CEDRIC Research Report n^o 1892

On Estimating the Indexability of Multimedia Descriptors for Similarity Searching

Stanislav Barton
CNAM/CEDRIC
292, rue Saint-Martin
F75141 Paris Cedex 03
stanislav.barton@cnam.fr

Valerie Gouet-Brunet
CNAM/CEDRIC
292, rue Saint-Martin
F75141 Paris Cedex 03
valerie.gouet@cnam.fr

Marta Rukoz
POND University
200 Av. de la Republique
92001 Nanterre, France
mrukoz@yahoo.com.mx

Christophe Charbuillet
IRCAM
1, place Igor-Stravinsky
75004 Paris
christophe.charbuillet@ircam.fr

Geoffroy Peeters IRCAM
1, place Igor-Stravinsky
75004 Paris
geoffroy.peeters@ircam.fr

March 11, 2010

Abstract

A study on properties of data sets representing public domain audio and visual content and their relation to their indexability is presented. Data analysis considers the pairwise distance distributions and various techniques to estimate the true intrinsic dimensionality of the studied data. One own alternative to dimensionality estimation is also presented. These results are contrasted with the indexability results gathered using indexing techniques M-Tree, LSH and hierarchical k-means tree.

1 Introduction

In order to make the multimedia data searchable by its content, various methods of mapping the multimedia content into high-dimensional spaces have been introduced for images [7] and audio [10]. The similarity search by content using this feature data is then transformed into a search for *close* points in the feature space. The distance between points can be measured using various types of functions. The dimensionality of such feature space is called an embedding dimensionality.

Due to known problem of *the curse of dimensionality* – the complexity of search degrades exponentially with growing dimensionality of the feature space – the problem of indexing such data is still under heavy research. Firstly, the research supposed that the complexity of searching such data grows exponentially with its embedding dimensionality [3], but later a tighter relation between the complexity of making such data well searchable and data's intrinsic dimensionality was found [13]. The intrinsic dimensionality of data refers to a real *degree of freedom* of data within the feature space.

Therefore, in this paper we present a comparative study of the properties of multimedia data sets representing visual and audio descriptors acquired from the public domain content provided by EWA [1] and contrast them with its *indexability* – the comparison of general performance of indexing techniques on this data with a sequential scan.

The study is carried out in terms of pairwise distance distribution and mainly of the estimation of the intrinsic dimensionality where among other techniques one own is presented. Using the same methodology and criteria and by comparing the results, we would like to depict the different characteristics of these two types of multimedia contents considering also data sets where the characteristics is known.

The indexability is tested using indexing structures M-Tree, hierarchical k-means tree and LSH. Because it was previously observed that the intrinsic dimensionality and pairwise distance distribution have great correlation with the indexability of data, the results of the study of the data sets are verified by their indexability results.

The conclusions of this study form the foundations for extensible indexability prediction of arbitrary data based on the proposed data analysis. The prediction of the indexability from the data's structure was already tested for R-Trees in [13] via fractal dimensionality. This method relies heavily on a vector space and thus its extensibility to general metric spaces is very limited. Some of the techniques, like [6, 16] rely on automated empirical tests of the performance on particular data and the rest usually rely on a human supervision needed to estimate the best parameters for particular data.

The paper is structured as follows. Firstly, the recapitulation of investigated data is presented in Section 2. Secondly, Section 3 contains the comparative study of the data sets properties followed by Section 4 where several indexing structures on this data are evaluated.

2 Multimedia Contents and Descriptors Considered

Table 1(a) summarizes the multimedia contents and descriptors that were subject of our study, respectively described in Sections 2.1 and 2.2. Besides these

[1] European Web Archive (EWA, http://www.europarchive.org/) is an open archive that hosts collections of public domain content crawled from publicly available resources.

principal data sets, some auxiliary data sets have been also included in order to better understand and interpret the results of the study (Section 2.3).

2.1 Content Descriptors Considered

2.1.1 Global Visual Descriptors

The *global descriptors* resume in one feature vector all the image content. They have the advantage of encapsulating some global semantics or ambiance such as *indoor* or *painting*, while requiring a low amount of data to describe it. Despite the simplicity, such family of descriptors was evaluated as relevant for content-based information retrieval applications [8]. The visual features included in the MPEG-7 standard consist of histogram-based descriptors, spatial color descriptors and texture descriptors [15].

In this study, we use color histogram which counts the proportion of each color in the image[18]. The color space chosen is classical RGB (for Red, Green and Blue). Because a 24 bytes image is able to store more than 17 millions of colors, a discretization of the space is required. For descriptor of ID 1 in Table 1(a) , we have respectively considered (5,5,5) bits for the three RGB channels, leading to a $5^3 = 125$ dimensional feature vector. Descriptor of ID 2 consists of $7^3 = 343$ dimensions. The associated similarity measure is L_2.

2.1.2 Local Visual Descriptors

When considering sub-image retrieval or object recognition, a *local* representation of the image content is usually more appropriate. In particular, it allows to gain robustness against occlusions and cluttering. Traditionally, the sites of description considered are points of interest (often called key or salient points), described by a set of features encapsulating visual information locally around each point [21].

