Multi-channel audio statistical restoration

Yinhong Liu
Cambridge University Engineering Department
Cambridge, UK
williamliuyinhong@gmail.com

Simon Godsill
Signal Processing Group
Cambridge University Engineering Department
Cambridge, UK
sjg@eng.cam.ac.uk

***Abstract*—Phonograph record is an analog sound storage medium that has played an important part in sound history. Modulated spiral groves on the disc are usually inevitably damaged by scratches and dust, which lead to noises with different statistical characteristics. Among those different type of noises, there is one common click-shape noise that have some degree of correlation between channels. Its origins have been explained as a deviation of the probe needle means it shifts closer to one wall but further from another with similar distances. Our aim is to restore this type of noises on dual mono audio signals with a statistical method. There are previous researches on restoring single-channel noisy data based on generative models. However multi-channel noisy data contain more audio information than single-channel data, due to its channel correlation. To exploit this property, we propose multivariate Gaussian models for both signal and noise models. Then we derive the Maximum A Posteriori (MAP) estimation of the underlying data. Additionally, we also use the Maximum Likelihood method to optimize model parameters. In the end we compare the restoration performances between our model and the baseline model on both synthetic and real-life noisy music data.**

***Index Terms*—Multi-channel audio siganl, dual mono audio, signal restoration, multivariate Gaussian, Maximum A Posterior, Maximum Likelihood**

I. Introduction

The music signals are considered as time-series sequences, with sampling frequency at 44.1kHz. We treated the noisy signals as a sum of noise-free music signals and additive noises in the following way, which means we could model them separately.

$$\mathbf{y}_t = \mathbf{x}_t + i_t \mathbf{n}_t. \tag{1}$$

In Equation 1, $\mathbf{y}_t$ is a single data point of the noisy signal at time t and for dual mono signals it has form $\mathbf{y}_t = [y_{lt} \; y_{rt}]^T$, where subscript l and r stand for left and right channels . $\mathbf{x}_t$ is for the underlying noise-free signal and $\mathbf{n}_t$ is for multi-channel noises. They have same shapes with $\mathbf{y}_t$. For $\mathbf{x}_t$, the dual mono has suggested that both channels have the same values. i_t is a flag which is 1 when this data point is considered as corrupted and 0 when it is believed as noise-free. When i_t is 0, $\mathbf{y}_t = \mathbf{x}_t$, which means we believed we have clean and uncorrupted data for this time t. Therefore, for a piece of raw signals, data that are considered as noisy are all labelled out. In [1], it has proposed a MAP missing data interpolation method dealing with single-channel noisy data, which means it treated all $\mathbf{y}_t$, $\mathbf{x}_t$ and $\mathbf{n}_t$ as 1 dimensional variables. Study in [2] proposes another interpolation method for multi-channel noisy data based on auto-regression model, but it treats the corrupted data as completely missing, so not using noisy information at all. We take the [1] as a baseline model. The baseline model uses uni-variate Gaussian models for both x_t and n_t. It cannot exploit information from the extra channel. We extend the baseline models, so that $\mathbf{x}_t$ and $\mathbf{n}_t$ are treated as random Gaussian vectors.

II. Method

We first use multivariate Gaussian random vectors to model the music signals and the multi-channel noises separately. Then we derive a MAP estimation of the underlying data $\mathbf{x}_u$ and a ML estimation of the noises parameters. In Section II-D, we use an iterative algorithm to repeatedly update estimated noise parameters and underlying audio data. It converges after a few iterations and best performance is achieved. show how to apply this method.

A. Signal Modelling

We use short music signal pieces with length $N = 5000$ data points, which is equivalent to duration of approximately 0.1s, therefore we can assume this short piece is wide sense stationary. For one piece of the underlying signal, we treat it as a joint Gaussian distribution, which is a common model for audio data. It could be expressed in the following vector-form equations:

$$\begin{aligned} P(\mathbf{x}) &= \mathcal{N}\big(\mathbf{x} \mid \mathbf{0}, \mathbf{R_x}\big) \\ &= \frac{1}{(2\pi)^{N/2}|\mathbf{R_x}|^{1/2}} \exp\Big(-\frac{1}{2}\mathbf{x}^{\mathbf{T}}\mathbf{R_x}^{-1}\mathbf{x}\Big), \end{aligned}$$

