# USING MUTUAL PROXIMITY TO IMPROVE CONTENT-BASED AUDIO SIMILARITY

**Dominik Schnitzer**[1,2], **Arthur Flexer**[1], **Markus Schedl**[2], **Gerhard Widmer**[1,2]
[1]Austrian Research Institute for Artificial Intelligence (OFAI), Vienna, Austria
[2]Department of Computational Perception, Johannes Kepler University, Linz, Austria
dominik.schnitzer@ofai.at, arthur.flexer@ofai.at,
markus.schedl@jku.at, gerhard.widmer@jku.at

## ABSTRACT

This work introduces Mutual Proximity, an unsupervised method which transforms arbitrary distances to similarities computed from the shared neighborhood of two data points. This reinterpretation aims to correct inconsistencies in the original distance space, like the hub phenomenon. Hubs are objects which appear unwontedly often as nearest neighbors in predominantly high-dimensional spaces.

We apply Mutual Proximity to a widely used and standard content-based audio similarity algorithm. The algorithm is known to be negatively affected by the high number of hubs it produces. We show that without a modification of the audio similarity features or inclusion of additional knowledge about the datasets, applying Mutual Proximity leads to a significant increase of retrieval quality: (1) hubs decrease and (2) the $k$-nearest-neighbor classification rates increase significantly.

The results of this paper show that taking the mutual neighborhood of objects into account is an important aspect which should be considered for this class of content-based audio similarity algorithms.

## 1. INTRODUCTION

A number of audio similarity algorithms which have been published so far are affected by the so called "hub problem" [1, 4, 6, 16]. Hubs are over-popular nearest neighbors, i.e. the same objects are repeatedly identified as nearest neighbors. The effect is particularly problematic in algorithms for similarity search, as the same "similar" objects are found over and over again. In 2010 Radovanović et al. [19] published an in-depth work about hubs, showing that they are yet another facet of the curse of dimensionality. Radovanović also showed that "bad hubs" (objects which are a bad retrieval result, in addition to being a hub) can degrade the retrieval quality of algorithms significantly.

The work of this paper was inspired by these problems and presents a straightforward method to reduce the "hub problem" significantly. In the case of the standard audio similarity algorithm we use in this work we can show how to reduce its number of hubs while simultaneously increasing its retrieval quality.

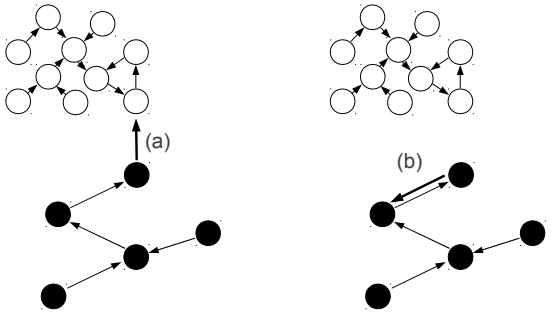


## 2. RELATED WORK

Nearest neighbor search (NNS) is a well defined task: given an object $x$ find the most similar object in a collection of related objects. In the simplest case the problem is solved by a linear search, computing a distance/similarity between $x$ and all other objects, sorting the distances/similarities to return the top $k$-nearest neighbors.

A natural aspect of nearest neighbor relations is that they do not need to be symmetric: that is, object $y$ is the nearest neighbor of $x$, but the nearest neighbor of $y$ is another object $a$ ($a \neq x$). This behavior is problematic if $x$ and $y$ belong to the same class but $a$ does not, thus it is said $a$ violates the *pairwise cluster stability* [3]. Although $a$ is, in terms of the distance measure, the correct answer to the nearest neighbor query for $y$, it may be beneficial to use a distance measure enforcing symmetric nearest neighbors. Thus a small distance between two objects would be returned only if their nearest neighbors concur. Figure 1 illustrates this effect.