In this work, we use the popular SIFT local descriptor [14], of ID 9 in Table 1(a). SIFT was designed to be invariant to image translation, rotation, scale and is robust to viewpoint changes and localization errors. The associated point descriptor encapsulates the local orientations of pixel's gradients in the neighborhood of the point into a histogram of 128 dimensions. Differently to the global approaches, such a descriptor provides several hundreds or thousands of feature vectors per image. The associated similarity measure is L_1 or L_2.

2.1.3 Global Audio Descriptors

In this study, we have used global audio descriptors which represent the time evolution of short-term audio descriptors using a spectral decomposition [17].

We have selected four different short-term audio descriptors: 13 Mel-Frequency Cepstral Coefficients (which represent the signal spectral shape), 12 Chroma/ Pitch-Class-Profile coefficients (which represent the signal tonal content), 8 Spectral Flatness/Crest Coefficients (which represent the signal harmonic or noise content in several frequency bands) [12]. The short-term audio features are extracted on a frame-base using a 20ms analysis window and 10ms hop size, resulting in a 33 dimensional temporal sequence. The temporal evolution of each

(a)

ID	Descriptor type	Dim.	Distance
Global Visual Descriptors Data Sets			
1	RGB global histogram	125	L_2
2	RGB global histogram	343	L_2
Local Visual Descriptor Data Set			
9	SIFT	128	L_2
Global Audio Descriptors Data Sets			
11	Short Term Descriptor	132	L_2
12	Short Term Descriptor	330	L_2
Synthetic Data Sets			
101	Random Uniform	125	L_2
102	Random Uniform	343	L_2
103	Random Clustered	125	L_2
104	Random Clustered	343	L_2
Adopted Data Sets			
201	ISOMAP Face Data Set	4096	L_2
202	Animal Data Set	72	L_2

(b)

ID	# Clips	# Frames	# Vectors
A	1	1,000	1,000*
B	10	1,000	10,000
C	100	100	10,000
D	1000	10	10,000
E	10,000	1	10,000

Table 1: (a) Summary of descriptors that have been considered for evaluation. (b) Sample selection summary. *SIFT sample A has 10,000 vectors.

dimension is then modeled using its amplitude spectrum. The dimensionality of this decomposition is reduced by grouping frequencies into frequency bands (filter bank). In this study, we have considered to different filter-banks: firstly, rectangular filters centered on $[0, 1-2, 3-15, 20-43]Hz$, referred as ID 11 in Table 1(a), its dimensionality is 132. Secondly, 10 rectangular filters, equally distributed in $[0, 43]Hz$, referred as ID 12 in Table 1(a). Its dimensionality is 330. The resulting signal is then converted to a logarithmic-scale in order to be amplitude independent.

2.2 Visual and Audio Data Sets Considered

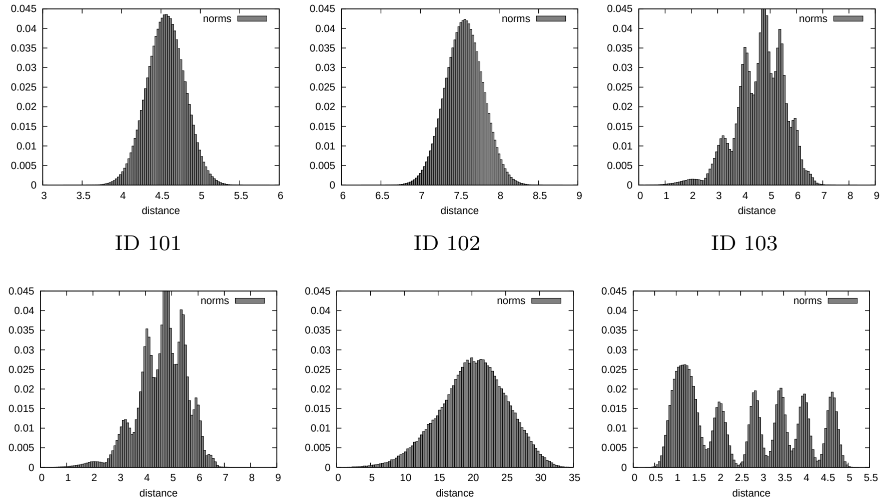
As was mentioned earlier, the data sets were acquired processing the public domain content provided by EWA. In the case of visual data sets, about 2,000 hours of video were processed, computing the RGB global descriptor from one frame every two seconds. Thus, 10,000 videos were processed and to about 3,500,000 frames the feature vectors were extracted. Data sets ID 1 and 2 differ by bits per color resulting in different dimensionality.

The audio descriptors (ID 11 and 12) were extracted from 10,000 musical audio files, totalizing 927 hours of signal. The temporal modeling was performed on a 3s window with a shift of 0.5s, producing 6,674,400 feature vectors.

2.2.1 Sample Selection Method

The criteria evaluation is often computationally intensive task that makes infeasible to use on the input the whole acquired data set – millions of objects. This fact brings out the necessity to select sample from the whole data set on which the evaluation will be done. In order to avoid biased results caused by improper or superficial sample selection, a set of sample data sets have been selected for each studied data set.

