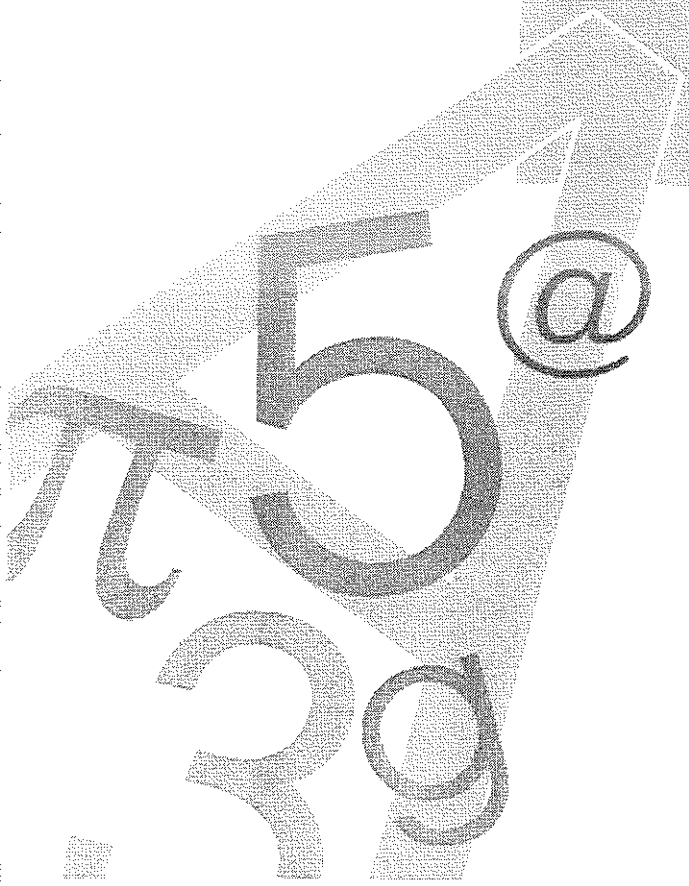
Consiglio Nazionale delle Ricerche

*M*etric-Preserving Data Compression by Noise-like Coding

Sergio Bottini

IEI: B4-38-12-99



Metric-Preserving Data Compression by Noise-like Coding

Sergio Bottini

Istituto di Elaborazione della Informazione
Consiglio Nazionale delle Ricerche
Area della Ricerca di Pisa

Abstract

The noise-like coding, which was first defined in our early model of associative memory, provides a very good solution to the problem of massively measuring the degree of similarity between high-dimensional items over large collections, according to a specific metric, with the minimum computational cost. In this paper, we give the most efficient form of the noise-like coding to implement different metrics, among which the city block (L_1) and Euclidean (L_2) metrics. Our coding transforms the representative feature vectors of the original items into new vectors, called noise-like keys, with a much lower dimension. Superimposition is the metric-preserving principle used for reducing the item dimensionality. This is achieved by *i)* assigning pre-fixed random keys to the features of the items; *ii)* suitably weighting these random keys with the specific values of the features to realize the metric chosen; and finally *iii)* superimposing the results additively to obtain the compressed noise-like keys. In the established correspondence between the high-dimensional feature space of the items and the lower-dimensional space of such noise-like keys, the distances are conserved on the average, with a very good accuracy. Their reduced calculation, for all the defined metrics, is invariably based on the measure of the cosine of the angle separating certain two noise-like keys. Thus, the level of lowering of the computation time is the same as the spatial compression. For items of dimension 10^n (with n here tested up to seven) the compression factor may range from the order of about 10^{n-3} to 10^{n-2}, depending on the desired accuracy in the metric measurements.

Categories and Subject Descriptors: E.4 [**Coding and Information Theory**]: *Data compaction and compression*; H.3.0 [**Information Storage and Retrieval**]: General; I.4.2 [**Image Processing and Computer Vision**]: Compression (Coding) – *Approximate methods*; I.5.1 [**Pattern Recognition**]: Models – *Neural nets*; I.5.3 [**Pattern Recognition**]: Clustering – *Similarity measures*

1. *Introduction*

Performing massive measures of similarity over large collections of items which are represented by feature vectors with a very high dimensionality requires more efficient and effective methods of treating unstructured information. The aim is making these measurements extremely faster, nevertheless maintaining a good accuracy. Generally, the similarity between items is evaluated in accordance with a given metric: the closer two items are in distance, the more similar they are considered. This kind of measures is widely used in the field of information retrieval and in many applications throughout a variety of different sectors. Among the most common criteria of similarity there are the city block and Euclidean metrics, which belong to the class of the L_p metrics, for p = 1 or 2, respectively, with the distances defined by

$$d(I',I)_p = [\sum_{j=1}^{N_{item}} |I'_j - I_j|^p]^{\frac{1}{p}} \tag{1}$$

where I_j and I'_j are the respective components of two N_{item}-dimensional vectors, $\boldsymbol{I}$ and $\boldsymbol{I}'$, describing two given items. Also used is a measure related to the Euclidean metric, namely the measure of the cosine of the angle which separates these representative vectors

$$\cos\phi\,(\boldsymbol{I}',\boldsymbol{I}) = \frac{\boldsymbol{I}'\cdot\boldsymbol{I}}{\|\boldsymbol{I}'\|\ \|\boldsymbol{I}\|} \tag{2}$$

i.e. their normalized inner product. The significance of the cosine similarity relies on the fact that two vectors with close values of their homologous components point into near directions. Essentially, all these measures make it possible to rank the items of a collection in accordance with the found degree of similarity to a given current item, thus serving functions such as the recovering of relevant items from an archive for a given query, the clustering process, or the automated categorization (for a review, see Refs. [1-3]).

The present strategy for obtaining a reliable measure of the item similarity with the minimum amount of computation consists in strongly lowering the item dimensionality through a coding pre-processing step, while preserving the chosen metric in the compressed space. A solution to this difficult task has been initially found in the context of the associative memories long before it was recognized as a problem. This was the case of our model of associative memory based on the noise-like coding [4 - 9]. Our early coding consisted of a set of pre-assigned random vectors, one for each feature of the items, by which the items are transformed into the so called *noise-like keys*, through the linear combination of these random vectors with the specific values of the item features as coefficients. The space of the original items is thus made to correspond to the space of the noise-like keys in which the cosine measurement is conserved on the average. Therefore, it is possible to perform a reduced, faster measure of the cosine similarity between high-dimensional items by choosing a much lower dimension for the noise-like keys. The possibility of reducing the dimensionality of the representative space of the items in our associative memory was, on the other hand, responsible for the dramatic damage-tolerance for cancellation of large parts of the recall-keys and memory traces.

There are two crucial points of the early noise-like coding that will be substantially improved here. One strictly concerns the metric-preserving pre-processing step which codes the items into the corresponding noise-like keys. This operation of coding, which is preliminary to the metric computation, will be made much faster by surprisingly reducing its complexity by some orders of magnitude. The other point concerns the possibility of defining distinct variants of the noise-like coding to establish different metrics by which the degree of similarity between items can be alternatively measured. Specific encoding rules will be introduced to implement the Euclidean (computable from the cosine measure) and the city block metrics. In all cases, the accuracy of the approximate calculation of the various distances carried out in the reduced space of the noise-like keys will result to be very good even for factors of spatial compression, and equivalent levels of time reduction, extraordinarily high.

