Efficient Iterative Mean Shift based Cosine Dissimilarity for Multi-Recording Speaker Clustering

Mohammed Senoussaoui[1,2], *Patrick Kenny*[2], *Pierre Dumouchel*[1] and *Themos Stafylakis*[1,2]

[1]École de Technologie Supérieure (ÉTS), Canada
[2]Centre de Recherche Informatique de Montréal (CRIM), Canada

ABSTRACT

Speaker clustering is an important task in many applications such as Speaker Diarization as well as Speech Recognition. Speaker clustering can be done within a single multi-speaker recording (Diarization) or for a set of different recordings. In this work we are interested by the former case and we propose a simple iterative Mean Shift (MS) algorithm to deal with this problem. Traditionally, MS algorithm is based on *Euclidean* distance. We propose to use the *Cosine* distance in order to build a new version of MS algorithm. We report results as measured by *speaker* and *cluster* impurities on NIST SRE 2008 datasets.

***Index Terms*—** Speaker Clustering, Mean Shift (MS), *Cosine* distance, *Speaker* Impurity, *Cluster* Impurity.

1. INTRODUCTION

The objective of clustering is to create clusters that are as much as possible dense and distant from others by linking nearby observations in terms of a given metric. For speech processing and many other disciplines, clustering task is essential especially when dealing with unlabeled data. In speech processing, we often want to label data according to units such as linguistic (phonemes, words, phrases…) or to a speaker state (emotion, sex, health, age…) or even to the speaker himself. When label corresponds to the speaker we call this *Speaker Clustering*. Speaker clustering is an important task for many fields such as automatic speaker adaptation in speech recognition systems and speaker diarization. It could also serve to improve storage allocation and help searching into a huge multimedia dataset, etc.

In this work, our main contribution is twofold. In one hand, it is to adopt the Mean Shift algorithm [1] to perform clustering of a large dataset (SRE 2008 data). In other hand, it lies in the use of *Cosine* distance in order to build a new version of MS algorithm. Traditionally, MS algorithm uses *Euclidean* distance.

Now that we fixed our goal and described our clustering approach, one question remains. What speech signal representation will be used in our system? It will be the *i-vector* which became the state-of-the-art feature vector in speaker recognition field [2].

The rest of this paper is organized as follows. In the next section, we present some preliminaries about the baseline Mean Shift algorithm as well as our proposed version of this algorithm. In section 3, we expose the trade-off plotting and impurity metrics used to evaluate the performance of our clustering system. Before concluding, we outline our experiments and discuss results in section 4.

2. MEAN SHIFT

Mean Shift is a nonparametric iterative mode-seeking algorithm introduced by Fukunaga [1]. Despite its first appearance in 1975, Mean Shift remained in oblivion except for works as in [3] which generalize the original version proposed by Fukunaga. MS algorithm reappeared in 2002 with the work of Comaniciu [4] in image processing. Recently, Stafylakis et al. [5][6] published nice works in which they generalized the basic *Euclidean* space MS to the manifolds of parameters space. They applied this version to the clustering problem tested on broadcast news diarization task. The most important propriety of MS algorithm is that it performs clustering without any prior knowledge about neither clusters number nor distribution shape of these clusters.

2.1. Basic Mean Shift

Mean Shift is a member of Kernel Density Estimation (KDE) family also known as Parzen windows. Given a set of *d*-dimensional observations (*i-vectors* in our case) $S=\{x_1,x_2,\ldots,x_n\}$ the kernel density function in a given point *x* is given by the following formula:

$$\hat{f}(x)=\frac{1}{nh^d}\sum_{i=1}^{n}k\left(\frac{x-x_i}{h}\right) \tag{1}$$

where *k*(*x*) is a kernel function and *h* is its radial width or the so-called kernel bandwidth. The bandwidth *h* is the only tuned parameter in the Mean shift algorithm. Its role is to smooth the estimated density function. Both kernel and bandwidth should satisfy some conditions in order to ensure some proprieties like asymptotic, unbiasedness and consistency. These conditions are discussed in details in [1].

978-1-4799-0356-6/13/$31.00 ©2013 IEEE

Considering a differentiable kernel function, we can estimate the density gradient as the gradient of the kernel density estimate given in (1) as follows:

$$\hat{\nabla} f(x) \equiv \nabla \hat{f}(x) = \frac{1}{nh^d}\sum_{i=1}^{n} \nabla k\left(\frac{x-x_i}{h}\right). \tag{2}$$

One simple type of kernels (the shadow of the flat kernel) is the Epanechnikov kernel, given by the following formula:

$$K(x) = \begin{cases} \frac{1}{2} c_d^{-1}\left(d+2\right)\left(1-x^T x\right) & \text{if } x^T x < 1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where c_d represents the volume of the d-dimensional unit sphere.

