MODEL BASED MULTIPLE AUDIO SEQUENCE ALIGNMENT

Doğaç Başaran [*], *A. Taylan Cemgil* [†], *Emin Anarım* [*]

Boğaziçi University
Electrical and Electronics Engineering Department [*]
Computer Engineering Department [†]
İstanbul, Turkey
{dogac.basaran,taylan.cemgil,anarim}@boun.edu.tr

ABSTRACT

We formulate alignment of multiple and partially overlapping audio sequences in a probabilistic framework. We define and compare four generative models for time varying features extracted from audio clips that are recorded independently and asynchronously. We are able to handle missing data and multiple clips where no clip is covering the entire material. We define proper scoring functions for each model and the matching is achieved with a sequential alignment algorithm. The simulation results on real data suggest that the approach is able to handle difficult ambiguous scenarios or partial matchings.

Index Terms— Audio alignment, Audio matching, Maximum likelihood, Probabilistic Model

1. INTRODUCTION

Audio alignment is often regarded as an identification problem where an unknown audio segment is matched to a large audio database. There exist robust audio fingerprinting methodologies that achieve high matching performances under very noisy conditions [1],[2],[3]. In this paper, we focus on multiple alignment problem, where we view audio matching from a different perspective. Imagine that there are several microphones that record an audio scene but the microphones are not synchronized. Each microphone starts and stops recording at different times independent of each other. Hence, two recorded audio clips may or may not overlap. The aim is to align these audio clips according to their starting points on an unknown time line, somewhat like solving a puzzle. One major difference from the common audio alignment setup is that there is no clean original source database but only some possibly noisy observations of the source and none of the audio clips have to cover the entire timeline.

Our motivation in dealing with multiple audio matching problem is that we wish to use the precisely aligned recordings in source separation, restoration or remastering frameworks where the sources are highly fragmented. Such a scenario might occur for example in a concert hall during a performance. Assume that some of the audience record their favorite parts of the concert with recording devices of varying quality. These audio clips each of which are recorded from a different perspective would also have different amplitude levels and noise. A possible application might be collecting these unsynchronized audio recordings on a website and try to produce a full recording of the performance by precisely aligning these sources on the generic time line. A similar approach exists in genetics that is called shotgun sequencing where the long DNA strands are assembled from shorter sequences [4]. Another visual analogy for our approach is image stitching [5] where multiple images taken from slightly different perspective are assembled into a full panoramic view.

In principle, the problem can be approached using deterministic methods such as correlation and template matching. However there are also certain limitations. First of all, the computational cost in these methods is quite high in audio applications. Most of the audio matching applications work pairwise even when they are matching multiple clips. Assuming there are K number of clips, one needs to apply pairwise matching on the order of $O(K^2)$ which can be prohibitive. In addition to that, if the audio clips do not overlap or some of the data is missing in one of the audio clips, it is not always clear how to apply simple correlation or template matching ideas. An obvious way to reduce computational complexity and the number of data is working on a feature space instead of working directly on audio data. Energy of the signal over short time windows [1], local chroma energy distributions [2] and positive spectral difference [3],[6] are the features that are widely used in the audio matching framework. Even when working with features, the problem can be still challenging when there are multiple shorter recordings and not a 'ground truth timeline'.

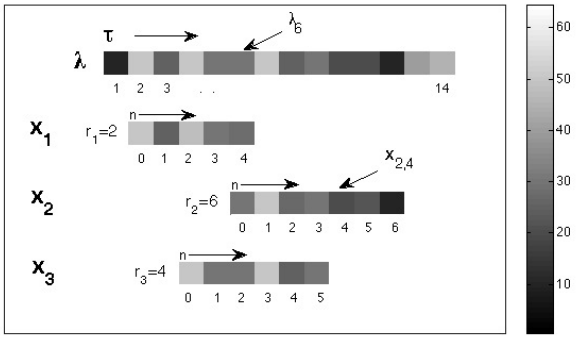
We propose a model based approach and define four generative models for different audio features. The modelling approach is flexible in a way that any feature vector is appropriate (e.g., non-negative, real, binary, discrete levels). Proper scoring functions are derived from each model. When there are only two sources, evaluating the scoring function for all possible alignments, including partial and non-matchings, is feasible. The framework extends directly to multiple sequences but exact scoring becomes intractable. Here, we propose a sequential alignment algorithm for matching multiple clips on a common time-line.