2. *The early noise-like coding*

In our model of an associative noise-like coding memory, the recall performs a measure of the similarity degree between the item acting as the current input to the memory and the items stored, according to the above mentioned measure of the cosine of the angle formed by the respective representative vectors. The items are feature vectors $\boldsymbol{I}$ of dimension N_{item}, generally with real components normalized between 0 and 1, *i.e.* $0 \leq I_j \leq 1$ (for $j = 1, \ldots, N_{item}$). The noise-like coding (NLC) is a pre-processing step by which all the items ($\boldsymbol{I}$) that enter the memory system, both on storage and recall, are transformed into noise-like keys (v):

$$\boldsymbol{I} \xrightarrow{\text{NLC}} v\,. \tag{3}$$

The central idea was that "each feature (of the item) is made to correspond to a given fixed random key" [7], specifically a random vector $\alpha^{(j)}$ of dimension N_{key} (for $j = 1, 2, \ldots, N_{item}$) with zero-mean components, so that, in the model simulations [4, 5], the whole item $\boldsymbol{I}$ was coded by a weighted linear superimposition of all these elemental random vectors, *i.e.*

$$v = \sum_{j=1}^{N_{item}} I_j\, \alpha^{(j)}. \tag{4}$$

The sequence of convolution and correlation (indicated with $*$ and $\circledast$) that, in the model, performs the storage of item $\boldsymbol{I}$ into the memory and its recall ($\boldsymbol{R}$) by the current input $\boldsymbol{I}'$, respectively, yields

$$\boldsymbol{R} = \frac{v'}{\|v'\|} \circledast \left(\frac{v}{\|v\|} * \boldsymbol{I}\right) \approx \boldsymbol{I} \cos\theta(v', v) \tag{5}$$

in which v' is the noise-like key obtained from $\boldsymbol{I}'$ still through Eq. (4); $\|\cdot\|$ denotes the L_2 norm; $\theta(v', v)$ is the angle comprised between the directions of the two noise-like vectors v' and v; and

$$\cos\theta(v', v) = \frac{v' \cdot v}{\|v'\|\ \|v\|}. \tag{6}$$

The recalled item $\boldsymbol{R}$ is thus scaled according to whether $\boldsymbol{I}'$ is a greater or a smaller part of $\boldsymbol{I}$. In fact, from the property of quasi-orthogonality [4, 7] of the random vectors $\alpha^{(j)}$, with a constant L_2 norm, *i.e.*

$$\alpha^{(j)} \cdot \alpha^{(j')} \propto \delta_{jj'} \tag{7}$$

(where the proportionality factor equals their norm squared and δ is the Kronecker symbol), it follows

$$\cos\theta\,(v', v) \approx \cos\phi\,(\boldsymbol{I}', \boldsymbol{I}) \tag{8}$$

with $\phi\,(\boldsymbol{I}', \boldsymbol{I})$ being the angle between the vectors $\boldsymbol{I}'$ and $\boldsymbol{I}$, Eq. (2). This means that for a dimension of the noise-like vectors v, N_{key}, much lower than the dimension of items $\boldsymbol{I}$, N_{item}, the approximate calculation of cos $\phi\,(\boldsymbol{I}', \boldsymbol{I})$, performed in the space of these noise-like keys as cos $\theta\,(v', v)$, is much less expensive than its exact calculation in the space of the original items $\boldsymbol{I}$. This fact, furthermore, is definitely more remarkable since, as it will be seen, even the computation complexity of the noise-like coding expressed by Eq. (4) can be strongly reduced by some orders of magnitude in the number of the elementary operations needed.

3. *The most efficient noise-like coding*.

Very interestingly, a drastic simplification can be introduced in our original noise-like coding. It consists in making each feature I_j (with $j = 1, \ldots, N_{item}$) of the item being coded to correspond to only one random number, instead of a random vector of dimension N_{key} as initially done. Furthermore, more than one metric (and, potentially, a large variety of metrics) can be realized with our coding. The new noise-like coding takes the following form

$$v_i = \sum_{j=(i-1)n+1}^{i\,n} f(I_j) \tag{9}$$

for $i = 1, \ldots, N_{key}$, in general with $N_{key} << N_{item}$; where $n = N_{item}/N_{key}$, and f is a scalar-valued "randomizing" function that defines the metric. In other words, f determines which type of distance between the items $\boldsymbol{I}$ can be obtained from the direct measure of the cosine of the angle formed by the corresponding noise-like keys. The number of operations required by Eq. (9) to code item $\boldsymbol{I}$ is N_{key} times smaller than that required by Eq. (4), with N_{key} being, in general, of the order of a hundred to a thousand. The particular choice of dimension N_{key} of the noise-like keys is related to the desired accuracy in the calculation of the various distances, with a lower variance for a greater N_{key}.

***The Euclidean (L_2) metric*.** For

$$f(I_j) = u_j\, I_j \tag{10}$$

where the u_j are fixed zero-mean real random numbers with a uniform distribution, or, alternatively, fixed binary random numbers equal to ±1, we again obtain Eq.(8) but at much less computational expense. As an implication of Eq. (8), it is

$$d^2(\boldsymbol{I}',\boldsymbol{I})_{Euclidean} \approx \frac{1}{<u^2>}[\|v'\|^2+\|v\|^2-2\|v'\|\|v\|\cos\theta(v',v)] \tag{11}$$

where $<u^2>$ is the mean value of the u_j squared, and $d(\boldsymbol{I}',\boldsymbol{I})_{Euclidean}$ is the usual Euclidean distance given by Eq. (1) for $p = 2$. In particular, in the important case of items with a constant L_2 norm, $\|\boldsymbol{I}'\|^2 = \|\boldsymbol{I}\|^2 = C^2$, Eq. (11) becomes

$$\frac{1}{C}\, d(\boldsymbol{I}',\boldsymbol{I})_{Euclidean} \approx [2\,(1-\cos\theta(v',v)]^{\frac{1}{2}} = 2\sin\tfrac{1}{2}\theta(v',v). \tag{12}$$

Proof. To demonstrate that for the noise-like coding function f given by Eq. (10) the approximate equality expressed by Eq. (8) still holds true, we start defining $\alpha^{(j)}$ as a vector of dimension N_{key} having only one nonzero component equal to u_j in the i-th location determined by $i = \mathrm{int}\left(\frac{j-1}{n}\right)+1$, *i.e.* the maximum integer contained in $(j - 1)/n$ increased by one. Thus, Eq. (9) takes the form of a scalar version of Eq. (4), while Eq. (7) remains valid on the average as

$$\mathrm{E}\{\alpha^{(j)}\cdot\alpha^{(j')}\} \propto \delta_{jj'}. \tag{13}$$

Consequently, it still turns out $v'\cdot v \propto \boldsymbol{I}'\cdot\boldsymbol{I}$, $\|v\|^2 \propto \|\boldsymbol{I}\|^2$ and $\|v'\|^2 \propto \|\boldsymbol{I}'\|^2$ with a common proportionality factor, here equal to $<u^2>$, by which Eq. (8) and, likewise, also Eq. (11) or (12), are proved. Let us note that for $u_j = \pm 1$ the Euclidean distance in the space of the noise-like keys is also conserved in form (besides that in value). This is always true, for any u_j, in the particular case of Eq. (12).