Substituting (3) in (2) we obtain the density kernel estimate for the Epanechnikov kernel:

$$\begin{aligned} \hat{\nabla} f(x) &= \frac{1}{n(h^d c_d)} \frac{d+2}{h^2} \sum_{x_i \in S_h(x)} \left(x_i - x\right) \\ &= \frac{n_x}{n(h^d c_d)} \frac{d+2}{h^2} \left(\frac{1}{n_x} \sum_{x_i \in S_h(x)} \left(x_i - x\right) \right) \end{aligned} \tag{4}$$

where $S_h(x)$ is the set of n_x points that the pairwise *Euclidean* distance to x is less or equal to a threshold (i.e. the bandwidth h):

$$S_h\left(x\right) \equiv \left\{ x_i : \left\| x_i - x \right\| \le h \right\} \tag{5}$$

The second part of equation (4) is what we called the Sample Mean Shift $M\left(x\right)$:

$$\begin{aligned} M(x) &= \left(\frac{1}{n_x} \sum_{x_i \in S_h(x)} \left(x_i - x\right) \right) \\ &= \left(\frac{1}{n_x} \sum_{x_i \in S_h(x)} x_i \right) - x \\ &= \mu - x \end{aligned} \tag{6}$$

Finally, we can observe that Mean shift in a given point x is nothing than the shift of this point to the sample mean $\mu = \frac{1}{n_x} \sum_{x_i \in S_h(x)} x_i$ of its neighboring points falling within a hypersphere of radius h (i.e. bandwidth). The *Euclidean* distance governs the allocation of neighboring points to that hypersphere. Thus, an iterative executing of sample mean calculation followed by window shifting (as depicted in the **Algorithm 1**) leads to get a stationary point (density mode). Proofs of convergence and mathematical details of this procedure can be found in [1]. Additionally, a generalization of the basic *Euclidean* space MS to the manifolds of parameters space could be found in [6].

Algorithm 1 Mean Shift Intuition

```
    – $i \leftarrow 0$
    – Center a window around $x_i$ //Initialization
repeat
        – $\mu$ //estimate the sample mean of data falling
          within the window
        – $x_{i+1} \leftarrow \mu$
        – Move the window from $x_i$ to $x_{i+1}$
        – $i \leftarrow i+1$
until Stabilization //No more move of the sample mean
```

2.2. Cosine distance Mean Shift

Recently, classification based on *Cosine* distance became the state-of-the-art methods used in speaker recognition fields [2][7]. Moreover, the success of *Cosine* distance for distinguishing between speakers motivates scientists to study in more details the *Cosine* distance. In speaker recognition field, researches lead to discover that length (*Euclidean* Norm) normalization of *i-vectors* is advantageous for generative model based on Gaussian assumption [8]. Note that the length normalization is an intrinsic operation in the *Cosine* distance (see equation (7)). Furthermore, it is also proved that whitening of high dimensional data followed by projecting it onto the unit sphere (i.e. length normalization of data) Gaussianize these data. For an entertaining discussion of this curious fact, we will refer you to a web site[1].

In *Euclidean* space geometry, the *Cosine* distance measures dissimilarity between two points relatively to the space origin. In fact, this dissimilarity is nothing more than a dot product of the Cartesian coordinates vectors normalized by their *Euclidean* norms (lengths). Let $\{x,y\}$ be a set of two d-dimensional vectors of Cartesian coordinates of two points in the space. The Angular distance D between these points is given by the following:

$$D(x,y) = \frac{x \cdot y}{\left\|x\right\|\left\|y\right\|} \tag{7}$$

The original Mean Shift version based on a flat kernel relies on *Euclidean* distance to find points falling within the window as shown in (5). The main contribution of this work is to propose the use of the angular distance instead of the *Euclidean* one. In order to get this new version, one modification is introduced in equation (5) which is:

$$S_h\left(x\right) \equiv \left\{ x_i : D\left(x_i, x\right) \ge h \right\} \tag{8}$$

where $D\left(x_i, x\right)$ is the *Cosine* distance between x_i and x.

[1] http://ontopo.wordpress.com/2009/03/10/reasoning-in- higher-dimensions-measure/

2.3. Clustering with Mean Shift

The natural way to perform clustering using Mean Shift algorithm is to run separately the iterative Mean Shift for each point in the dataset. Indeed, some observations will converge to the same stationary point (density mode). The number of unique stationary points (after pruning) corresponds the number of detected clusters and obviously data converge to a same mode that belongs to the same cluster. We usually call these points, *basin of attraction* of its mode.

3. PERFORMENCE MESURING

We use *speaker* to refer to a reference cluster i.e. the actual cluster or the speaker and *cluster* to refer to the detected cluster. Recently in [9], authors propose two *impurity* measures, one for speaker and the other for *cluster*. Mathematical details of these impurities are described in [9].