2. PROPOSED MODEL

In this section, we introduce our probabilistic model for the multiple alignment problem with a toy example given in Figure 1. The features are the time varying energy coefficients in one sub-band. The main idea of the model is that properly aligned feature sequences are noisy realizations or functions of a common but unobserved feature sequence, if a full length recording of the audio scene would be available. We denote this hidden feature vector with $\lambda_{1:T}$. Here $\tau = 1 \dots T$ is a global time frame index. When considering a sin-

DB is supported by DPT - TAM Project number 2007K120610
ATC is supported by TÜBİTAK project number 110E292

Figure 1: Model Illustration via a toy example. λ is hidden, x_1, x_2 and x_3 are observed



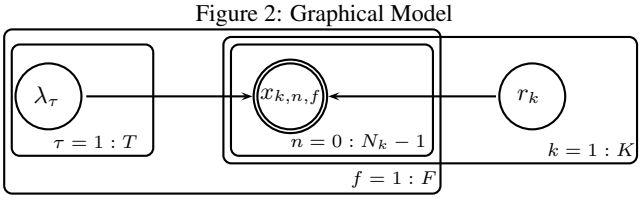
gle sub-band, λ_τ is a scalar. There are three clips observed in the Figure 1 and x_k denotes the feature vector of the k'th clip. The length of the feature vector of the k'th clip is denoted as N_k. In this example, $T = 14$, $N_1 = 5$, $N_2 = 7$, and $N_3 = 6$. Here n is a local time frame index and the spectrum coefficient of the k'th clip at local time n is denoted by $x_{k,n}$. Again, if we would consider several sub-bands, $x_{k,n}$ would be a vector. The alignment variable for the k'th clip is denoted as r_k. The second recording is aligned at global time $\tau = 6$ therefore $r_2 = 6$. In this scenario, the clips overlap with each other at several points. To be specific, $x_{1,4}, x_{2,0}$ and $x_{3,2}$ coincide at global time $\tau = 6$ and It can be observed from the figure that each of these coefficient values are close to each other since they are observations of a common source λ_6. Following this idea, a template generative model is defined as;

$$\lambda_{1:T} \sim p(\lambda_{1:T})$$
$$r_k \sim p(r_k) = \prod_{\tau=1}^{T-N_k+1} \pi_{k,\tau}^{[r_k=\tau]}$$
$$x_{k,n} \sim p(x_{k,n}|r_k, \lambda_{1:T}) = \prod_{\tau=1}^{T} p(x_{k,n}|r_k, \lambda_\tau)^{[n=\tau-r_k]}$$

where $[\cdot]$ is the indicator function which is equal to one if the expression inside is true. The alignment variable r_k is chosen to be distributed with a generic distribution where the alignment of the k'th clip is at time τ is represented with the probability $\pi_{k,\tau}$. In this paper, we assume that each r_k is uniformly distributed. The hidden coefficients λ_τ are assumed to be a-priori independent ($p(\lambda_{1:T}) = \prod_{\tau=1}^{T} p(\lambda_\tau)$). Here, the $[n = \tau - r_k]$ expression in the observation model indicates that $x_{k,n}$ is conditioned on λ_τ only if $\tau = r_k + n$ which means the n'th coefficient of the k'th source is aligned to time τ. The graphical model is shown in Figure 2. It includes an extra index f which denotes the sub-band number. However in the rest of the paper, the f index is omitted to ease the representations.