The city block (L_1) metric. The L_1 metric is implemented by applying the following type of noise-like coding to item $\boldsymbol{I}$:

$$f(I_j) = d_j + (u_j - d_j)H\{I_j - \xi_j\} \tag{14}$$

where the u_j and d_j are fixed zero-mean real random numbers with a uniform distribution, or, alternatively, fixed binary random numbers (*i.e.* $u_j = \pm 1$ and $d_j = \pm 1$); the ξ_j are fixed real random numbers with a uniform distribution in the same range as the feature values I_j of the items, *i.e.* with $0 \le \xi_j \le 1$; and $H\{\cdot\}$ is the Heaviside function (unity for positive values and zero elsewhere). In fact, it results

$$\frac{1}{N_{item}}\, d(\boldsymbol{I}',\boldsymbol{I})_{city-block} \approx 1-\cos\theta(v',v) \tag{15}$$

where $d(\boldsymbol{I}',\boldsymbol{I})_{city-block}$ is the city block distance between the items $\boldsymbol{I}'$ and $\boldsymbol{I}$, given by Eq. (1) for $p = 1$.

Proof. The noise-like coding function *f* defined by Eq. (14) assigns two pre-fixed random numbers to each item feature, u_j and d_j (for $j = 1, \ldots, N_{item}$). These numbers can be considered to form two N_{key}-dimensional vectors, $\alpha^{(j)}$ and $\alpha_{\#}^{(j)}$, with u_j and d_j, respectively, as i-th component for $i = \mathrm{int}\left(\frac{j-1}{n}\right) + 1$, and zero elsewhere. Then, Eq. (9) can be re-written as

$$v = \sum_{j=1}^{N_{item}} \mu^{(j)}(I_j) \tag{16}$$

where $\mu^{(j)}(I_j)$ is either $\alpha^{(j)}$ or $\alpha_{\#}^{(j)}$ depending on I_j, precisely according to whether $I_j > \xi_j$ or $I_j \le \xi_j$. Since

$$\mathrm{E}\{\, \mu^{(j)}(I'_j) \cdot \mu^{(j')}(I_{j'}) \,\} = 0 \qquad\qquad \text{for } j \ne j' \tag{17}$$

and

$$\mathrm{E}\{\, \mu^{(j)}(I'_j) \cdot \mu^{(j)}(I_j) \,\} \propto 1 - | I'_j - I_j | \tag{18}$$

we have

$$v' \cdot v \propto N_{item} - \sum_{j=1}^{N_{item}} | I'_j - I_j | \tag{19}$$

and

$$\| v' \| \; \| v \| \propto N_{item} \tag{20}$$

with the same proportionality factor. Hence Eq. (15) is shown to be valid.

Another metric. From the restriction of Eq. (14) realized for all $d_j = 0$, *i.e.*

$$f(I_j) = u_j \, H\{ I_j - \xi_j \} \tag{21}$$

we obtain

$$d(\boldsymbol{I}', \boldsymbol{I})_{min} \approx 1 - \cos\theta\,(v', v) \tag{22}$$

where

$$d(\boldsymbol{I}', \boldsymbol{I})_{min} = 1 - \frac{1}{(\|\boldsymbol{I}'\|_1 \, \|\boldsymbol{I}\|_1)^{1/2}} \sum_{j=1}^{N_{item}} min\left\{ I'_j ; I_j \right\} \tag{23}$$

is the distance defined as the complement to unity of the summation of the lower values of the respective features of items $\boldsymbol{I}'$ and $\boldsymbol{I}$ divided by the square root of the product of their L_1 norms (indicated with $\|\cdot\|_1$).

Proof. For the f given by Eq. (21), we can retrace the scheme of the demonstration used in the case of the city block metric. Again, Eq. (9) takes the form of Eq. (16), where, however, $\mu^{(j)}(I_j)$ now is either $\alpha^{(j)}$ or 0 according to whether $I_j > \xi_j$ or $I_j \leq \xi_j$. Equation (17) still holds true, while Eq. (18) writes

$$\mathrm{E}\{\mu^{(j)}(I'_j)\cdot\mu^{(j)}(I_j)\} \propto min\left\{I'_j;I_j\right\}. \tag{24}$$

Accordingly

$$v'\cdot v \propto \sum_{j=1}^{N_{item}} min\left\{I'_j;I_j\right\} \tag{25}$$

and

$$\|v'\|\,\|v\| \propto \|\boldsymbol{I}'\|_1\,\|\boldsymbol{I}\|_1 \tag{26}$$

with the same proportionality factor, whence Eq. (22) follows.

Observation. If threshold ξ_j in Eq. (21) was changing at random any time for each j (instead of being fixed), then the equivalent of Eq. (22) would be

$$\cos\theta(v', v) \approx \frac{\boldsymbol{I}'\cdot\boldsymbol{I}}{(\|\boldsymbol{I}'\|_1\,\|\boldsymbol{I}\|_1)^{1/2}}. \tag{27}$$

Similarly, if the noise-like key v was coded by Eq. (14) and the noise-like key v' by Eq.(21), still with the ξ_j varying at chance any time, we would obtain

$$\cos\theta(v', v) \approx \frac{\boldsymbol{I}'\cdot\boldsymbol{I}}{(N_{\text{item}}\,\|\boldsymbol{I}'\|_1)^{1/2}}. \tag{28}$$

This means that the method of the variable threshold provides a new coding rule to implement again the inner product and, under the condition that items $\boldsymbol{I}$ have a constant L_2 norm, the cosine measure and, accordingly, the Euclidean metric.

Generalization of the noise-like coding. Assigning a single random number to each item feature is clearly the most economical choice. However, in view of covering any possible range of applications, we consider that in certain cases it might be useful to have a supplementary degree of freedom in associating strings of random numbers of length greater than one with the item features (although, possibly, only a small fraction of the entire length N_{key} of the noise-like keys v). Thus, for the sake of completeness, in the Appendix we give such a general form of the noise-like coding.

4. *Item collections re-written as highly-compressed associative memories*

With our coding, we can then build a representation of a given collection of N_{TOT} items $\boldsymbol{I}^{[k]}$

$$A = \{\boldsymbol{I}^{[k]}\}_{k=1,\ldots,N_{TOT}} \tag{29}$$

in terms of the corresponding noise-like keys as

$$M = \{ \frac{v^{[k]}}{\|v^{[k]}\|} \}_{k=1,\dots,N_{TOT}} \tag{30}$$

in which the dimension of the vectors representing the items is lowered from N_{item} to N_{key}. As has been seen, the transformation of A into its compressed version M preserves the metric established by the specific noise-like coding applied. The choice of N_{key} will, of course, determine the accuracy in the reduced calculation of the distances performed on M. Dimension N_{key} and the number of bits used for each component of $v^{[k]}$ determine together the factor of spatial compression, *i.e.* the ratio between the capacities required to store A and M. For example, in the case of the city block metric with binary u_j and d_j, where therefore any single contribution in forming the component values of $v^{[k]}$ in Eq. (9) is constant for all j, this number of bits per key-element can be easily evaluated as being of the order of $log_2(6n^{1/2})$ [cf. Ref. 7]. This value is, furthermore, substantially indicative even of all other cases, so that the minimum capacity sufficient to store M is always readily computable.