where $\mathbf{x} \in \mathbb{R}^N$ is the signal vector. As aforementioned, only a fraction of data points in $\mathbf{x}$ are considered as corrupted. Therefore, we partition $\mathbf{x}$ into corrupted data points $\mathbf{x}_u \in \mathbb{R}^M$ which is unknown, and all noise-free data points $\mathbf{x}_k \in \mathbb{R}^{N-M}$ which is known. The posterior probability distribution of $\mathbf{x}_u$ given $\mathbf{x}_k$ can be expressed as follows:

$$\begin{aligned} P(\mathbf{x}_u|\mathbf{x}_k) &= \frac{P(\mathbf{x})}{P(\mathbf{x}_k)} \\ &= \frac{P(\mathbf{U}\mathbf{x}_u + \mathbf{K}\mathbf{x}_k)}{P(\mathbf{x}_k)}, \end{aligned} \tag{2}$$

where $\mathbf{U} \in \mathbb{R}^{N\times M}$ and $\mathbf{K} \in \mathbb{R}^{N\times(N-M)}$ are two partition matrix, which map $\mathbf{x}_u$ and $\mathbf{x}_k$ into $\mathbf{x}$. Therefore, the MAP estimation of $\mathbf{x}_u$ can be derived by maximizing Equation 2 w.r.t. $\mathbf{x}_u$:

978-1-7281-7202-6/20/$31.00 ©2020 Crown

$$\mathbf{x}_u^{\text{MAP}} = -\mathbf{M}_u^{-1}\mathbf{M}_k\mathbf{x}_k,$$

where matrix $\mathbf{M_u} \in \mathbb{R}^{M\times M}$ and $\mathbf{M_k} \in \mathbb{R}^{M\times(N-M)}$ and they can be interpreted as the selections of elements in $\mathbf{R_x}$ that corresponding to $\mathbf{x}_u$ and $\mathbf{x}_k$:

$$\mathbf{M}_u = \mathbf{U^T R_x^{-1} U},$$
$$\mathbf{M}_k = \mathbf{U^T R_x^{-1} K}.$$

B. Multi-channel Noises modelling

As aforementioned, for the noise at a single time point $\mathbf{n}_t$, we consider it is from two channels, thus the dimension is 2. We model it as a zero mean multivariate Gaussian random vector. Therefore it has probability distribution as follow:

$$P(\mathbf{n}_t) = P\big(\mathbf{y}_t \mid \mathbf{x}_t; \boldsymbol{\Sigma}_t\big) = \mathcal{N}\big(\mathbf{0}, \boldsymbol{\Sigma}_t\big), \tag{3}$$

where the co-variance matrix $\boldsymbol{\Sigma}_t \in \mathbb{R}^{2\times 2}$, that has the following structure with 3 parameters σ_{lt}, σ_{rt} and ρ_t controlling its characteristics.

$$\boldsymbol{\Sigma}_t = \begin{bmatrix} \sigma_{lt}^2 & \rho_t\sigma_{lt}\sigma_{rt} \\ \rho_t\sigma_{lt}\sigma_{rt} & \sigma_{rt}^2 \end{bmatrix}. \tag{4}$$

We would like to consider noises for all corrupted data points in a music piece together, whose number is M. Therefore, we stack $\mathbf{n}_t$, for all t that the noise flag $i_t = 1$, into a vector $\mathbf{n}_{lr} \in \mathbb{R}^{2\times M}$. In a similar method, we also stack $\mathbf{y}_t$ and $\mathbf{x}_t$, for all t, into vectors $\mathbf{y}_{lr}$ and $\mathbf{x}_{lr}$, both $\in \mathbb{R}^{2\times N}$. The three stacked vectors $\mathbf{n}_{lr}$, $\mathbf{y}_{lr}$ and $\mathbf{x}_{lr}$ are still multivariate Gaussian random vectors.