3.1. *Cluster* Impurity

Cluster impurity I_c measures the heterogeneousness of detected clusters i.e. data from different speakers. *Cluster* impurity ranges between 0 to almost 1. Smaller the value of I_c more discrimination exists between clusters. $I_c = 0$ means zero uncertainty about assigning each utterances to clusters.

3.2. *Speaker* Impurity

Speaker impurity I_s measures the dispersion of observations of an actual cluster (*speaker*) amid different *clusters* (detected clusters). More smaller the value of I_s more certain about assigning utterances of a same *speaker* to the same *cluster*. $I_s = 0$ corresponds to the trivial solution of assigning all observations in a single cluster.

3.3. *DET plot illustration*

A certain analogy exists between errors committed by a Speaker Verification system (*missed detections* vs. *false alarms*) and the previous cited impurities [9]. This is why using a trade-off plotting as DET plot can help in interpretation and comparison of results between clustering systems.

4. EXPERIMENT SETUP

4.1. Feature extraction

4.1.1. Short-time signal parameterization

Each 10ms, 60 Mel Frequency Cepstral Coefficients (MFCC) were extracted within a 25ms hamming window (19 MFC Coefficients + Energy + first & second Deltas) from speech signal. These features were normalized with a short time Gaussianization.

4.1.2. Universal Background Model (UBM)

We use a gender-independent GMM UBM containing 2048 Gaussians. This UBM is trained with the LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004–2005 SRE (only telephone speech).

4.1.3. I-vectors extraction

We use a gender independent *i-vector* extractor of dimension 800. Its parameters are estimated on the following data: LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; Fisher data and NIST 2004 and 2005 SRE (telephone + Microphone speech).

4.2. Channel compensation

4.2.1. Linear Discriminant Analysis (LDA)

LDA projection matrix is A estimated on the 800 dimensions *i-vectors* representing the same speech data used in *i-vector* extractor training except of Fisher part. Only telephone data is used to estimate the between classes scatter matrix however we used both telephone and microphone data to estimate the within class scatter matrix. The LDA is applied to reduce *i-vectors* dimensionality to 200.

4.2.2. Within Class Covariance Normalization (WCCN)

Same data used to estimate A is used to estimate the within class covariance matrix W. However, the *i-vectors* were first subject to a mapping to 200 dimensions using LDA. All *i-vectors* will be rotated (normalized) using Cholesky decomposition matrix of the inverse of W.

4.3. Test dataset

In this work we adopted telephone SRE 2008 data as our test dataset. This dataset contains 3090 telephone recordings of 1270 gender independent speakers.

4.4. Experiments and results

In order to evaluate and compare performance of our proposed Mean Shift algorithm based *Cosine* distance, we ran it several times on the test dataset by changing the threshold h from *0.1* to *0.99*. In an analog way, we also ran the original version of Mean Shift algorithm i.e. based *Euclidean* distance on the same dataset also by changing the threshold h from *10* to *35*.

In this paper, we present the most interesting results in which we can observe the important operating points like equal impurities and when the estimated number of clusters N_c is approximately equal to the actual number of speakers (see Tab. 1).

Observing reported results in Table 1 we can draw some conclusions. Firstly, one can state that both systems (i.e. *Euclidean* and *Cosine* based systems) reached an equal impurities point (0.123 for *Euclidean* vs. **0.09** for *Cosine*) after over estimating the number of clusters (1496 for *Euclidean* vs. 1414 for *Cosine*). However, the over estimation in the *Cosine* distance system is less than the one in the *Euclidean* system. Fortunately, over estimation is usually better than under estimation in clustering systems.

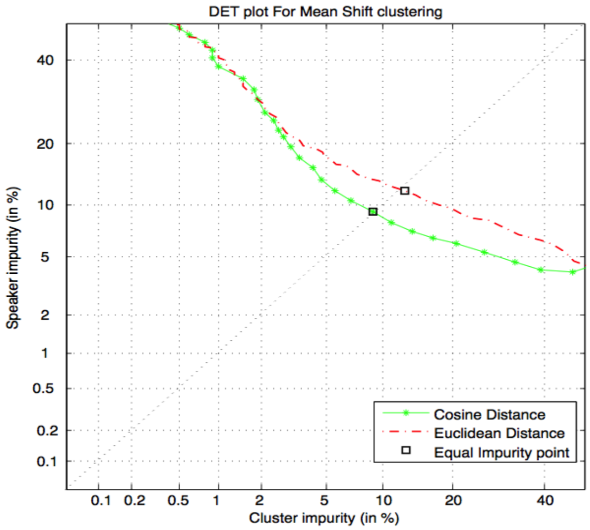
In terms of equal impurity, it is clear that the system based on *Cosine* distance outperforms the *Euclidean* one (0.123 for *Euclidean* vs. **0.09** for *Cosine*). If we compare equal impurity point to the NIST Equal Error Rate (EER) in the verification system (in the same way as in [9]), we observe a relative improvement of **11%** (from 12.3% to **9%**) between *Cosine* distance system and *Euclidean* system.