Repairing sometimes contradicting, asymmetric nearest neighbor information in a similarity measure was already investigated in a number of works. The first publication which exploits common near neighbor information dates back as far as 1973. Jarvis and Patrick [11] propose a "Shared Near Neighbor" similarity measure to improve the clustering of non-globular clusters. As the name may suggest the Shared Near Neighbor ($SNN$) similarity is based on computing the




(a) Original nearest neighbor relations

(b) Desired nearest neighbor relations

**Figure 1**: Schematic plot of two classes (black/white filled circles). Each circle has its nearest neighbor marked with an arrow: (a) violates the *pairwise stability* clustering assumption, (b) fulfills the assumption. In many applications (b) would be the desired nearest neighbor relation for the dataset.

overlap between the $k$ nearest neighbors of two objects $x$, $y$:

$$SNN_k(x, y) = |NN_k(x) \cap NN_k(y)|. \quad (1)$$

Shared Near Neighbor similarity was also used by Ertöz et al. [5] to find the most representative items in a set of objects. Jin et al. [12] use the Reverse Nearest Neighbor (RNN) relation to define a general measure for outlier detection.

Other work which takes advantage of the asymmetry of nearest neighbors to correct the distance space was performed by Pohle et al., who propose a method named *Proximity Verification* ($PV$) [17]. Two objects are considered similar if both objects have a low nearest neighbor rank according to their counterpart. An unsupervised technique using the local neighborhood of objects to improve the retrieval accuracy of cover song detection systems is proposed by Lagrange and Serrà [13].

An effect of high dimensionality which affects particularly NNS is the hub problem. Berenzweig [4] suspected a connection between the hub problem and the high dimensionality of the feature space. Radovanović et al. [19] were able to provide more insight by linking the hub problem to the property of *distance concentration* in high dimensions. Concentration is the surprising characteristic of all points in a high dimensional space to be at almost the same distance to all other points in that space. It is usually measured as a ratio between spread and magnitude, e.g. the ratio between the standard deviation of all distances to an arbitrary reference point and the mean of these distances. If the standard deviation stays more or less constant with growing dimensionality while the mean keeps growing, the ratio converges to zero with dimensionality going to infinity. In such a case it is said that the distances concentrate. This has been studied for Euclidean spaces and other $\ell^p$-norms. Radovanović presented the argument that in the finite case, some points are expected to be closer to the center than other points and are at the same time closer, on average, to all other points. Such points closer to the center have a high probability of being hubs, i.e. of appearing in nearest neighbor lists of many other points.

Hubs were observed in music information retrieval [2], image [9] and text retrieval [19] making this phenomenon a general problem for information retrieval and recommendation algorithms.

A music similarity algorithm which is adversely affected by the "hub problem" is the method published by Mandel and Ellis [15]. The algorithm is widely seen as a standard method for computing music similarity and its hub problems have already been noticed and investigated (for example by Flexer et al. [6]). The algorithm uses a timbre model computed from the audio signal for music similarity. In its core the basic method stores the music similarity information for each music piece in a single multivariate Gaussian, which is estimated from the Mel Cepstrum Frequency Coefficients [14] (MFCCs) of the audio signal. To compute the similarity usually closed form solutions of Kullback-Leibler related divergences are used.

## 3. AUDIO SIMILARITY

This work uses the basic algorithm from Mandel and Ellis [15] to compute audio similarity. To compute the features we use $25$ MFCCs for each $46ms$ of audio with a $23ms$ hop size. This corresponds to a window size of $1024$ and a hop size of $512$ audio samples at a sampling rate of $22.05kHz$. A Gaussian model is estimated from the MFCC representation of each song so that finally a single timbre model is described by a $25$-dimensional mean vector, and a $25 \times 25$-dimensional covariance matrix. We use the Matlab music analysis (MA) toolbox [1] to compute the features.

To compute the similarity between two timbre models we use a Jensen-Shannon approximation (js), a stable symmetrized version of the Kullback-Leibler divergence from the multivariate normal (MVN) toolbox [2] .