It is important to mention that the goal is not to estimate the hidden features $\lambda_{1:T}$ but to find the most likely alignment of the clips denoted as $r^*_{1:K}$. This is the prime mode of the joint conditional probability $p(r_{1:K}|x_{1:K,0:N_k-1})$. Assuming there is no prior information about the true alignment of the sources, one can use the marginal likelihood $p(x_{1:K,0:N_k-1}|r_{1:K})$ instead of the posterior

Figure 2: Graphical Model



probability:

$$p(x_{1:K,0:N_k-1}|r_{1:K}) = \int d\lambda_{1:T} \prod_{k=1}^{K} \prod_{n=0}^{N_k-1} p(x_{k,n}|r_k, \lambda_{1:T}) \prod_{\tau=1}^{T} p(\lambda_\tau)$$

Note that, λ_τ are independent from each other and $x_{k,n}$ are conditionally independent given $\lambda_{1:T}$ and $r_{1:K}$. It is important to mention that the choices of prior and likelihood distributions are conjugate pairs for all models is essential for the derivation of the computation of the exact marginal likelihood. Then by maximizing the loglikelihood

$$\mathcal{L}_K(r_{1:K}) = \log p(x_{1:K,0:N_k-1}|r_{1:K})$$

the optimum alignment is achieved as,

$$r^*_{1:K} = \arg\max_{r_{1:K}} \mathcal{L}_K(r_{1:K})$$

We can interpret this formulation also from a Bayesian model selection perspective [7]. Each configuration of $r_{1:K}$ correspond to an alternative alignment, and we are comparing different alignments after integrating out the model parameters to find the 'model' that describes the data best.

Note that this model is quite generic and can be used for a variety of feature sets. In the sequel, we will propose for generative models for positive, non-negative, real and binary features. Each model follows the template model but with different choices of prior ($p(\lambda_\tau)$) and likelihood ($p(x_{k,n}|r_k, \lambda_\tau)$) distributions which are listed in Table 1. Through out the paper, $\mathcal{IG}, \mathcal{G}, \mathcal{N}, \mathcal{B}, \mathcal{BE}$, Dir and $\mathcal{M}$ represent inverse gamma, gamma, Gaussian, beta, Bernoulli, Dirichlet and Multinomial distributions respectively. This list is by no means exhaustive; we could choose other conjugate pairs as well (such as Poisson-Gamma or Gaussian-Gaussian).

Gamma observation model: The first model is useful for positive features such as factors obtained from non-negative matrix decomposition or time varying spectral energy. In this paper, we investigate two feature sets for this model. The feature set $1a$ is directly defined as the energy in sub-bands. The feature set $1b$ is defined as positive spectral difference [6]. We choose a gamma distribution for modelling positive random variables and inverse gamma as a conjugate prior for the hidden sequence. Here, the mean and the variance

Table 1: Prior and likelihood distributions for each model

Models	$p(\lambda_\tau)$	$p(x_{k,n}\|r_k, \lambda_\tau)$
Model 1	$\mathcal{IG}(\lambda_\tau; \alpha_\lambda, \beta_\lambda)$	$\mathcal{G}(x_{k,n}; \alpha, \frac{\alpha}{\lambda_\tau})$
Model 2	$\mathcal{IG}(\lambda_\tau; \alpha_\lambda, \beta_\lambda)$	$\mathcal{N}(x_{k,n}; 0, \lambda_\tau)$
Model 3	$\mathcal{B}(\lambda_\tau; \alpha_\lambda, \beta_\lambda)$	$\mathcal{BE}(x_{k,n}; \lambda_\tau)$
Model 4	$\text{Dir}(\lambda_{1:Q,\tau}; \alpha_{1:Q})$	$\mathcal{M}(x_{1:Q,k,n}; 1, \lambda_{1:Q,\tau})$

of $x_{k,n}$ coefficient are λ_τ and λ_τ^2/α respectively. Therefore α behaves as a control parameter and adjusts how much $x_{k,n}$ deviates from λ_τ.

Gaussian variance observation model: In the second model, the observations are real, but the hidden sequence is assumed to be positive, and corresponds to the variance of the observations. We define the feature set 2 as the difference of the spectral energy between consecutive windows in sub-bands. Here, there is no control parameter, λ_τ directly determines how much $x_{k,n}$ deviates from zero.

Bernoulli observation model: In the third model, the observations are binary and are drawn from a Bernoulli distribution. The hidden sequence is defined as the parameter of the Bernoulli distribution and assumed to be beta distributed. Here, the feature set 3 is chosen as the thresholded spectral energy coefficients.

