their distribution uniform over the unit sphere. The conditioning method is verified experimentally, and its application to the distance approximation scheme proposed yields complexity improvements as high as 3 orders of magnitude for the same performance.

## 7. REFERENCES

- [1] Zhen Jia, Laurent Amsaleg, and Patrick Gros, "Content-based image retrieval from a large image database," *Pattern Recogn.*, vol. 41, no. 5, pp. 1479–1495, 2008.
- [2] Hervé Jégou, Matthijs Douze, and Cordelia Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *ECCV*, 2008, vol. I, pp. 304–317.
- [3] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proceedings of the ICCV*, Oct. 2003, vol. 2, pp. 1470–1477.
- [4] Joaquin Zepeda, Ewa Kijak, and Christine Guillemot, "Sift-based local image description using sparse representations," in *Proceedings of the IEEE International Workshop on MMSP*, 2009.
- [5] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," in *In British Machine Vision Conference*, 2002, pp. 384–393.
- [6] Krystian Mikolajczyk and Cordelia Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.
- [7] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *IJCV*, vol. 65, no. 1-2, pp. 43–72, 2005.
- [8] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.