The samples are denoted using capitalized letters to distinguish from the IDs of the data set and are summarized in Table 1(b). The sample selection method takes into consideration the scale of redundancy in the sample. For each data set type (audio or video) the clip and frame selection was identical for



ID 101 ID 102 ID 103

ID 104 ID 201 ID 202

Figure 1: Pairwise distance distributions of the auxiliary data sets.

both dimensionalities. As for the SIFT data set, since there is 0 to 500 SIFTs per frame, 10,000 SIFTs per sample have been randomly selected.

2.3 Auxiliary Data Sets and Descriptors

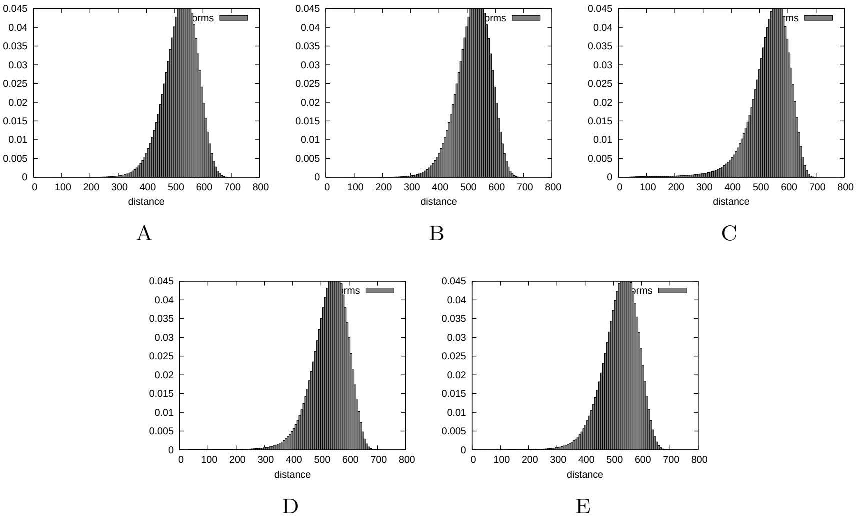
Synthetic floating type data sets ID 101 and 102 with predefined number of dimensions and uniformly distributed vectors were randomly generated. The values are ranging in an interval $[0, 1]$. The pairwise distance distribution is normal.

As for data sets ID 103 and 104, the cluster centers are placed in the corners of the hyper-cube. The data sets have 50 clusters with 200 objects each. The dimensions utilized by cluster centers is $log_2(50) = 5.64$, the objects utilize all of possible dimensions. The clusters are overlapped, the cluster centers are 2.3 far from each other and the cluster diameter is 2.5.

ISOMAP face data set ID 201 is a data set of vectors representing synthetic faces used for evaluation of ISOMAP dimensionality reduction algorithm [19]. The data set consists of 698 vectors of linearized images (256 gray levels) of size 64 x 64 of the synthetic face where the rotation of the face and the lighting varies. The last auxiliary data set ID 202 is clustered and represents four kinds of quadruped animals where each is described by a 72-dimensional vector – geometrical description of one particular animal. It contains four clusters each having 2,500 feature vectors. This data set was used for classification methods evaluation for instance in [9].

3 Data Analysis

Data is analyzed in terms of pairwise distance distribution (Section 3.1) and of intrinsic dimensionality estimation (Section 3.2).



A B C

D E

Figure 2: Pairwise distance distributions of the visual descriptor data set ID 9, samples A, B, C, D and E.

3.1 Pairwise Distance Distribution

The pairwise distance distribution gives an insight into the organization of the distances among the feature vectors in the data set. With comparison to the auxiliary data sets, whose structure is known, the overall structure of the data is discovered. Because some of the following methods rely on the pairwise distance distribution histogram it is necessary to recapitulate this matter.

In [1], is published the qualitative study of the pairwise distance histograms of the global visual and audio and the auxiliary data sets ID 101, 102, 201 and 202 data sets. Fig. 1 depicts the pairwise distance distributions of the auxiliary data sets to recall the various shapes of the histograms. Fig. 2 depicts the samples of SIFT data sets which was not subject of the study in [1].

The main conclusions of the qualitative study is that neither the visual nor the audio data sets are comprised of distinct clusters. Yet, the visual data sets seem to be comprised of heavily overlapped large clusters and the audio data sets are structurally more closer to the uniform random data sets. The last important conclusion of this study for the visual global descriptor is that adding more than 200 dimensions to the ID 2 data set did not bring any finer granularity as in the case of the data set ID 1.

As for the histograms of SIFT descriptor in Fig. 2, the fact that it is a local technique resulting into multiple feature vectors for one processed image stands behind the fact that the sample technique did not reveal any redundancy in the description – vectors closer to each other – even for the A sample.

3.2 Intrinsic Dimensionality Estimation

The actual dimensionality of the data needs not necessarily reflect the intrinsic dimensionality of the data. In fact, according to the common sense, most of the real life data is an embedding of a lower dimensional data to the higher dimensional space. This is the main reason why it is still possible to index such high dimensional data because it was discovered that the indexability of data depends rather its intrinsic dimensionality than on the dimensionality of the embedding. Therefore, in this section the estimation of the intrinsic dimensionality of the data sets is presented using five approaches.