## 4. THE METHOD

In this section we introduce a method that is based on: (i) transforming distances between points $x$ and $y$ into probabilities that $y$ is closest neighbor to $x$ given the distribution of all distances to $x$ in the data base, (ii) combining these probabilistic distances from $x$ to $y$ and $y$ to $x$ via the product rule. The result is a general unsupervised method to

[1] http://www.pampalk.at/ma/

[2] http://www.ofai.at/~dominik.schnitzer/mvn

transform arbitrary distance matrices to matrices of probabilistic *mutual proximity* (MP). The first step of transformation to probabilities re-scales and normalizes the distances like a z-transform. The second step combines the probabilities to a mutual measure akin to shared near neighbor approaches. By supporting symmetric nearest neighbors the method leads to a natural decrease of asymmetric neighborhood relations and as a result, to a decrease of hubs.

### 4.1 Preliminaries

Given a non-empty set $M$ with $n$ objects, each object $m_x \in M$ assigned an index $x = 1..n$. We define MP to be used for a divergence measure $d : M \times M \rightarrow \mathbb{R}$ with the following properties:

- non-negativity: $d(m_x, m_y) \geq 0$,
- identity: $d(m_x, m_y) = 0, \iff m_x = m_y$,
- symmetry: $d(m_x, m_y) = d(m_y, m_x)$.

Individual elements $m_x \in M$ are referenced in the text by their index $x$. The distance between two elements referenced by their index is denoted as $d_{x,y}$.

### 4.2 Mutual Proximity (MP)

In a first step for each element $x$ the average distance $\hat{\mu}_x$ and the standard deviation $\hat{\sigma}_x$ of all its distances $d_{x,i=1..n}$ in $M$ is computed, estimating a Gaussian distance distribution $X \backsim \mathcal{N}(\hat{\mu}_x, \hat{\sigma}_x)$ for each element $x$ (Equation 2). This is based on the assumption that our data is normally distributed due to the central limit theorem. The estimated normal $X$ thus models the spread of distances from $x$ to all other elements in $M$:

$$\hat{\mu}_x = \frac{1}{n}\sum_{i=1}^{n} d_{x,i}, \qquad \hat{\sigma}_x^2 = \frac{1}{n}\sum_{i=1}^{n}(d_{x,i} - \hat{\mu}_x)^2.$$

Figure 2a shows a schematic plot of the probability density (pdf) function which was estimated for the distances of $x$. The mean distance ($\hat{\mu}_x$) is in the center of the density function. Objects with a small distance to $x$ (i.e. objects with high similarity in the original space) find their distance on the left-side of the density function. Note that the leftmost distance in the Gaussian is $d_{x,x} = 0$.

By estimating a normal distribution $X$ from the distances $d_{x,i=1..n}$, it is possible to reinterpret the distance $d_{x,y}$ as the probability that $y$ is the nearest neighbor of $x$, given the distance $d_{x,y}$ and normal $X$ (that is the probability that a randomly drawn element $z$ will have a distance $d_{x,z} > d_{x,y}$):

$$\begin{aligned} P(X > d_{x,y}) &= 1 - P(X \leqq d_{x,y}) \\ &= 1 - \mathcal{F}_x(d_{x,y}). \end{aligned}$$

$\mathcal{F}_x$ denotes the cumulative distribution function (cdf) of the normal distribution defined by $X$. The probability of an element being a nearest neighbor of $x$ increases the more left its distance is on the x-axis of the pdf (cf. Figure 2a). To illustrate that Figure 2b plots the probability of $y$ being the nearest neighbor of $x$ given $d_{x,y}$ (the filled area).

Transforming all original distances into the probability that a point $y$ is a nearest neighbor of $x$ offers a convenient way to combine this with the opposite view (the probability $x$ is the nearest neighbor of $y$) into a single expression.

**Definition 1** *Under the assumption of independence, we compute the probability that $y$ is the nearest neighbor of $x$ given $X$ (the Normal defined by the distances $d_{x,i=1..n}$) and $x$ is the nearest neighbor of $y$ given $Y$ (the Normal defined by the distances $d_{y,i=1..n}$). We call the resulting probability* ***Mutual Proximity (MP)****:*

$$\begin{aligned} MP(d_{x,y}) &= P(X > d_{x,y} \cap Y > d_{x,y}) \\ &= P(X > d_{x,y}) \cdot P(Y > d_{x,y}), \forall d_{x,y} > 0 \end{aligned} \tag{2}$$

Clearly the assumption of independence of $P(X)$ and $P(Y)$ will be violated, still MP has, as we will show empirically, largely positive effects especially in high dimensional data spaces with high hubness.