Multinomial observation model: The fourth model is an extended version of the model 3 where there are more than just two distinct levels. This model is useful when the features are categorical. Accordingly, the feature set 4 is chosen as the quantized spectral energy coefficients with Q levels. Note that the multinomial distribution has the number of trial parameter as 1. Then the $x_{1:Q,n,k}$ is a vector for which only one element of the vector is active and the rest of the elements are equal to zero. As an example, if there are $Q = 3$ levels and the second level is selected, the vector is, $x_{1:Q,n,k} = \{0, 1, 0\}$.

Since the r_k are discrete, the search domain is finite and by computing the score for each possible alignment $r_{1:K}$, it is straightforward to obtain the most possible alignment $r^*_{1:K}$. However, for large K, searching the entire space for each possible alignment is clearly intractable due to the astronomical state space size. Our preliminary experiments with batch methods such as the Gibbs sampler, even when enhanced with different annealing schemata such as gradually decreasing the α parameter has not proven very effective. The likelihood surface is very rough, therefore, we resort here to an intuitive sequential greedy algorithm.

3. SEQUENTIAL ALIGNMENT ALGORITHM

Our algorithm proceeds sequentially where sequences are selected in some random order. We fix the position of the first sequence, i.e., fix $r^*_1 = 0$, and the alignment of the second sequence r_2 is computed relative to first sequence. For each possible value of r_2, the log-likelihood $\mathcal{L}_2(r^*_1, \hat{r}_2)$ is computed and the maximum is chosen as r^*_2. We proceed in a greedy fashion where for $k = 2 \dots K$ the alignment of the k'th sequence r_k, the log-likelihood $\mathcal{L}_k(r^*_{1:k-1}, r_k)$ is computed for each possible value of r_k and the one that maximizes the likelihood is chosen as r^*_k.

We observe that the success of the sequential algorithm depends on the success of the alignments of first few sequences. Since the sequences are aligned sequentially, if the first few sequences are not matched correctly, the remaining sequences can not be aligned correctly. Our key idea to overcome this problem is to randomize the ordering of the sequences in the alignment procedure. If k'th sequence do not overlap with the previous sequences or a very small overlap occurs, the alignment r_k is treated as unreliable and the ordering of the sequences is changed such that the non-overlapping source is put to the end of a queue. Misalignments are typically prevented by re-ordering or permuting sequences when there is no overlap or a small overlap.

Initial ordering also plays a crucial role in the success of the algorithm. Some permutations may lead to more successful alignments. Clearly, it is not feasible to apply the algorithm to each of the $K!$ permutations. Therefore the algorithm is applied for P random permutations of the sequences and among the resulting alignments, the one that maximizes the log-likelihood is chosen to be the estimated alignment. We also make sure that we include 'promising' permutations such as when the sequences are sorted decreasing according to their length as longer sequences tend to be matched more reliably. The pseudocode of the sequential matching algorithm is given in Algorithm 1.