The selection criteria of the approaches were mainly the simpleness of the estimator and its possible application on domains with non-Minkowski distance functions. Since the results of the estimators vary, literature was studied in order to set a base line for the estimations. Besides several data sets with known estimations like the data sets ID 201, in [22] is stated that the data sets with normal distribution of distances, i.e. data sets ID 101 and 102, have intrinsic dimensionality very close to its embedding dimensionality. Therefore we use those to justify the estimations.

3.2.1 Principal Component Analysis

PCA [20] is a statistical method that gives an insight into the internal structure of the data through its eigenvalue decomposition. It transforms the original vector data into lower dimensional data that respects its variance. Only the components of the eigenvalue decomposition, that significantly contribute to the data's energy (cumulative sum of the eigenvalues) are kept. Though, for each data set minimal number of components needed to achieve the 95% of the energy of data was computed.

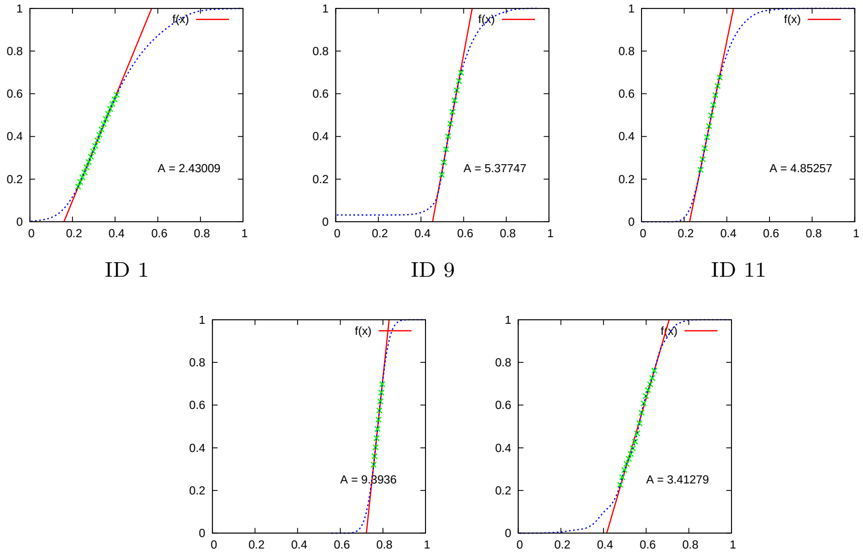
3.2.2 kNN Intrinsic Dimensionality Estimator

To estimate the intrinsic dimensionality, the estimator (kNN-IDE) described in [5] utilizes the notion of k-NN graph and its total length. The k-NN graph (kNNG) puts an edge between each point in the data set ($\mathcal{X}$) and its k-nearest neighbors with the distance between these points as the edge's weight. Let $\mathcal{N}_{k,i}(\mathcal{X})$ be the k-nearest neighbors of point $\mathbf{X}_i \in \mathcal{X}$, the total length of kNNG is defined as follows: $\hat{L}_\gamma(\mathcal{X}) = \sum_{\mathbf{X_i} \in \mathcal{X}} \sum_{\mathbf{X} \in \mathcal{N}_{k,i}(\mathcal{X})} d^\gamma(\mathbf{X}, \mathbf{X}_i)$, where d represents the distance function.

The authors in [5] found and proved the strong dependence of the length of the kNNG to the intrinsic dimensionality. Therefore they stated a simple estimator of it m: $\log \hat{L}_\gamma(\mathcal{Y}_n) = a \log n + b + \epsilon_n$ where $\hat{L}_\gamma(\mathcal{Y}_n)$ is a total length of the kNNG of a uniform sample $\mathcal{Y}_n$, $a = (m - \gamma/m)$ and γ is power weighting constant, in our case $\gamma = 1$, b represents the entropy of the data set and for the estimation of the intrinsic dimensionality is not necessary, ϵ_n is an error residual. a and b are approximated using several bootstrapping samples $\mathcal{Y}_n$ and using the method of moments and linear least squares. For our estimations we have used the same parameters as the authors. Each result of was rounded to the next greater integer and ten estimations were averaged to get the final estimation.

3.2.3 PDD Histogram Intrinsic Dimensionality Estimator

In order to estimate the intrinsic dimensionality, the authors in [2] use the pairwise distance distribution (PDD) histogram. Their definition of intrinsic dimensionality ρ is $\rho = \frac{\mu^2}{2\sigma^2}$, where μ denotes the particular histogram's mean and σ^2 variance. The idea behind this concept is that with the dimensionality



ID 1 ID 9 ID 11

ID 101 ID 103

Figure 3: Normalized CDD histograms (blue) with fitted slope function $f(x)$ (red) with marked points used for fitting (green) and estimated value of parameter a.

the mean grows. Also the high intrinsic dimensionality goes hand in hand with low variance.

3.2.4 Fractal Correlation Dimensionality

Given a E-dimensional space, we can divide the space into hyper-cubic grid cells of side r. The definition of Fractal Correlation Dimensionality (FCD) according to [13] is $D_2 \equiv \frac{\partial \log \sum_i p_i^2}{\partial \log r}$ for $r \in (r_1, r_2)$, where p_i is the percentage of points which fall inside the ith cell.