5. *Binary noise-like keys*

A further property of our coding is its capability of producing binary noise-like keys that still preserve the metrics. This is realized by operating a quantization into two levels of the components of the final noise-like key v associated with each item I, only retaining the bit of their sign. The transformation of a noise-like key into a binary vector still maintains valid all the results shown, although, of course, with an increase in the variance for the calculated distances. This property was studied in Ref. [7], where the memory was quantized into two levels at the end of the processes of storage of all the items, with the consequence that the storage capacity only decreased by a factor of $\pi/2$ (changing from the value of $\frac{1}{2}log_2 e$ to $\frac{1}{\pi}log_2 e$ bit/element).

Thus, by applying the sign function to the noise-like keys

$$\hat{v}_i^{[k]} = sgn\,\{v_i^{[k]}\} \tag{31}$$

(with $\hat{v}_i^{[k]} = \pm 1$), a minimal representation of the original item collection A, Eq. (29), can be obtained as

$$\hat{M} = \{\hat{v}^{[k]}\}_{k=1,\dots,N_{TOT}} \tag{32}$$

in terms of the two-level quantized noise-like keys, *i.e.* a collection of N_{TOT} binary vectors, each $\hat{v}^{[k]}$ only consisting of N_{key} bits. The relationship between the cosines of the angles in the space of the noise-like keys in the two cases in which v' is applied to M or $\hat{M}$ is

$$\cos\theta(v', v^{[k]}) \approx \sqrt{\pi/2}\,\cos\theta(v', \hat{v}^{[k]}) \tag{33}$$

from which the correct normalization of the cosine measurements on $\hat{M}$ is obtained.

6. *Computer tests*

The real effectiveness of the new noise-like coding, Eq. (9), in correctly implementing all the defined metrics has been successfully tested on both synthetic vectors of dimension up to 10^6, with uniform distribution components, and items with a potential interest in practical applications, specifically Meteosat images of 800×800 dimensions (with 8-bit pixels). For the synthetic items we have a statistics from samples of a hundred experiments each. There is a complete agreement between the average values of the distances calculated in the highly compressed space of the noise-like keys and the distances measured directly on the original high-dimensional items (Fig. 1). The accuracy depends on the length N_{key} of the noise-like keys v. For statistically independent random keys, the variance in the calculation of $\cos\theta(v', v)$ would be $1/N_{key}$ or, in the particular case of two-level quantized keys, $\pi/(2N_{key})$ [7]. But, by their construction, our keys v strongly correlate with each other even for completely independent items $\boldsymbol{I}$. Consequently, as demonstrated in general in the Appendix of Ref. [7], the variance tends to vanish as the distance between the items approaches zero

$$\lim_{dis\tan ce \to 0} \sigma^2 = 0 \tag{34}$$

still maintaining the dependence on the inverse of the key length

$$\sigma^2 \propto \frac{1}{N_{key}}. \tag{35}$$

This is shown in Figure 2 for the L_1 metric with both real and binary keys. The property expressed by Eq. (34) is particularly lucky for applications. It means that, for all N_{key}, the error introduced by computing the distances through the noise-like coding tends to become increasingly smaller just as the items are more similar, *i.e.* when a greater accuracy is a more crucial quality. This is well evident in the reported tests on the Meteosat images for our three metrics (Figs. 3-5), where, for an exemplifying purpose, values of N_{key} ranging from one hundred to two thousands are used, for 256-level and two-level quantized keys, *i.e.* keys with 1-byte elements (graded keys) or 1-bit elements (binary keys).

Finally, as a further application, we have used the new noise-like coding for the fast calculation of the L_1 and L_2 distances between whole sequences of Meteosat images (Fig. 6). In one case, these sequences have a global dimension of about N_{item}=10^7, which is their total number of pixels, and are coded in single binary noise-like keys with dimension N_{key} = $2\cdot10^2$. Thus, the factor of spatial compression is $4.1\cdot10^5$, which corresponds to only $2\cdot10^{-5}$ bits per pixel spent in the space of the keys to represent one sequence. Factor $4.1\cdot10^5$ also gives the present level of efficiency, *i.e.* the order of lowering of the computational cost.

7. *Concluding comments*

The noise-like coding, that we first defined in our model of associative memory, was a pre-processing step by which any item described by a feature vector could be transformed into a quasi-orthogonal vector, called a noise-like key, and thus stored with the maximum storage efficiency. The early coding consisted in pre-assigning random vectors to the item features and combining them linearly with the current feature values to obtain the final noise-like keys. In the transformation from the original space of the items to the space of the noise-like keys, the dimensionality can be strongly reduced, still preserving the cosine measure. This is due to the property of the pre-assigned random vectors of forming highly-overcomplete quasi-orthogonal bases.

Here, we have introduced a substantial improvement of the noise-like coding just in the crucial step of the preliminary transformation of the items, which precedes the metric measures, by disentangling the choice of the dimension of the noise-like keys, which is related to the accuracy of the measures, from the length of the random keys associated with the item features. In the early noise-like coding, these pre-fixed random keys, whose number equals the item dimensionality, had all the same dimension as the final noise-like keys. Especially for very-high-dimensional items, this would require uselessly-expensive calculations for the pre-processing step of the specific operation of item coding. In the present formulation of the noise-like coding, used in our tests, the new random keys consist of only one random number each, which makes this coding extremely simple and then fast. We have shown, furthermore, that our coding allows us to implement different metrics in addition to the cosine measure, among which the most common ones, namely the city block (L_1) and Euclidean (L_2) metrics, and, potentially, unconventional suitable metrics for specific applications.

Summarizing, the new noise-like coding makes it possible to realize very high levels of compression of data with the minimum effort of computation, while conserving the chosen distance on the average. Huge collections of high-dimensional items can be easily coded into ultra-compressed versions, in which similarity measures are very quickly performed according to the current metric. The accuracy of these fast calculations of distances can be made very good, with the variance decreasing to zero as the items are increasingly similar. The level of reduction in the computation time is the same as the factor of spatial compression of the items, generally of the order of about 10^{n-3} to 10^{n-2} for an item dimension of 10^{n}, depending on the desired accuracy. This is the most efficient noise-like coding for a highly effective and efficient computation of distances in any high-dimensional space.

APPENDIX.