$$\mathbf{n}_{lr} = [...n_{lt_1}\ n_{rt_1}\ n_{lt_2}\ n_{rt_2}\ n_{lt_3}\ n_{rt_3}\ ...]^T,$$
$$\mathbf{y}_{lr} = [y_{l1}\ y_{r1}\ y_{l2}\ y_{r2}\ y_{l3}\ y_{r3}\ ...]^T,$$
$$\mathbf{x}_{lr} = [x_{u1}\ x_{u1}\ x_{u2}\ x_{u2}\ x_{u3}\ x_{u3}\ ...]^T$$

where t_1, t_2, t_3... satisfy $i_t = 1$ and 1, 2, 3, ... are all time points in a music piece. The three vectors have the relation that $\mathbf{n}_{lr} = \mathbf{y}_{lr} - \mathbf{x}_{lr}$. Then we define a mapping matrix $\mathbf{N} \in \mathbb{R}^{2M\times M}$, so that $\mathbf{x}_{lr} = \mathbf{N}\mathbf{x}_u$:

$$\mathbf{N} = \begin{bmatrix} 1 & & \\ 1 & & \\ & 1 & \\ & 1 & \\ & & ... \end{bmatrix}.$$

Similar to Equation 3, the p.d.f of the noise vector $\mathbf{n}_{lr}$ is also a multivariate Gaussian distribution, which can be expressed as follow:

$$P(\mathbf{n}_{lr}; \mathbf{C_m}) = P(\mathbf{y}_{lr}|\mathbf{x}_{lr}; \mathbf{C_m}) = \mathcal{N}(\mathbf{y}_{lr}|\mathbf{x}_{lr}; \mathbf{C_m}).$$

Theoretically, the co-variance matrix $\mathbf{C_m} \in \mathbb{R}^{2M\times 2M}$ can have any arbitrary symmetric structure. However, for the purpose of later simplifying estimation of the co-variance matrix, we make some assumptions on its structure, which will be discussed in details in Section 4. Now the posteriori probability (Equation 2) of $\mathbf{x}_u$ can be re-written, which now also conditions on double channel noisy data $\mathbf{y}_{lr}$.

$$P\big(\mathbf{x}_u \mid \mathbf{x}_k, \mathbf{y}_{lr}; \mathbf{C_m}\big) = \frac{P\big(\mathbf{x}, \mathbf{y}_{lr}; \mathbf{C_m}\big)}{P\big(\mathbf{x}_k, \mathbf{y}_{lr}; \mathbf{C_m}\big)} = \frac{P\big(\mathbf{x}\big)P\big(\mathbf{n}_{lr}; \mathbf{C_m}\big)}{P\big(\mathbf{x}_k, \mathbf{y}_{lr}; \mathbf{C_m}\big)}.$$

Given the relation between $\mathbf{y}_{lr}$ and $\mathbf{n}_{lr}$, it is easy to show that $P(\mathbf{x}, \mathbf{y}_{lr}) = P(\mathbf{x}, \mathbf{n}_{lr})$. The underlying music data $\mathbf{x}$ and the noise $\mathbf{n}_{lr}$ are independent. The denominator evidence is a constant. Therefore, the MAP estimation of the corrupted data $\mathbf{x}_u$ can be derived by maximizing the nominator w.r.t. $\mathbf{x}_u$:

$$\mathbf{x}_u^{\text{MAP}} = \big(\mathbf{M}_u + \mathbf{N}^T\mathbf{C_m}^{-1}\mathbf{N}\big)^{-1}\big(\mathbf{N}^T\mathbf{C_m}^{-1}\mathbf{y}_{lr} - \mathbf{M}_k\mathbf{x}_k\big). \tag{5}$$

C. Noises parameters estimation

The MAP interpolation Equation 5 requires the co-variance matrix $\mathbf{C_m}$ to compute the corrupted data estimation. $\mathbf{C_m}$ is a matrix describing characteristics of the noises and it is usually unknown. Therefore, we propose a Maximum Likelihood estimation method to estimate its value using the raw data $\mathbf{x}$. As aforementioned, $\mathbf{C_m}$ can have arbitrary symmetric structure. However, we apply a few assumptions to simplify the estimation process. We assume that the random vector $\mathbf{n}_t$ is i.i.d. This implies two things: firstly, the noises are temporarily uncorrelated, which means there is no correlation between $\mathbf{n}_{t_1}$ and $\mathbf{n}_{t_2}$ if $t_1 \neq t_2$. Additionally, $\mathbf{n}_{t_1}$ and $\mathbf{n}_{t_2}$ have the same co-variance matrix $\boldsymbol{\Sigma}_t$ for any t. Therefore, the resulting structure of $\mathbf{C_m}$ can be expressed as:

$$\mathbf{C_m} = \begin{bmatrix} \boldsymbol{\Sigma} & & & \\ & \boldsymbol{\Sigma} & & \\ & & \boldsymbol{\Sigma} & \\ & & & ... \end{bmatrix}. \tag{6}$$