The actual estimation is done by estimating the slope of the curve representing the box count plot of the D_2 values for specified range (r_1, r_2). Yet, the condition on the data given is that it has to be *self-similar*, i.e. the data from view in great or low resolution exhibit same patterns, for detailed definition see [13]. Yet, this property has most of the real life data.

3.2.5 Cummulative Distribution Slope (CDS)

This method is a result of the analysis of the pairwise distance and cummulative distance distribution histograms. We have developed to provide own design of the intrinsic dimensionality estimation. It follows the same ideas as the previous method by taking into consideration the number of neighbors within certain range. Contrary to the FDD which relies on a vector space and uses the coordinates to make the estimation, our method considers only the distances between the vectors and thus is also extensible to metric spaces where the data object are not necessary vectors.

The computation is done on a normalized cummulative distance distribution (CDD) histogram. It differs from the usual CDD histogram by normalized x-axis by the largest distance between any two points encountered during the computation. Thus all the values x-values range from 0 to 1. In this histogram, the flat parts are removed and a line representing linear function $f(x) = ax + b$ is fitted using linear least squares method. In Fig. 3 are depicted the normalized CDD histograms (blue) with fitted slope function $f(x)$ (red) with marked points used for the fitting (green) and estimated value of parameter a. The value of the parameter a forms the data set's difficulty of indexability estimation.

Recall that the intrinsic dimensionality represents the degree of freedom of the data where the uniformly distributed random data using all available dimensions maximizes this freedom. Table 2 contains values of a for the uniform random data sets with varying dimensionality. Using this table, the value of parameter a can be mapped onto a domain representing the data's intrinsic dimensionality. Thus the final CDS estimation values are comparable with the results of the other estimation methods.

Dimensionality	2	3	5	10	15	30	60	125	343
Value of a	1,37	1,92	2,74	3,65	4,28	5,51	6,99	9,39	14,14

Table 2: Values of a for uniform random data sets with varying dimensionality.

3.2.6 The Results

The results of the intrinsic dimensionality estimations are summarized in Table 3. To interpret them, the intrinsic dimensionality estimations of the auxiliary data sets ID from 101 to 201 need to be explained at first. From [11] is known that the intrinsic dimensionality, in other words the degree of freedom, of the data set ID 201 is three. Estimation using the kNN-IDE is 4.2, a slight discrepancy might been introduced due to a different rounding method of the implementations. However, using PCA leads into significant overestimation of the intrinsic dimensionality. FCD method was very close and the CDS estimated the intrinsic dimensionality to be 5. Recall that the data set ID 201 has 4096 dimensions.

As for the data sets ID 101 and 102, the intrinsic dimensionality is very close to the actual dimensionality of the data. The PCA estimated correctly the intrinsic dimensionalities since all the dimensions are mutually independent. However, using kNN-IDE lead to significant underestimations of the intrinsic dimensionality. On the other hand the FCD method, and in even greater extent the PDD-IDE, overestimated the dimensionality. The PDD-IDE overestimation is caused by the non-normal distribution of the pairwise distances. Since the CDS method uses the estimations of these data sets for the mapping of the slope fitting, the values are exact.

The true intrinsic dimensionality of synthetic clustered data sets ID 103 and 104 is not known but due constraints on the distance between each point in space and its assigned cluster center, the true degree of freedom should be lower than of similar synthetic datasets ID 101 and 102. The estimations are listed for the validation purposes of conclusions of the data sets indexability. For same reasons the estimations are listed also for adopted clustered data set ID 202.

ID	*k*NN-IDE					PCA					PDD-IDE					FCD					CDS				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
1	3.5	4.3	4.9	7.2	10.6	3	8	15	25	25	1.2	2.6	2.9	2.7	2.7	2.1	2.1	3.2	2.9	3	7.6	3.7	4.9	4.3	4.2
2	3.8	4.0	5.6	9.2	11.6	4	11	26	43	48	1.8	3	3.1	2.7	2.7	2.1	2.3	4.2	3.5	3.6	6.3	4.7	7.9	5.4	4.8
9	15.4	17.8	8.6	12.1	14.9	76	76	67	73	75	38.8	38.8	26.4	34.4	34.8	59	59	34.2	47.3	41.4	21.2	21.2	19.6	20	19.9
11	3.7	6.3	12.2	15.4	25	46	44	47	50	50	7.2	8.2	8.2	9.7	7.8	6.3	15.6	23.4	25	22.6	14.3	13.2	20.7	19.3	22
12	4.5	8.3	13.8	17.8	27	128	141	154	169	157	12.8	9.3	9.2	11.3	8.7	0.6	9.1	5.7	3.4	8.3	16.9	15.8	22.9	25.7	25.2
101	52.4					118					177.8					132.8					125				
102	92.8					319					489.9					457.8					343				
103	7.6					85					12.8					15.2					8.7				
104	8.6					222					13.2					19.41					9				
201	4.2					59					8.3					2.4					5				
202	12.1					11					1.9					12					4.9				

Table 3: Intrinsic dimensionality estimations.