Tab. 1 *Pertinent results (Cosine vs. Euclidean) as measured by cluster impurity I_c, speaker impurity I_s and the corresponding detected number of clusters N_c in function of the threshold h. Not that the actual number of clusters (speakers) is 1270. The gray highlighted rows represent when the system estimates approximately the actual number of clusters. Finally the bold entries rows represent when I_c is approximately equal to I_s.*

***Euclidean* distance**				***Cosine* distance**			
h	I_c	I_s	N_c	*h*	I_c	I_s	N_c
23.8	0.238	0.085	1297	0.54	0.207	0.060	1161
23.7	0.215	0.090	1337	0.55	0.168	0.065	1225
23.6	0.199	0.096	1368	0.56	0.137	0.071	1286
23.5	0.178	0.101	1397	0.57	0.109	0.080	1352
23.4	0.153	0.108	1438	**0.58**	**0.089**	**0.092**	**1414**
23.3	0.142	0.113	1461	0.59	0.069	0.106	1471
23.2	**0.126**	**0.120**	**1496**	0.60	0.056	0.120	1537
23.1	0.105	0.128	1533	0.61	0.047	0.135	1602

Unfortunately we are not able to perform a back-to-back comparison with results reported in [9], since authors report results on SRE 2006 dataset. Note that it is well known to the speaker recognition community that the speaker verification task on this data was easier than other datasets like SRE 2008. However their equal impurity was 13.9% on SRE 2006 data compared to our **9%** on SRE 2008.

Fig. 1 *DET plots for Mean Shift clustering using Cosine distance vs. Euclidian distance.*



Finally, the trade-off graphs (depicted in Fig. 1) of the both systems reveal that the system based on *Cosine* distance outperform the *Euclidean* system over almost the whole of the graph especially in the low speaker impurity rate region.

5. CONCLUSION

In this paper we have presented a modified version of Mean Shift algorithm obtained by replacing the *Euclidean* distance with the *Cosine* one. As it is tested on a large dataset (SRE 2008) we achieved a relative improvement of 11% - as measured by equal impurities - compared to the baseline version of MS. Also, we have shown that new MS version outperforms the baseline for most operating points in the trade-off plotting. Finally, we expect that these results could be improved if we apply a score normalization method (as ZT-norm or s-norm) to the *Cosine* scores [7]. Furthermore, we expect also an improvement by implementing a gender independent scoring in the way presented in [10].

6. REFERENCES

[1] K. Fukunaga and L. Hostetler, “The estimation of the gradient of a density function, with applications in pattern recognition,” IEEE Trans. on Information Theory, vol. 21, no. 1, pp. 32–40, January 1975.

[2] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” IEEE Transactions on Audio, Speech, and Language Processing, July 2010.

[3] Y. Cheng, “Mean Shift, Mode Seeking, and Clustering,” IEEE Trans. PAMI, vol. 17, no. 8, pp. 790-799, 1995.

[4] D. Comaniciu and P. Meer, “Mean shift: A robust approach toward feature space analysis,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pp. 603 – 619, May 2002.

[5] T. Stafylakis, V. Katsouros, and G. Carayannis, “Speaker clustering via the mean shift algorithm,” in Odyssey 2010: The Speaker and Language Recognition Workshop - Odyssey-10, Brno, Czech Republic, June 2010.

[6] T. Stafylakis, V. Katsouros, P. Kenny, and P. Dumouchel, “Mean Shift Algorithm for Exponential Families with Applications to Speaker Clustering,” Proc. Odyssey Speaker and Language Recognition Workshop, Singapore, June 2012.

[7] N. Dehak, R. Dehak, J. Glass, D. Reynolds, and P. Kenny, "Cosine Similarity Scoring without Score Normalization Techniques," Proc. IEEE Odyssey Workshop, Brno, Czech Republic, June 2010.

[8] D. Garcia-Romero, “Analysis of i-vector length normalization in Gaussian-PLDA speaker recognition systems,” in Proceedings of Interspeech, Florence, Italy, Aug. 2011.

[9] D. van Leeuwen, “Speaker linking in large data sets,” Proc. IEEE Odyssey Workshop, Brno, Czech Republic, June 2010.

[10] M. Senoussaoui, P. Kenny, N. Brummer, E. de Villiers and P. Dumouchel, “Mixture of PLDA Models in I-Vector Space for Gender-Independent Speaker Recognition,” in Proceedings of Interspeech, Florence, Italy, Aug. 2011.