### 4.3 Properties

MP is symmetric $MP(d_{x,y}) = MP(d_{y,x})$ and its values are normalized to the interval $[0 - 1]$. Note that the method can therefore be easily used to linearly combine multiple different distance measures.

MP will only be high if both nearness probabilities are high and thus if their distance indicates a close mutual relationship in terms of their distance distributions. If this is not the case, i.e., one of the probabilities is small, their MP will be small too.

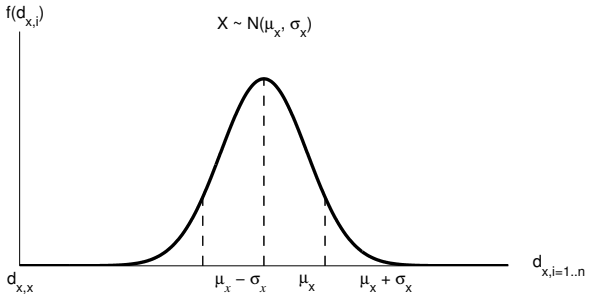
### 4.4 Matlab

The following Octave [3] /Matlab [4] code snippet demonstrates the simplicity of the method. It computes $\mathbf{D}_{MP}$ for a given $n \times n$ distance matrix $\mathbf{D}$:

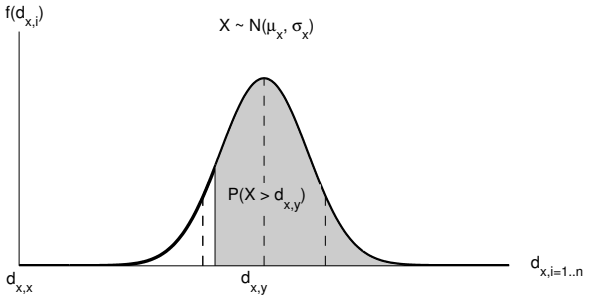
```
m = mean(D);
s = std(D);

for i = 1:n
  for j = (i+1):n
    D_MP(i, j) =
      (1 - normcdf(D(i, j), m(i), s(i))) *
      (1 - normcdf(D(i, j), m(j), s(j)));
  end
end
```

[3] http://www.gnu.org/software/octave/

[4] http://www.mathworks.com/products/matlab/




(a) The closer other elements are to $x$, the more left their distance is located on x-axis of the density function plot. The leftmost point is the distance $d_{x,x} = 0$.




(b) The shaded area shows the probability that $y$ is the nearest neighbor of $x$ based on the distance $d_{x,y}$ and $X$. The closer $y$ is to $x$ (the smaller $d_{x,y}$) the higher the probability.

**Figure 2**: Schematic plot of the probability density function of a normal distribution $X \backsim \mathcal{N}(\hat{\mu}_x, \hat{\sigma}_x)$ which was estimated from the distances $d_x$.

## 5. EVALUATION

To evaluate the effects of using MP for the selected audio similarity algorithm we use eight different music collections (see Table 1 for collection characteristics like collection size or numbers of genres). The collection sizes range from 100 to 16 000 music pieces. Four collections (*homburg* [10], *ismir2004-train*[5] and *ismir2004-dev*, *ballroom* [7]) are public benchmark sets. The other collections (DB-S, DB-XL, DB-RBA, DB-L) are private benchmark collections. Each individual song in the collections is assigned to a music genre.

### 5.1 Metrics

The following metrics are used to evaluate the Mutual Proximity transformation with the music similarities:

#### *5.1.1 Leave-One-Out, $k$-Nearest Neighbor Genre Classification ($C^k$)*

We compute the $k$-nearest neighbor classification accuracy using a leave-one-out genre classification. The $k$-NN classification accuracy is denoted with $C_k$. Higher values indicate more consistent retrieval quality in terms of the class/genre. It is one of the standard methods to measure the retrieval quality of audio similarity algorithms.