The main observation for the pairs of data sets that share the same description technique and differ in resulting embedding dimensionality – ID 1 and 2 and ID 11 and 12 – show that the intrinsic dimensionality do not grow proportionally to the growth of the embedding dimensionality, besides the PCA method that was found to be the less suitable for these purposes. This observation yields the conclusion of the pairwise distance distribution analysis that the greater dimensionality does not provide the user with equally greater descriptiveness. This means that the computational overhead resulting from the larger vector representation does not redeem the better description of the particular data object. This is the main reason why in the following indexability evaluation of the data sets only the smaller dimensionality data set from each pair was considered.

Notable are very inconsistent estimations of FCD method on data sets ID 11 and 12 caused probably by the absence or weak presence of the self-similarity property.

We have studied both, the estimations on data sets whose degree of freedom is close to the actual dimensionality and data sets which represent an embedding of some data with much smaller dimensionality. Then, even though the intrinsic dimensionality of data sets ID 1 – 12 is not known, the mutual relations between the individual estimations needs to be taken into consideration while evaluating the particular method. This will be discussed in Section 4.4 where the results of the indexing methods are presented.

4 Indexability

In order to test the ability of the various multimedia data sets presented in this paper of being indexed, various indexing techniques were deployed. The mission of the tests presented in this section is not the mutual comparison of the performance of the indexing structures presented here but the comparison of how a particular indexing approach perform on the set of provided data sets and the comparison of such results. The implementations were provided by the respective authors. They differ by the programming language used for implementation and thus the mutual comparison by running times, which is the only available measurable value, is not possible.

For evaluation, five data sets having the dimensionality between 125 and 130 were used – ID 1, 9, 11, 101 and 103. For each descriptor, two large samples were used, one having 500,000 and the second 1,000,000 data points randomly selected from the whole data set, see Section 2.2. The data sets ID 2 and 12 were not considered for evaluation for the previous conclusions made about its descriptiveness.

4.1 E2LSH Package

The indexing structure in this package[2] solves the r-NN queries using the locality sensitive hashing (LSH) approach presented for instance in [6] for Euclidean distance. The r-NN query can be envisioned as a probabilistic approximate range query where r represents radius. The indexing structure is built to process range queries, where the probability of missing a data point that lies within the range r of query point q is lower or equal to predefined P. This probability was set to 0.1. In other words, the structure retrieves at least $1 - P$ of true data points lying within range r from query point q.

The LSH indexing approach transforms the original data set using locality sensitive hashing functions to easier searchable domain, i.e., integer numbers. When processing a query, it checks only those data points that collide with the query point after hashing. In order to reduce the number of collided data points, more hashing functions are used at once.

There are two necessary parameters to create the indexing structure, L the number of hashing functions and k denoting a k-tuple each of the hashing functions produces. They are computed as a function of a data set, a set of query points to minimize the query time considering the probability P and the amount of available main memory. This auto-tuning method computed for the 500,000 data sets $k = 16$ and $L = 231$ and for the 1 million data sets $k = 12$ and $L = 91$.

4.2 M-Tree Access Structure

As a representative of access structures for metric spaces, a popular tree structure called M-Tree[3] [4] was selected. The motivation to evaluate the data sets on this indexing structure is the possibility to use any metric function as a measure of the distance between objects, because the vector space access structures rely mostly on the use of the Minkowski distances (L_p metrics, where $p \in \{1, 2\}$). Even though, the distance between data points in the presented data sets is measured using L_2 distance, this might change in the future.

The M-tree is inspired by B^+-trees. In the internal nodes, it stores predefined number of objects and the covering radius of all objects that are on lower levels below the particular *pivot*. The structure is balanced since it is built bottom-up by splitting its fixed sized nodes. M-Tree is able to process both kNN and range queries in both approximated and exact variants. Yet, considered in this paper is only the exact range query evaluation.

Two parameters reflecting the capacities of the internal and leaf nodes are set to 30 and 100, respectively. They are common for all evaluation runs of all data sets and sizes. They were chosen from an empirical evaluation that showed that the variance of those parameters do not affect the evaluation speed much.

4.3 FLANN Library

This library represents a framework for automatic indexing structure selection together with proper parameter setting[4]. The scientific ideas behind this framework were published as [16]. In the current version (1.2), the library selects