The most general noise-like coding. There are, of course, many intermediate possibilities between making to correspond an N_{key}-dimensional random vector with all nonzero components or a single random number to each item feature I_j (with $j = 1, \ldots, N_{item}$). We could, in fact, choose strings of random numbers of length in between 1 and N_{key}. Then, the most general form of the noise-like coding is

$$v_{i=(q-1)N_s+p} = \sum_{j=(q-1)n+1}^{qn} f_p(I_j) \qquad \text{(A1)}$$

where $p = 1, \ldots, N_s$, with N_s a fixed submultiple of N_{key}; $q = 1, \ldots, N_{key}/N_s$; $n = (N_s\, N_{item})/N_{key}$; and the f_p are N_s scalar-valued randomizing functions that define the metric. In detail, for the Euclidean metric

$$f_p(I_j) = u_{jp}\, I_j \qquad \text{(A2)}$$

for the city block metric

$$f_p(I_j) = d_{jp} + (u_{jp} - d_{jp})\, H\{I_j - \xi_{jp}\} \qquad \text{(A3)}$$

and for the metric defined by Eq. (23)

$$f_p(I_j) = u_{jp}\, H\{I_j - \xi_{jp}\} \qquad \text{(A4)}$$

where u_{jp}, d_{jp}, and ξ_{jp} are the elements of three fixed $N_{item} \times N_s$ matrices consisting of random numbers with the already seen qualities. The value of N_s fixes the length of the strings of random numbers associated with the item features. Clearly Eq. (A1) contains Eqs. (9) and (4) as opposite extreme cases for the length N_s of these strings. In fact, Eq. (A1) becomes Eq. (9) for $N_s = 1$ (accordingly, with $q = i$, and p dropped because always $p = 1$); or Eq. (4) for $N_s = N_{key}$ (with $p = i$), although written in a scalar instead of a vector form, precisely with $f_i(I_j) = \alpha_i^{(j)} I_j \equiv u_{ji}\, I_j$.

To prove that the noise-like coding functions f_p expressed by Eqs. (A2-A4) actually make it possible to calculate the approximate distances in the three given metrics, it suffices to define the analogues of vectors $\alpha^{(j)}$ and $\alpha_{\#}^{(j)}$ introduced in the demonstration of Eq. (9) (*i.e.* with $N_s = 1$) and then follow all its steps in the three cases considered (with the supplementary vectors $\alpha_{\#}^{(j)}$ used only for the city block metric). This is accomplished by considering N_{key}-dimensional vectors $\alpha^{(j)}$ and $\alpha_{\#}^{(j)}$, for each j, having only N_s nonzero components equal to the u_{jp} and d_{jp}, respectively, for $p = 1, \ldots, N_s$, arranged consecutively starting from location $i = N_s \times \text{int}\left(\frac{j-1}{n}\right) + 1$. Let us note, finally, that random vectors $\alpha^{(j)}$ with many zero components were already tested in the simulations reported in Ref. [7], although they were of a sparse kind, *i.e.* not structured in the present convenient form.

References

[1] R Baeza-Yates and B Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley, Harlow, 1999

[2] D A Grossman and O Frieder, *Information Retrieval: Algorithms and Heuristics*, Kluwer Academic Publishers, Boston, 1998

[3] R R Korfhage, Information Storage and Retrieval, John Wiley & Sons, New York, 1997

[4] S Bottini, *An Algebraic Model of an Associative Noise-like Coding Memory*, Biol. Cybernetics **36,** 221-228 (1980)

[5] S Bottini, *Modello Matematico di una Memoria Associativa*, Atti 5° Congresso Nazionale di Cibernetica e Biofisica, 335-342, Pisa 9-11 Aprile 1979

[6] S Bottini, *Un Modello di Memoria Associativa*, Le Scienze, Quaderni n. 19, 41-46, Dicembre (1984)

[7] S Bottini, *An After-Shannon Measure of the Storage Capacity of an Associative Noise-like Coding Memory,* Biol. Cybernetics **59,** 151-159 (1988)

[8] S Bottini, *An Associative Content-Addressable Memory*, Conference on Neural Networks for Computing, 13-16 April Snowbird – Utah – Abstract's Book (1986)

[9] S Bottini, *Noise-like Coding in Associative Memories*, IEEE First Annual International Conference on Neural Networks, June 21-24, San Diego - Abstract's Book (1987)

Figure Captions

Fig. 1. Average distances (dots) between synthetic items (with dimension N_{item}) calculated in the reduced space of the noise-like keys (with dimension N_{key}), for the city block (L_1) and Euclidean (L_2) metrics, with the normalization given by Eqs. (15) and (12), respectively. Both real (**a**) and two-level quantized (**b**) noise-like keys are used. The items have real components uniformly distributed in the range (0, 1). The solid lines represent the same distances directly measured in the space of the original items. The degree of similarity reported in abscissa expresses the percentages of the components with common values between any two items, while the other components have statistically independent values. Various other ways of introducing differences between the items, including slight and/or heavy, random and/or systematic changes of the values of all the components together, were also tested, always with as very good results as the present one. For completely independent items the L_1 and L_2 distances have the respective mean values of 1/3 and $1/\sqrt{2}$. Also shown are the standard deviations of the calculated distances, which tend to zero as the items become increasingly similar.

Fig. 2. Standard deviations, σ, of the measurements of the city block distance in the reduced space of real (**a**) and binary (**b**) noise-like keys, at different values of dimensions N_{key}. For items increasingly closer, σ decreases to zero for all N_{key}. For any fixed value of the distance between the items, the values of σ obey the property of being inversely proportional to the square root of dimension N_{key}.

Fig. 3. City block distances (diamonds) between Meteosat images, with $N_{item} = 6.4 \cdot 10^5$, calculated in the compressed space of the corresponding noise-like keys, graded (with 256-level elements, *i.e.* 1-byte elements) or binary, for two values of N_{key}. For comparison, the same distances (squares) directly measured on the original images are reported. The order in which the various pixels of the images are taken to form the noise-like keys in Eq. (9) is irrelevant, provided, of course, that it remains settled once for all. The factor of compression is 640 in (**a**); 5120 in (**b**), equivalent to $1.6 \cdot 10^{-3}$ bits per pixel; and 2560 in (**c**).

Fig. 4. Euclidean distances (diamonds) calculated through the noise-like coding, compared to their direct measurements (squares) on Meteosat images. In (**a**) and (**b**) the cosine measures (circles) are also shown. The factor of compression in (**c**) is 51200, equivalent to $1.6 \cdot 10^{-4}$ bits per pixel.

Fig. 5. Comparison between the reduced calculations of the distances (diamonds) defined by Eq. (23) and their direct measurements (squares) on Meteosat images.

Fig. 6. Reduced calculation of distances (diamonds) for items consisting of sequences of Meteosat images. The reported distances concern two sequences formed, respectively, by the images occupying the odd and even positions of a series of contiguous images, and then shifted from each other by one position at a time. Each whole sequence of dimension N_{item} is coded as a single noise-like key of dimension N_{key}. Metric L_1 is used in (**a**), for sequences of 9 Meteosat images each, and metric L_2 in (**c**), for sequences of 16 images. For the last case, the reduced cosine measures (circles) are given in (**b**). The direct measurements of distance between the original sequences are indicated with squares. The factor of compression in (**b**) and (**c**) is extraordinarily high, $4.1\cdot 10^5$, equivalent to $2\cdot 10^{-5}$ bits per pixel.