Algorithm 1 Sequential Alignment Method

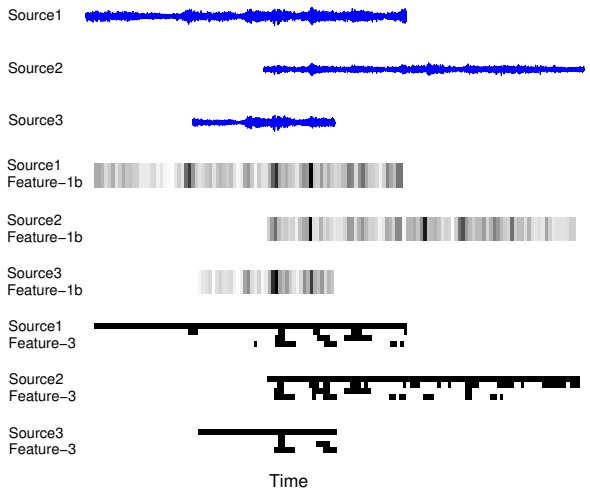
Initialize:
for $i = 1$ to *Max # of trials* **do**
  Choose a permutation of $1 \dots K$ as σ_i,
  Permute sequences as $x_k \leftarrow x_{\sigma_i(k)}$ for all k
  $R^{(i)}_1 = 0$, $k = 2$
  while $k \leq K$ **do**
    $r^*_k = \arg\max_{r_k} \mathcal{L}_k(R^{(i)}_{1:k-1}, r_k)$
    if Number of overlapping samples $> \epsilon$ **then**
      $R^{(i)}_{1:k} \leftarrow (R^{(i)}_{1:k-1}, r^*_k)$, $k = k + 1$
    else
      Move to back, $\sigma_i \leftarrow [\sigma_i(-k), \sigma_i(k)]$,
      Repermute $x_\kappa \leftarrow x_{\sigma_i(\kappa)}$ for all $\kappa \geq k$
    end if
  end while
end for
Winner $= \arg\max_i \mathcal{L}_K(R^{(i)}_{1:K})$, $r^*_{1:K} = R^{(\text{Winner})}_{1:K}$

4. SIMULATION RESULTS AND DISCUSSION

In this part, we discuss several aspects of the proposed model and whether the sequential algorithm is appropriate for the alignment problem. The questions investigated are: how much the model fits to the data, which features represent the audio data better and which features are more immune to noise and volume variations. Experiments are conducted with both synthetic data and real data. The synthetic data is generated from the models with various hyperparameter sets. Pairwise matching results suggest that the approach is successful as long as the parameters are set to their true values. The real data simulations are formed in the following way: A stereo music file of length 2 minutes that is recorded at 8KHz sampling rate is short time Fourier transformed with 25 ms non-overlapping windows. $K = 8$ sources are formed at each experiment with minimum length of 2 seconds and maximum length of 1 minute. The sources are formed equally from the right and left channels (4 sources from each channel). Each source is multiplied with a volume variable m^1_k which is in the range $0.5 < m^1_k < 1$. The starting points ($r_{1:K}$) and the lengths of the sources ($N_{1:K}$) and the volume variables are chosen randomly at each experiment. A stereo bar ambiance recording is divided into clips following the same alignments and lengths and added to the original signal as a structured noise. Noise sources are also multiplied with a volume variable m^2_k which is set randomly in 2 different ranges to simulate different SNR cases. In experiments, two different music signals are used with the same structured noise.

The spectrum is divided into sub-bands according to bark scale band edges up to $3150Hz$. For the feature sets $1a$ and 4, four sub-bands are used with band edges $[200 - 400]$, $[400 - 920]$, $[920 - 1720]$ and $[1720 - 3150]$ Hz. The squares of the coefficients are summed through frequency index in one band. The obtained matrices of size $4 \times N_k$ are used as feature set $1a$. Then the

Figure 3: Sources and features illustration



coefficients are non-uniformly quantized with μ-law into $Q = 6$ levels and $6 \times 4 \times N_k$ size matrices are used as feature set 4. For the feature set $1b$, the positive spectral difference values are squared and summed through frequency and each observation is represented with $1 \times N_k$ vectors. For the feature sets 2 and 3, 14 critical bands are used in the range $200Hz$ and $3150Hz$ and squared transform coefficients are summed through frequency in each sub-band. The first difference between sub-bands are used as feature set 2 where the observations are represented with $14 \times (N_k - 1)$ matrices. For the feature set 3, obtained $14 \times N_k$ matrices are thresholded with a threshold that preserves %95 of the total energy of the signal. Figure 3 shows three overlapping sources that are contaminated with noise and their respective feature sets $1b$ and 3 with 4 sub-bands. The estimation of the hyper-parameter set $(\alpha, \alpha_\lambda, \beta_\lambda)$ for the real scenario has a key role in the success of the alignment algorithm. In this work, we use an iterative Newton's method on the score functions to obtain optimum hyper-parameter sets for each model where the ground truth for the alignments is assumed to be known.