[2] http://www.mit.edu/~andoni/LSH/

[3] http://lsd.fi.muni.cz/trac/mtree/

[4] http://people.cs.ubc.ca/~mariusm/index.php/FLANN/

	JAVA				**C++**									
	Seq Scan		**M-Tree**		**Seq Scan**		**E2LSH**		**FLANN**					
ID	**CT**	**AQT**	**CT**	**AQT**	**CT**	**AQT**	**CT**	**AQT**	**CT**	**AQT**	**Rec.**	**Bra.**	**It.**	**Che.**
500,000 data points														
1	24	2.102	698	0.282	19.3	0.119	120.1	0.017	59	0.051	0.79	64	1	192
9	21	2.411	736	1.549	17.6	0.122	123.4	0.063	152	0.057	0.68	16	15	1472
11	33	2.396	776	1.298	35.3	0.125	124.8	0.123	160	0.059	0.64	64	15	168
101	33	2.468	760	1.706	31.7	0.119	124.9	0.322	691	1.543	0.66	256	10	172032
103	36	1.957	750	0.775	34.0	0.119	118.8	0.035	42	0.056	0.71	16	5	1344
1,000,000 data points														
1	44	3.991	1,607	0.702	39.8	0.238	102.0	0.037	347	0.046	0.75	64	5	168
9	42	4.871	1,659	3.162	35.3	0.245	104.5	0.195	144	0.058	0.69	16	5	1344
11	65	4.878	1,813	2.635	70.5	0.250	89.7	0.124	371	0.064	0.57	32	10	1280
101	43	4.863	2,047	3.533	64.0	0.238	110.0	0.643	694	2.001	0.56	64	15	200704
103	47	4.020	1,677	1.429	64.2	0.238	103.3	0.070	92	0.051	0.59	16	1	62

Table 4: Indexability results where CT denotes time to create the index and AQT Average Query Time, according to the implementation.

between two access structures (k-means hierarchical tree, randomized k-d trees) on the basis of a provided data set and query set in order to enable the user with a approximate kNN queries with certain predefined precision.

In our case, whatever the data set, the auto-tune mechanism never recommended the randomized k-d trees so only the parameters regarding the hierarchical k-means access structure are discussed. Two parameters affect the creation phase of the structure: the *branching* factor denotes the number of possible branches of one node in the tree; the number of *iterations* states a stop condition for the clustering step of the k-means algorithm rather than the traditional convergence condition. In order to achieve the desired target precision during the search, the number of *checks* denotes the amount of tree traversals during the search algorithm.

The k-means tree is built by splitting the data points at each level into k regions and then recursively applying the same method within the regions. The recursion stops when the number of points in a region is smaller than k. The search algorithm utilizes a priority queue for most promising tree traversals.

4.4 The Results

Table 4 summarizes the measured running times of the considered indexing methods. The time that the particular approach took to create the indexing structure and an average time to process one query were measured. In the case of the range queries, a diameter for each data set was found to retrieve on average 7,000 objects from the 500k data sets for the randomly selected set of queries. For the kNN queries, k was 7,000. For evaluation a machine with two 2.0GHz XEON processors with two cores each and 8 GB of RAM was used. The indexing structures were evaluated in terms of the time it took to create the index (CT) and the average time to process one query (AQT) both presented in seconds.

In Table 4 the time to create the index for sequential scan represents the time it took the algorithm to read the data from disk into the main memory. The various times measured are due to a fact that the data are stored in a text files whose sizes vary from data set to data set. Notice that some of the implementations are in JAVA and some in C++ and because significant differences between the speed of the platforms were encountered, sequential scan results using both languages are presented.

	Normalized Intrinsic Dimensionality					*Rank*					
ID	***k*NN-IDE**	**PCA**	**PDD-IDE**	**FCD**	**CDS**	**M-Tree**		**E2LSH**		**FLANN**	
						500k	1,000k	500k	1,000k	500k	1,000k
1	0.2	0.21	0.02	0.02	0.03	1	1	1	1	1	1
9	0.28	0.64	0.2	0.31	0.16	4	4	3	4	3	3
11	0.48	0.42	0.04	0.17	0.18	3	3	4	3	4	4
101	1	1	1	1	1	5	5	5	5	5	5
103	0.15	0.72	0.07	0.11	0.07	2	2	2	2	2	2

Table 5: Intrinsic dimensionality estimation and ranked indexing results juxtaposed.

The average query time of M-tree for data sets 1 and 103 are with considerable gain against the sequential scan. Due to its nature, it was anticipated that the M-Tree will perform bad on the data set 101. The covering radii of the internal nodes must significantly overlap which causes the traversal of the majority of the leaf nodes.

Recall that the E2LSH package enables the processing of approximate range queries with assured lower bound on recall – in this case to 0.9. Even though the parameter tuning was ran for each data set only two settings were identified – one for the 500,000 and one for 1,000,000 vectors data sets. This means that the results measured are comparable for the particular data set size for all data set types. Despite the different settings, the AQT roughly doubled for each data set type with the transition to the larger sample, only in the case of data set ID 9, the speed tripled and in the case of ID 11 remained almost the same.

Rather minor differences between the average query times were measured using the FLANN library and the hierarchical k-means tree. These *flat* average query times were acquired by tuning the parameters for each particular indexed data set. Yet, the assured recall was not met after comparing the approximated query answer to the exact one. This is due to the fact that the automatic tuning method works on subsamples of the data set in order to speed up the tuning process which propagates an error into the estimation. Again, proper setting was not found for the data set ID 101 since it gives by two orders of magnitude worse results than on the other data sets. The three most important parameters (*branching*, *iterations* and *checks*) are also listed to give the reader the insight about the shape of the built trees.

4.5 Estimating the Indexability

In order to estimate the indexability of the the particular data set the results of the estimations with the results of the indexability are contrasted. The intrinsic dimensionality estimations and the indexing results are juxtaposed in Table 5. The intrinsic dimensionality estimations are normalized by the estimation for the data set ID 101 and instead of reporting the average query time for particular index, the rank of the speed of the query processing is depicted. The observations of the results of the indexing structures that have similar settings for all the data sets make possible to derive the relationship between their intrinsic dimensionality and their indexability.