#### *5.1.2 Goodman-Kruskal Index ($I_{GK}$)*

To evaluate the impact of the MP transformation, we also compute the Goodman-Kruskal Index [8]. $I_{GK}$ is a ratio computed from the number of *concordant* ($Q_c$) and *disconcordant* ($Q_d$) distance tuples. A distance tuple is concordant if $d_{i,j} < d_{k,l}$ and objects $i$, $j$ are from the same classes and $k$, $l$ from different classes. It is is discordant if $d_{i,j} > d_{k,l}$.

[5] http://ismir2004.ismir.net/genre_contest/index.htm

$I_{GK}$ is bound to the interval $[-1; 1]$. The higher it is, the more concordant distance tuples were found, thus indicating tighter and better clustering.

#### *5.1.3 Hubness ($S^k$)*

We also compute the *hubness* [19] for each collection. Hubness is defined as the average skewness of the distribution of $k$-occurrences ($N_k$):

$$S^k = \frac{\mathrm{E}\left[(N_k - \mu_{N_k})^3\right]}{\sigma^3_{N_k}}$$

Positive skewness indicates high hubness (high number of hub objects), skewness values around zero a more even distribution of nearest neighbors.

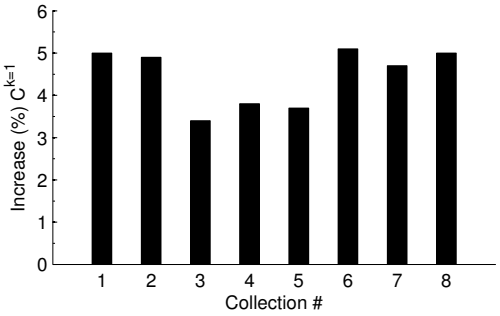
### 5.2 Results

Table 1 displays the full evaluation results of the selected audio similarity algorithm according to the metrics introduced in the previous section. In the table each collection spans two rows, the first row showing the evaluation metrics computed for the original data space and the second row listing the values when using MP.

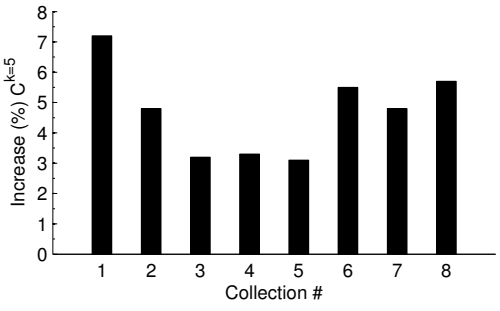
The collections listed in the table are sorted by their hubness value in the original distance space. From the high hubness values ($1.93 - 9.29$) the hub problem of the audio similarity algorithm can be clearly seen. For example, a single hub song in DB-L is occurring in over 10% of all $k = 5$ nearest neighbor lists in the collection. On the contrary hubness is sharply decreasing when looking at the values MP produces, which may indicate that MP creates a more evenly spread object space. The average hubness values per collection decrease from 4.6 to 1.2; Figure 4 shows the individual hubness values in a plot. Another metric which increases for all collections is the Goodman-Kruskal index($I_{GK}$), indicating a better separation of genres in the distance space after using MP.