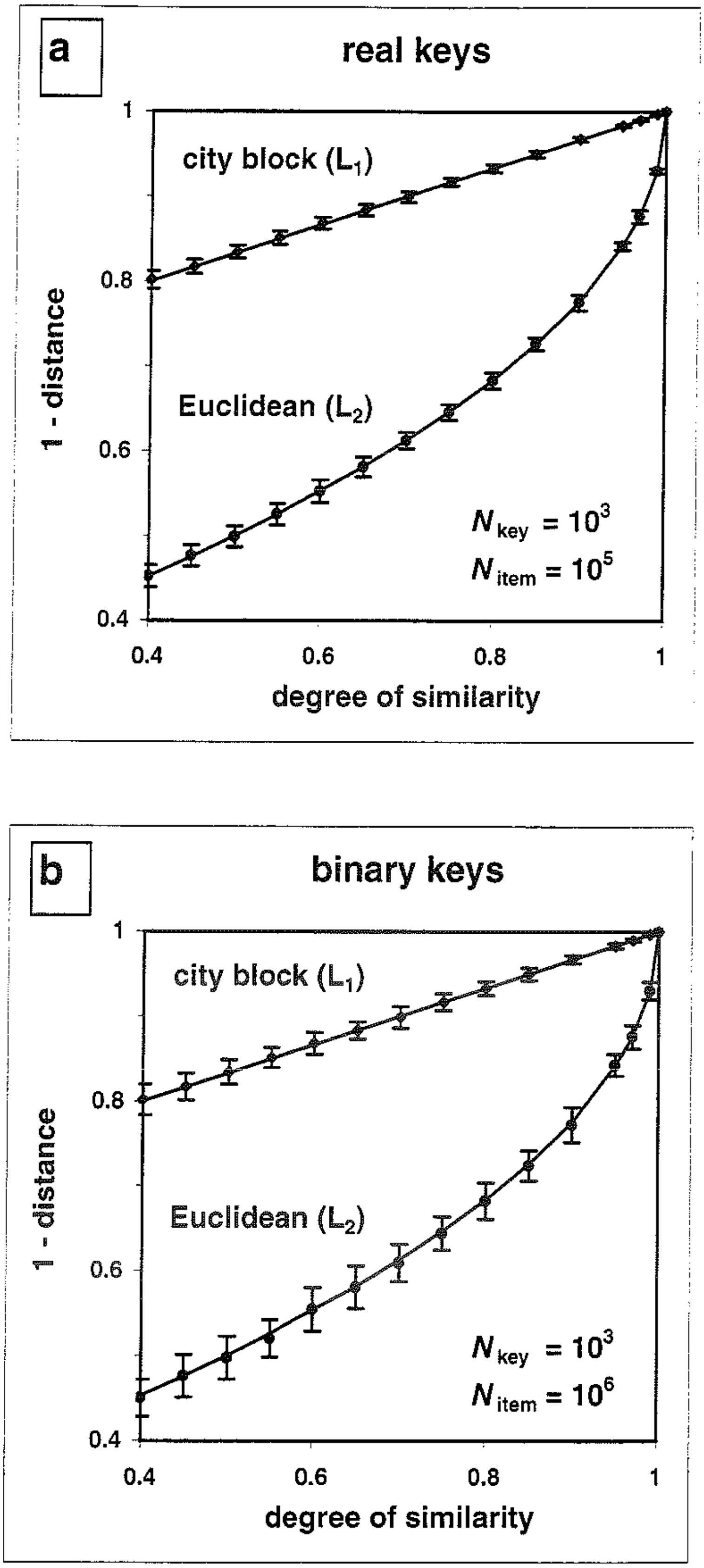

a
real keys
city block (L_1)
Euclidean (L_2)
$N_{key} = 10^3$
$N_{item} = 10^5$
1 - distance
degree of similarity
0.4
0.6
0.8
1
b
binary keys
city block (L_1)
Euclidean (L_2)
$N_{key} = 10^3$
$N_{item} = 10^6$
1 - distance
degree of similarity
0.4
0.6
0.8
1

FIG. 1

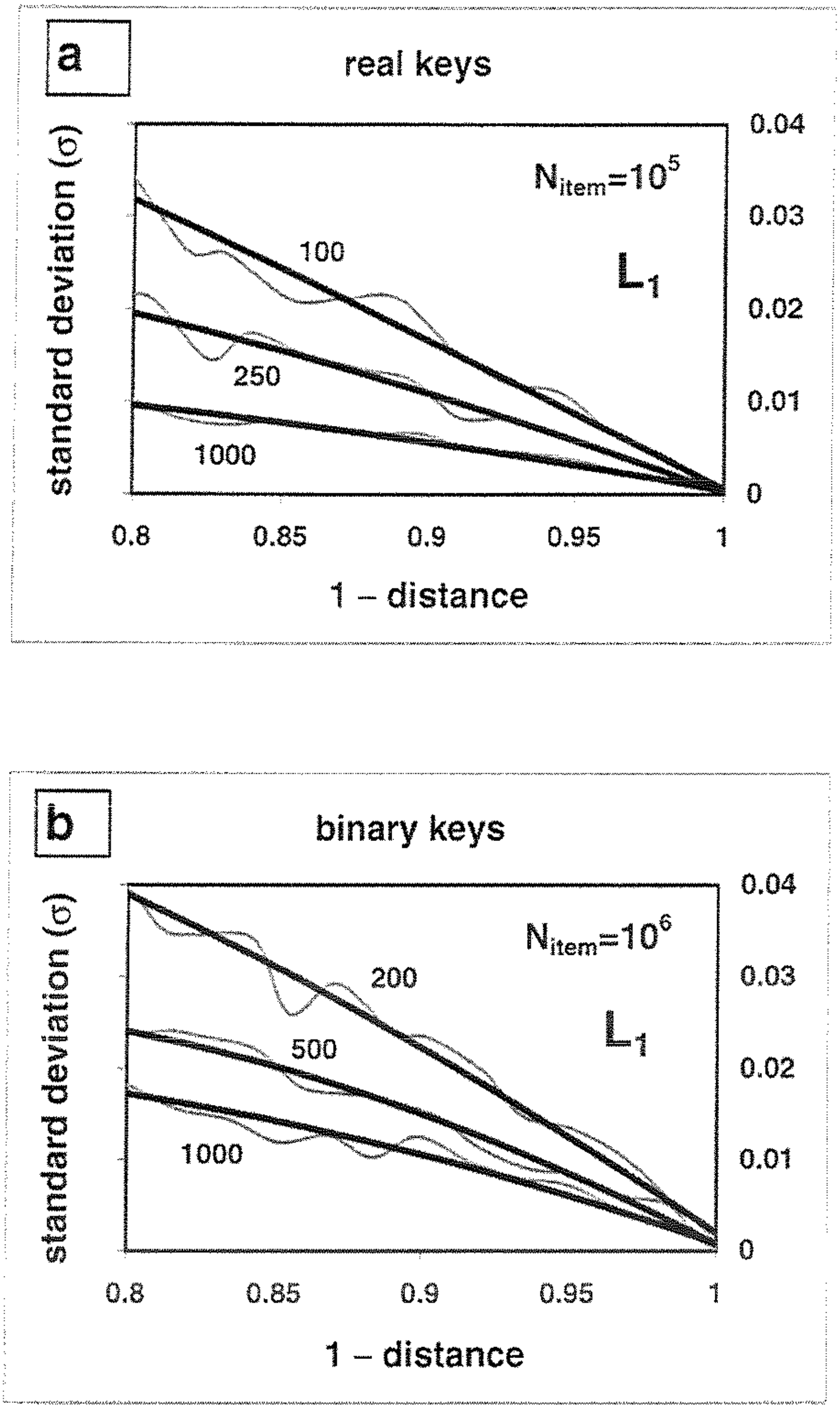

a
real keys
$N_{item}=10^5$
L_1
100
250
1000
standard deviation (σ)
0.04
0.03
0.02
0.01
0
0.8
0.85
0.9
0.95
1
1 – distance
b
binary keys
$N_{item}=10^6$
L_1
200
500
1000
standard deviation (σ)
0.04
0.03
0.02
0.01
0
0.8
0.85
0.9
0.95
1
1 – distance