Considering the kNN-IDE method, the estimation of data set ID 1 and ID 103 are in contradiction, since the indexability of the former is definitely the best. This is also the case of the PCA method that even identified the data set ID 103 as being worse indexable than the data sets IDs 9 and 11. Similar underestimation did PDD-IDE method for the data set ID 103 and 11.

The indexability of the data sets IDs 9 and 11 needs to be analyzed more thoroughly. The M-tree handled the ID 11 data set faster in both data set sizes. The k-means tree handled both for the 500,000 data set size similarly, considering the AQT and the recorded recall, yet, a significant difference arose in the case of the greater data set size. Similar discrapancy was recorded for the E2LSH package. From this, it can be concluded that the intrinsic dimensionality of ID 11 is very similar to that of ID 9 but cannot decide which one is greater. Yet, other structural differences are behind such performance fluctuation – which is caused most probably by the larger mean of the pairwise distance distribution of ID 9, see [1] or Fig. 3. The least difference of estimations for those two estimation was provided by the CDS method.

From this analysis, the most reliable method to relate the intrinsic dimensionality estimation to the indexability is the developed CDS method which keeps both correct order of the data sets according to their difficulty for indexing and the scale of the estimation.

5 Concluding Remarks

Many works foreshadowed that there is a high correlation between the intrinsic dimensionality and the following indexability of the data. In this paper we have evaluated and juxtaposed several approaches to estimate the intrinsic dimensionality. Also one own design to estimate this value was presented.

After the general insight into the internal structure of the data was presented, the indexability of the multimedia data sets and also some auxiliary data sets was tested by various approaches to indexing high-dimensional data for similarity searching and for two of them, self-parameter tuning was used to avoid the performance degradation.

The overall analysis showed that the highest correlation of the intrinsic dimensionality and the indexability of the studied data sets exhibited the newly introduced CDS method.

Acknowledgements

This work is done within the French Federation WISDOM (2007-2010) and is supported by the French project ANR MDCO DISCO (2008-2010).

References

[1] Stanislav Barton, Valerie Gouet-Brunet, Marta Rukoz, Christophe Charbuillet, and Geoffroy Peeters. Qualitative comparison of audio and visual descriptors distributions. In *MCIT'10*, pages 1–4, Sharjah, UAE, 2010.

[2] E. Chávez and G. Navarro. A probabilistic spell for the curse of dimensionality. In *ALENEX '01: Revised Papers*, pages 147–160, London, UK, 2001. Springer-Verlag.

[3] B. Chazelle. Computational geometry: a retrospective. In *Proc. of ACM STOC*, pages 75–94, 1994.

[4] Paolo Ciaccia, Marco Patella, and Pavel Zezula. M-tree: An efficient access method for similarity search in metric spaces. In *VLDB*, pages 426–435, 1997.

[5] J. A. Costa and A. O. Hero. Geodesic entropic graphs for dimension and entropy estimation in manifold learning. *Signal Processing, IEEE Transactions on*, 52(8):2210–2221, 2004.

[6] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *SCG'04: Proceedings of the twentieth annual symposium on Computational geometry*, pages 253–262, New York, NY, USA, 2004. ACM.

[7] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2), 2008.

[8] T. Deselaers, D. Keysers, and H. Ney. Features for image retrieval: an experimental comparison. *Inf. Retr.*, 11(2):77–107, 2008.

[9] J. H. Gennari, P. Langley, and D. Fisher. Models of incremental concept formation. *Artif. Intell.*, 40(1-3):11–61, 1989.

[10] Theo Gevers, Graham D. Finlayson, and Raimondo Schettini. Audio information retrieval: a bibliographical study. 2002.

[11] M. Hein and J.-Y. Audibert. Intrinsic dimensionality estimation of submanifolds in rd. In *ICML*, pages 289–296, 2005.

[12] J. Herre, E. Allamanche, and O. Hellmuth. Robust matching of audio signals using spectral flatness features. pages 127–130, 2001.

[13] Flip Korn, Bernd-Uwe Pagel, and Christos Faloutsos. On the 'dimensionality curse' and the 'self-similarity blessing'. *IEEE Trans. on Knowl. and Data Eng.*, 13(1):96–111, 2001.

[14] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[15] B. S. Manjunath, P. Salembier, and T. Sikora. *Introduction to MPEG-7: Multimedia Content Description Interface*. Wiley & Sons, April 2002.

[16] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISSAPP (1)*, pages 331–340, 2009.

[17] G. Peeters, A. Laburthe, and X. Rodet. Toward automatic music audio summary generation from signal analysis. In *Proc. of ISMIR*, pages 94–100, Paris, France, 2002.

[18] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, pages 11–32, November 1991.

[19] J. B. Tenenbaum, V. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, December 2000.

[20] M.E. Timmerman. Principal component analysis (2nd ed.). i. t. jolliffe. *Journal of the American Statistical Association*, 98:1082–1083, 2003.

[21] Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: a survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, January 2008.

[22] Peter J. Verveer and Robert P.W. Duin. An evaluation of intrinsic dimensionality estimators. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):81–86, 1995.