| *Name, # Collection* | *Genres* | $n$ | *Distance* | $C^{k=1}$ | +/- | $C^{k=5}$ | +/- | $S^{k=5}$ | $I_{GK}$ |
|---|---|---|---|---|---|---|---|---|---|
| DB-S | 16 | 100 | js | 57.0% | **5.0** | 42.0% | **7.2** | **1.93** | 0.59 |
| *1* | | | $MP$ | 62.0% | | 49.2% | | 0.65 | 0.74 |
| ballroom | 8 | 698 | js | 54.7% | **4.9** | 46.3% | **4.8** | **2.63** | 0.16 |
| *2* | | | $MP$ | 59.6% | | 51.1% | | 1.05 | 0.20 |
| ismir 2004 (tr) | 6 | 729 | js | 82.9% | **3.4** | 73.6% | **3.2** | **3.61** | 0.35 |
| *3* | | | $MP$ | 86.3% | | 76.8% | | 1.15 | 0.41 |
| ismir 2004 (tr+dev) | 6 | 1458 | js | 86.5% | **3.8** | 80.6% | **3.3** | **4.22** | 0.37 |
| *4* | | | $MP$ | 90.3% | | 83.9% | | 1.31 | 0.42 |
| homburg | 9 | 1886 | js | 46.7% | **3.7** | 43.6% | **3.1** | **4.26** | 0.30 |
| *5* | | | $MP$ | 50.4% | | 46.7% | | 1.33 | 0.34 |
| DB-XL | 21 | 16778 | js | 55.9% | **5.1** | 46.6% | **5.5** | **4.69** | 0.12 |
| *6* | | | $MP$ | 61.0% | | 52.1% | | 1.37 | 0.19 |
| DB-RBA | 36 | 3423 | js | 51.4% | **4.7** | 41.6% | **4.8** | **5.77** | 0.26 |
| *7* | | | $MP$ | 56.1% | | 46.4% | | 1.69 | 0.31 |
| DB-L | 22 | 2526 | js | 77.2% | **5.0** | 68.1% | **5.7** | **9.29** | 0.47 |
| *8* | | | $MP$ | 82.2% | | 73.8% | | 1.16 | 0.55 |

**Table 1**: The detailed evaluation results comparing the use of MP with a standard variant. The evaluation criteria are described in Section 5.




(a) Increase of 1-NN classification rates in %-points




(b) Increase of 5-NN classification rates in %-points

**Figure 3**: Using MP increases the genre 1/5-NN classification rates of each music collection significantly.

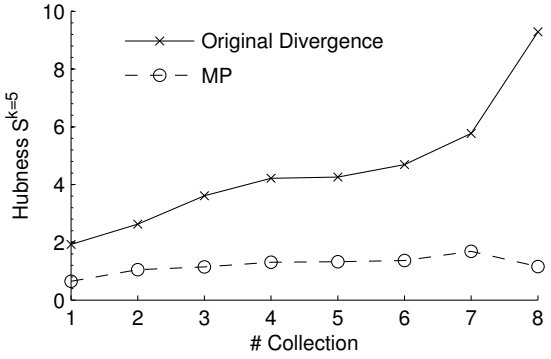



**Figure 4**: Hubness values decrease when using the MP; a desirable property for a music recommendation algorithm.

We also compute the $C^{k=1}$ and $C^{k=5}$ genre classification rates. When comparing the two values computed for the original audio similarity measure and MP, we see that in all collections the retrieval quality in terms of genre classification rates increases noticeable when MP is used. For $k = 1$ classification increases on average by 4.5%-points, for $k = 5$ on average by 4.7%-points. Figure 3 and Table 1 (columns +/-) show the increase in 1/5-NN genre classification rates per collection.

### 5.3 Summary

To summarize the evaluation we can see that all metrics we computed to evaluate the impact of MP lead to significant improvements in the retrieval quality of the basic audio similarity measure proposed by Mandel and Ellis [15] in 2005.

In the case of the *homburg* and *ismir 2004* music genre collections its performance is now very close to the reported performance of the audio similarity algorithm by Pohle et al. [18] which ranked top in the 2009/10 MIREX (task: *audio similarity and retrieval*) evaluations. Their quite sophisticated algorithm uses MFCCs, Spectral Contrast features, "Harmonicness", "Attackness" and a Rhythm component (Table 2).

| Collection | Pohle [18] | Mandel [15] | Mandel+MP |
|---|---|---|---|
| homburg | 50.9% | 46.7% | 50.4% |
| ismir 2004 (tr) | 87.6% | 82.9% | 86.3% |
| ismir 2004 (tr+dev) | 90.4% | 86.5% | 90.3% |

**Table 2**: Nearest-neighbor ($k = 1$) leave-one-out- genre classification accuracy comparison using MP. The numbers from Pohle are taken from the referenced paper [18].

## 6. DISCUSSION AND FUTURE WORK

The authors find it very exciting to see the potential for improvements that one of the most basic content-based audio similarity algorithms still offers without any modification of its MFCC similarity features. Without using any class information and only by using a simple unsupervised transformation rewarding common neighbors, the long standing problem of hub songs is alleviated and genre classification rates for the algorithm can be increased significantly.