For evaluation of the alignment performance, we define a function $\phi(\hat{r}_i, \hat{r}_j)$ that determines whether or not sequences i and j are mutually correctly aligned. Here, r denotes the ground truth alignment variables. Assuming that $r_i^{\text{true}} < r_j^{\text{true}}$, It is defined as;

$$\phi(\hat{r}_i, \hat{r}_j) = \begin{cases} [\hat{r}_i + N_i \leq \hat{r}_j + \epsilon], & r_i^{\text{true}} + N_i < r_j^{\text{true}} \\ [\hat{r}_i - \hat{r}_j = r_i^{\text{true}} - r_j^{\text{true}}], & r_i^{\text{true}} + N_i \geq r_j^{\text{true}} \end{cases}$$

ϕ acts like an indicator function that results in "1" if the alignment is correct and "0" if it is false. Sometimes non-overlapping sources are aligned back-to-back such that only very few samples overlap. In this case, if the number of overlapping samples are smaller than ϵ, which is chosen as 5, the alignment is considered to be true. The alignment performance criteria $\Omega(\hat{r}_{1:K})$, the total alignment score, is then defined as the number of true pairwise alignments over total number of pairs:

$$\Omega(\hat{r}_{1:K}) = \frac{2}{K(K-1)} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} \phi(\hat{r}_i, \hat{r}_j)$$

Highest score to be achieved is "1" where all the sources are aligned perfectly and lowest score is "0" where no sources are aligned correctly. The experiments are conducted for high SNR and low SNR cases. The mean and standard deviation of the performance scores for each feature set in each SNR case are listed in the Table 2.

Table 2: Experimental Results- Real data simulations

	Feature Sets				
SNR	$1a$	$1b$	2	3	4
High	0.93±0.1	0.97±0.06	0.89±0.1	0.74±0.15	0.65±0.18
Low	0.82±0.19	0.88±0.17	0.81±0.14	0. 67±0.15	0.52±0.12

The results suggest that the audio data in the alignment problem can be approached with the models given in Section 2. The sequential algorithm is able to find the true alignments in most of the cases. The feature set $1b$ (positive spectral difference) has the most successful scores in both SNR cases which supports the immunity against noise and volume variations. Thresholded data and spectral difference works only when there are enough number of sub-bands which are chosen as all critical bands in the range. The quantized feature set gives the best results when there are $Q = 6$ levels and 4 sub-bands. Among the robustness of the alignment, the processing time is yet another important criteria. One of the advantages of the model is that instead of pairwise matching of the observations, the model aligns each observation sequence with a hidden audio content which reduces the computational burden. Since the processing time increases with the amount of data to be processed, the feature sets with higher number of sub-bands require more time. Therefore the feature set $1b$ is also the best feature set in the processing time performance.

5. CONCLUSION

In this work, we proposed a model based approach for the multiple audio sequence alignment problem and defined 4 generative models for different feature sets. We derived proper score functions for each model. The results show that our approach is both fast and robust against noisy situations and volume variations. We obtain successful results with the sequential greedy algorithm however we believe that utilizing more advanced inference methods such as sequential Monte Carlo algorithms would increase the performance both in robustness and processing time.

6. REFERENCES

[1] Wang, A.L, "An Industrial-Strength Audio Search Algorithm", InProc. ISMIR, Baltimore, USA, 2003.

[2] M. Muller, F. Kurth and M. Clausen, "Audio Matching via Chroma-based statistical features", In Proc. Int. Conf. on Music Info. Retr. ISMIR-05, pages 288-295, London, 2005.

[3] S. Dixon and G. Widmer, "Match: A music alignment tool chest", in Proc. ISMIR, London, GB, 2005

[4] Weber J. L., Myers E. W., "Human Whole-Genome Shotgun Sequencing", Genome Res. 1997 7: 401-409

[5] Brown, M. and Lowe, D., "Automatic Panoramic Image Stitching using Invariant Features", IJCV: Vol. 74, pp.59-73, 2007

[6] J.P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. Sandler, "A tutorial on onset detection in musical signals", IEEE Transactions on Speech and Audio Processing, vol. 13, no. 5, pp. 1035-1047, 2005.

[7] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006