As shown in Equation 4, $\boldsymbol{\Sigma}$ contains three parameters σ_l, σ_r and ρ, so does $\mathbf{C_m}$. Therefore, we maximize the likelihood of $\mathbf{C_m}$ with respect to these parameters respectively. The log-likelihood distribution can be expressed as the marginalization of joint distribution of raw data $\mathbf{y}_{lr}$ and $\mathbf{x}_k$ and unknown underlying clean data $\mathbf{x}_u$.

$$\begin{aligned}\mathcal{L} &= log\big(P(\mathbf{y}_{lr}|\mathbf{C_m}; \mathbf{R_x}),\big) \\ &= \log\Big(\int P(\mathbf{y}_{lr}, \mathbf{x}_k, \mathbf{x}_u|\mathbf{C_m}; \mathbf{R_x})\ d\mathbf{x}_u\Big) \\ &= \log\Big(\int P(\mathbf{y}_{lr}|\mathbf{x}_u, \mathbf{C_m})P(\mathbf{x}_u, \mathbf{x}_k; \mathbf{R_x})\ d\mathbf{x}_u\Big) \\ &= \log\Big(\int \mathcal{N}\big(\mathbf{y}_{lr}|\mathbf{x}_{lr}, \mathbf{C_m}\big)\mathcal{N}\big(\mathbf{x}|\mathbf{0}, \mathbf{R_x}\big)\ d\mathbf{x_u}\Big).\end{aligned}$$

We expand the product of two Gaussian and complete the square to construct a new Gaussian p.d.f of $\mathbf{x}_u$: $\mathcal{N}(\mathbf{x}_u|\boldsymbol{\mu}, \mathbf{C})$, which can then be marginalized to 1. By dropping constant terms, the log-likelihood ends up with Equation 7:

$$\mathcal{L} = \log\left\{\frac{|\mathbf{C}|^{\frac{1}{2}}}{(2\pi)^{\frac{m+n}{2}}|\mathbf{C_m}|^{\frac{1}{2}}|\mathbf{R_x}|^{\frac{1}{2}}}\exp\Big[-\frac{1}{2}\big(\mathbf{y}_{lr}^T\mathbf{C_m}^{-1}\mathbf{y}_{lr} + \mathbf{x}_k^T\mathbf{K}^T\mathbf{R_x}^{-1}\mathbf{K}\mathbf{x}_k - \boldsymbol{\mu}^T\mathbf{C}^{-1}\boldsymbol{\mu}\big)\Big]\int\mathcal{N}(\mathbf{x}_u|\boldsymbol{\mu},\mathbf{C})d\mathbf{x}_u\right\}$$
$$\approx \frac{1}{2}\log(|\mathbf{C}|) - \frac{1}{2}\log(|\mathbf{C_m}|) - \frac{1}{2}\log(|\mathbf{R_x}|) - \frac{1}{2}(\mathbf{y}_{lr}^T\mathbf{C_m}^{-1}\mathbf{y}_{lr} + \mathbf{x}_k^T\mathbf{K}^T\mathbf{R_x}^{-1}\mathbf{K}\mathbf{x}_k - \boldsymbol{\mu}^T\mathbf{C}^{-1}\boldsymbol{\mu}). \quad (7)$$

The $\boldsymbol{\mu} \in \mathbb{R}^M$ and $\mathbf{C} \in \mathbb{R}^{M\times M}$ have following expressions:

$$\mathbf{C} = (\mathbf{N}^T\mathbf{C_m}^{-1}\mathbf{N} + \mathbf{U}^T\mathbf{R_x}^{-1}\mathbf{U})^{-1},$$
$$\boldsymbol{\mu} = \mathbf{C}(\mathbf{N}^T\mathbf{C_m}^{-1}\mathbf{y}_{lr} - \mathbf{U}^T\mathbf{R_x}^{-1}\mathbf{K}\mathbf{x}_k).$$