FIG. 2

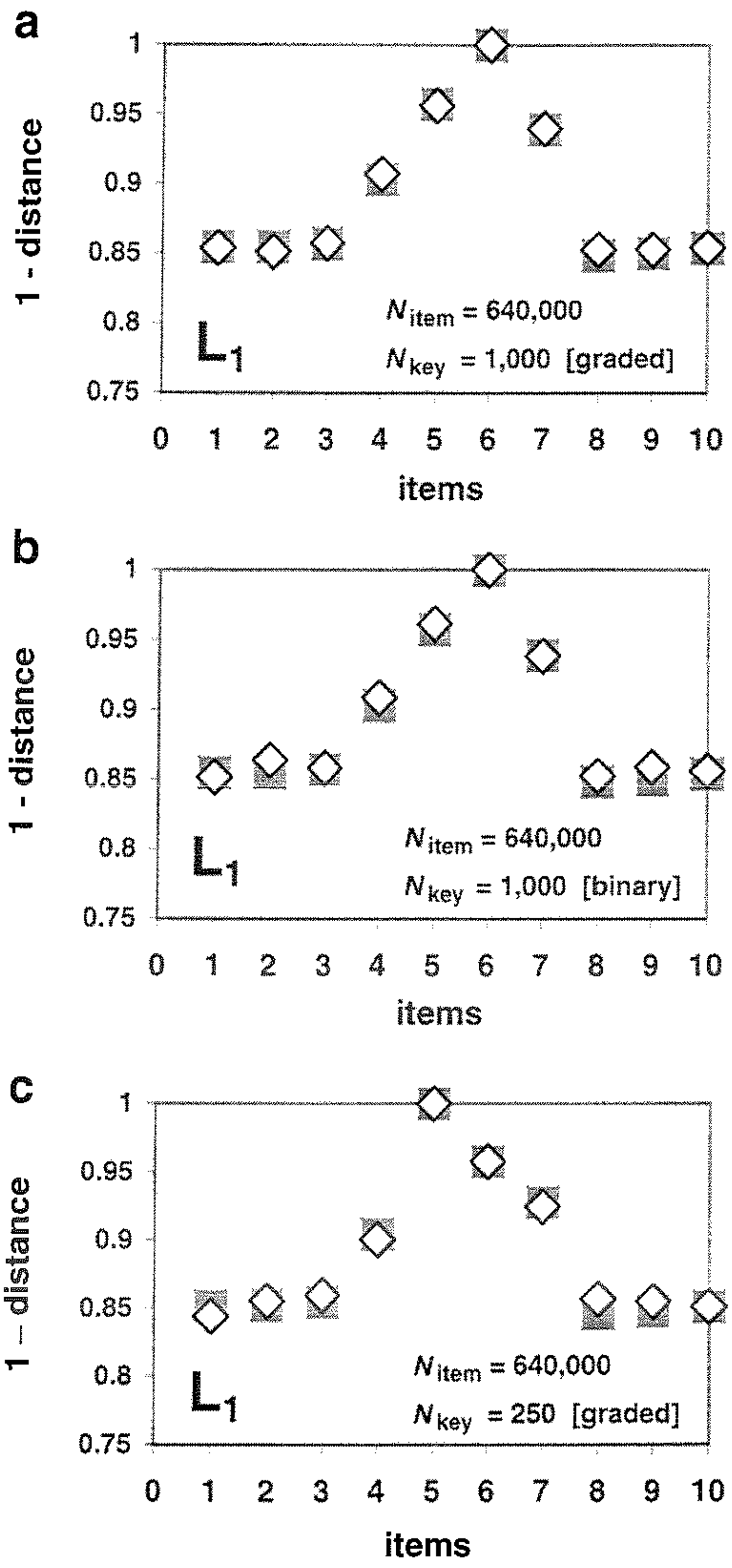

a
1 - distance
1
0.95
0.9
0.85
0.8
0.75
0 1 2 3 4 5 6 7 8 9 10
items
L_1
N_{item} = 640,000
N_{key} = 1,000 [graded]
b
1 - distance
1
0.95
0.9
0.85
0.8
0.75
0 1 2 3 4 5 6 7 8 9 10
items
L_1
N_{item} = 640,000
N_{key} = 1,000 [binary]
c
1 – distance
1
0.95
0.9
0.85
0.8
0.75
0 1 2 3 4 5 6 7 8 9 10
items
L_1
N_{item} = 640,000
N_{key} = 250 [graded]

FIG. 3

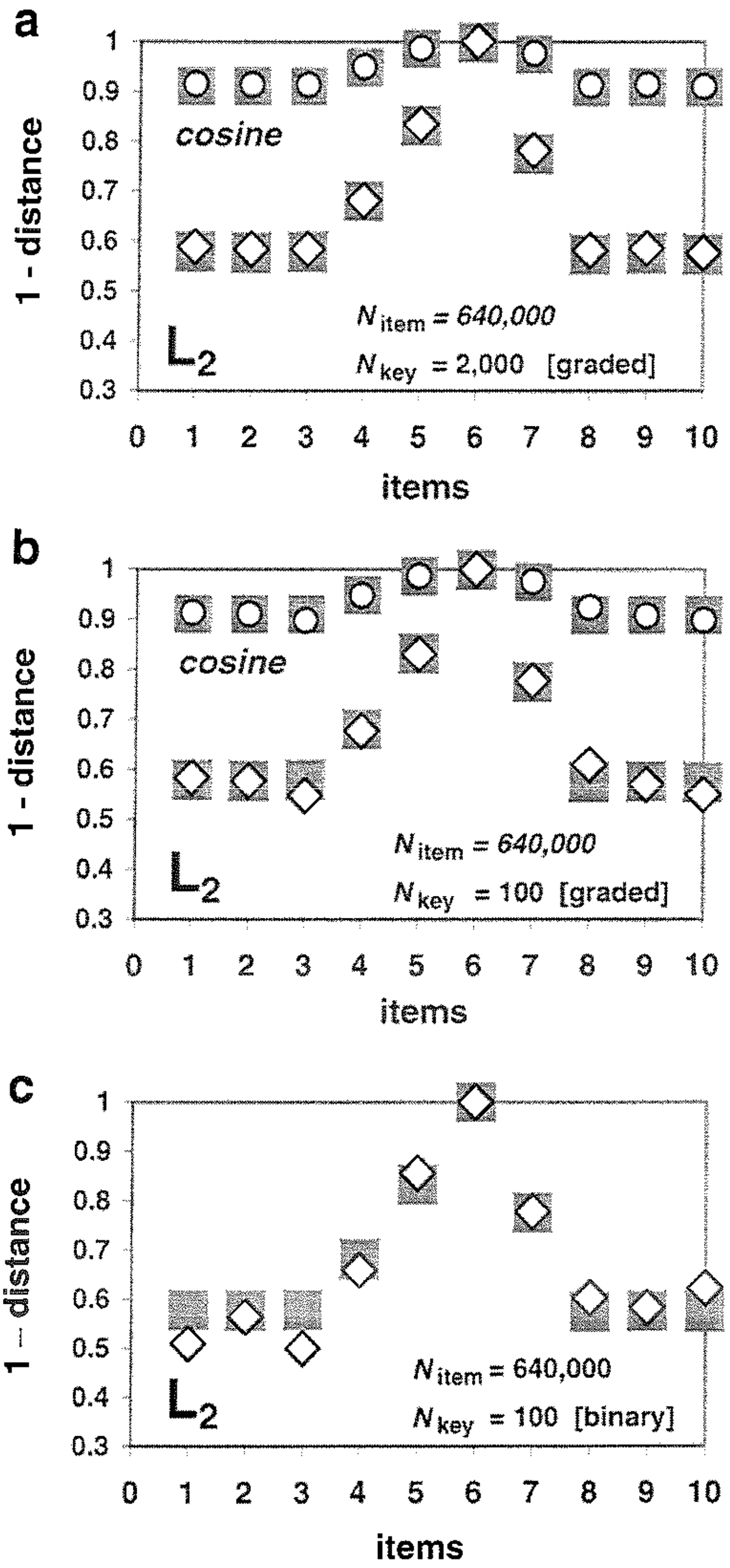
a
1 - distance
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
cosine
L2
N item = 640,000
N key = 2,000 [graded]
0 1 2 3 4 5 6 7 8 9 10
items
b
1 - distance
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
cosine
L2
N item = 640,000
N key = 100 [graded]
0 1 2 3 4 5 6 7 8 9 10
items
c
1 – distance
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
L2
N item = 640,000
N key = 100 [binary]
0 1 2 3 4 5 6 7 8 9 10
items