As Mutual Proximity can be used with arbitrary distance measures it is also interesting to study the effects of MP on datasets from different research areas. Preliminary tests in that direction show that MP has in fact similar beneficial effects on any high dimensional dataset suffering from high hubness in its original distance space.

## ACKNOWLEDGMENTS

This research is supported by the Austrian Research Fund (FWF) (grants P22856-N23, L511-N15, Z159) and the Vienna Science and Technology Fund WWTF (Audiominer, project number MA09-024). The Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry for Transport, Innovation, and Technology.

## 7. REFERENCES

[1] J.J. Aucouturier and F. Pachet. Improving timbre similarity: How high is the sky. *Journal of Negative Results in Speech and Audio Sciences*, 1(1):1–13, 2004.

[2] J.J. Aucouturier and F. Pachet. A scale-free distribution of false positives for a large class of audio similarity measures. *Pattern Recognition*, 41(1):272–284, 2008.

[3] K.P. Bennett, U. Fayyad, and D. Geiger. Density-based indexing for approximate nearest-neighbor queries. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 233–243. ACM, 1999.

[4] A. Berenzweig. *Anchors and hubs in audio-based music similarity*. PhD thesis, Columbia University, 2007.

[5] L. Ertöz, M. Steinbach, and V. Kumar. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In *SIAM international conference on data mining*, volume 47, 2003.

[6] A. Flexer, D. Schnitzer, M. Gasser, and T. Pohle. Combining Features Reduces Hubness in Audio Similarity. *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR'10), Utrecht, The Netherlands*, 11, 2010.

[7] F. Gouyon, S. Dixon, E. Pampalk, and G. Widmer. Evaluating rhythmic descriptors for musical genre classification. In *Proceedings of the AES 25th International Conference*, pages 196–204, 2004.

[8] S. Gunter and H. Bunke. Validation indices for graph clustering. *Pattern Recognition Letters*, 24(8):1107–1113, 2003.

[9] A. Hicklin, B. Ulery, and C.I. Watson. *The myth of goats: How many people have fingerprints that are hard to match?* US Dept. of Commerce, National Institute of Standards and Technology, 2005.

[10] H. Homburg, I. Mierswa, B. Möller, K. Morik, and M. Wurst. A benchmark dataset for audio classification and clustering. In *Proceedings of the International Conference on Music Information Retrieval*, pages 528–31, 2005.

[11] R.A. Jarvis and E.A. Patrick. Clustering using a similarity measure based on shared near neighbors. *IEEE Transactions on Computers*, pages 1025–1034, 1973.

[12] W. Jin, A. Tung, J. Han, and W. Wang. Ranking outliers using symmetric neighborhood relationship. *Advances in Knowledge Discovery and Data Mining*, pages 577–593, 2006.

[13] M. Lagrange, J. Serrà. Unsupervised Accuracy Improvement for Cover Song Detection Using Spectral Connectivity Networks. *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR'10), Utrecht, The Netherlands*, 11, 2010.

[14] B. Logan. Mel frequency cepstral coefficients for music modeling. In *International Symposium on Music Information Retrieval (ISMIR'00)*, 2000.

[15] M. Mandel and D. Ellis. Song-level features and support vector machines for music classification. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR'05), London, UK*, 2005.

[16] E. Pampalk, A. Flexer, and G. Widmer. Improvements of audio-based music similarity and genre classification. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR'05), London, UK*, 2005.

[17] T. Pohle, P. Knees, M. Schedl, and G. Widmer. Automatically adapting the structure of audio similarity spaces. *Proceedings of the 1st Workshop on Learning the Semantics of Audio Signals (LSAS 2006)*, page 66, 2006.

[18] T. Pohle, D. Schnitzer, M. Schedl, P. Knees, and G. Widmer. On rhythm and general music similarity. In *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR'09)*, 2009.

[19] M. Radovanović, A. Nanopoulos, and M. Ivanović. Hubs in space: Popular nearest neighbors in high-dimensional data. *The Journal of Machine Learning Research*, 11:2487–2531, 2010.