FIG. 4

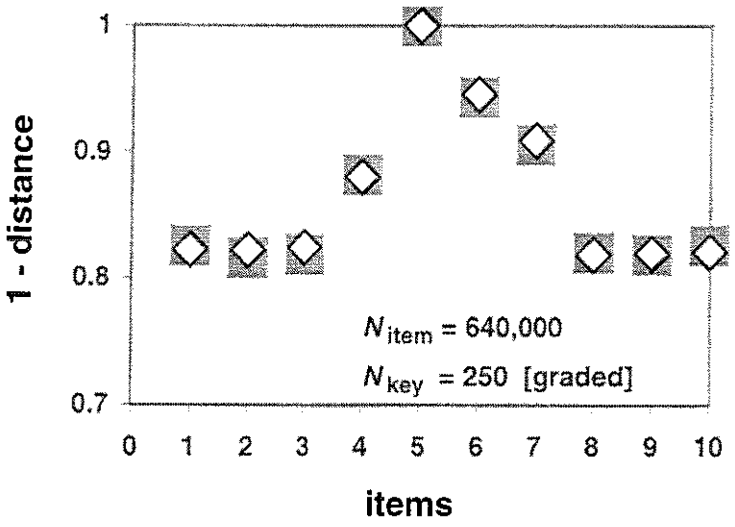
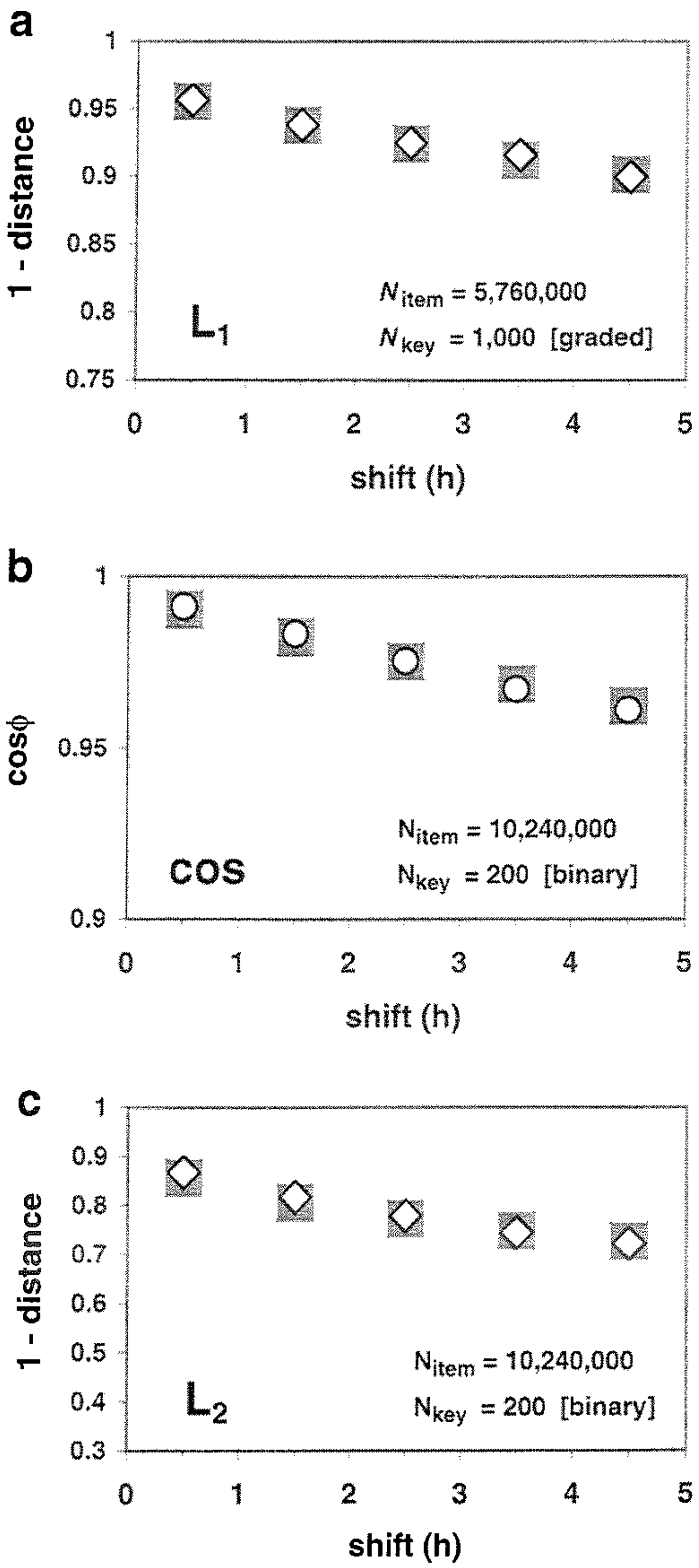

1
0.9
0.8
0.7
1 - distance
N_{item} = 640,000
N_{key} = 250 [graded]
0
1
2
3
4
5
6
7
8
9
10
items

FIG. 5



a
1
0.95
0.9
0.85
0.8
0.75
1 - distance
L$_1$
N_{item} = 5,760,000
N_{key} = 1,000 [graded]
0
1
2
3
4
5
shift (h)
b
1
0.95
0.9
cosφ
COS
N_{item} = 10,240,000
N_{key} = 200 [binary]
0
1
2
3
4
5
shift (h)
c
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
1 - distance
L$_2$
N_{item} = 10,240,000
N_{key} = 200 [binary]
0
1
2
3
4
5
shift (h)

FIG. 6