The ML estimation of $\mathbf{C_m}$ is obtained by maximizing Equation 7 with respect to σ_l, σ_r and ρ. Although it is hard to find an analytical solution, there are numbers of numerous optimization methods available, ranging from gradient based methods to searching methods [3]. Here we consider a gradient descent method. Scalar a is used to represent the optimization variables, which are σ_l, σ_r and ρ.

$$a^{n+1} = a^n - \lambda_a\frac{\partial\mathcal{L}}{\partial a^n}, n \geq 0.$$

We need to use following identities, as shown in [4], to help with the derivation of the gradients:

$$\frac{\partial\ln|\mathbf{M}|}{\partial x} = \mathrm{Tr}\big(\mathbf{M}^{-1}\frac{\partial\mathbf{M}}{\partial x}\big),$$
$$\frac{\partial\mathbf{M}^{-1}}{\partial x} = -\mathbf{M}^{-1}\frac{\partial\mathbf{M}}{\partial x}\mathbf{M}^{-1}.$$

Then the gradient can be expressed as follow:

$$\frac{\partial\mathcal{L}}{\partial a} = \frac{1}{2}\mathrm{Tr}(\mathbf{C}^{-1}\frac{\partial\mathbf{C}}{\partial a}) - \frac{1}{2}\mathrm{Tr}(\mathbf{C_m}^{-1}\frac{\partial\mathbf{C_m}}{\partial a}) + \frac{1}{2}\mathbf{y}_{lr}^T\mathbf{C_m}^{-1}\frac{\partial\mathbf{C_m}}{\partial a}\mathbf{C_m}^{-1}\mathbf{y}_{lr} - \frac{1}{2}\boldsymbol{\mu}^T\mathbf{N}^T\mathbf{C_m}^{-1}\frac{\partial\mathbf{C_m}}{\partial a}\mathbf{C_m}^{-1}N\boldsymbol{\mu} + \boldsymbol{\mu}^T\mathbf{N}^T\mathbf{C_m}^{-1}\frac{\partial\mathbf{C_m}}{\partial a}\mathbf{C_m}^{-1}\left[\mathbf{NC}\Big(\mathbf{N}^T\mathbf{C_m}^{-1}\mathbf{y}_{lr} - \mathbf{U}^T\mathbf{R_x}^{-1}\mathbf{K}\mathbf{X}_k\Big) - \mathbf{y}_{lr}\right], \quad (8)$$

where

$$\frac{\partial\mathbf{C}}{\partial a} = \mathbf{C}\mathbf{N}^T\mathbf{C_m}^{-1}\frac{\partial\mathbf{C_m}}{\partial a}\mathbf{NC}.$$

$\frac{\partial\mathbf{C_m}}{\partial a}$ has different formulas for different parameter a. They are shown below:

$$\frac{\partial\mathbf{C_m}}{\partial\sigma_l} = \begin{bmatrix}2\sigma_1 & \rho\sigma_r\\ \rho\sigma_r & 0\end{bmatrix},$$
$$\frac{\partial\mathbf{C_m}}{\partial\sigma_r} = \begin{bmatrix}0 & \rho\sigma_l\\ \rho\sigma_l & 2\sigma_l\end{bmatrix},$$
$$\frac{\partial\mathbf{C_m}}{\partial\rho} = \begin{bmatrix}0 & \sigma_l\sigma_r\\ \sigma_l\sigma_r & 0\end{bmatrix}.$$

To further improve the parameters estimation results, we can iteratively run the estimation process and use the newly estimated parameters to interpolate the corrupted signals. Then use the restored signals as the input of the next round parameter estimation. By this way, we can progressively achieve better estimation of the model parameters and therefore more accurate restored signals.

D. Complete iterative restoration algorithm

The detailed formulas for MAP estimation of $\mathbf{x}_u$ and ML estimation of $\mathbf{C_m}$ have been shown in Section II-B and II-C. The algorithm for the complete restoration process is shown in Algorithm 1. The first iteration uses the estimated $\mathbf{x}_u$ to recalculate $\mathbf{R_x}$ and $\mathbf{C_m}$ iteratively. This is because in the first place, we used noisy signal $\mathbf{x}$ to compute the two characteristic matrix, thus their values were inevitably noisy. By doing iterative restoration, each time we can have less noisy $\mathbf{R_x}$ and $\mathbf{C_m}$ and therefore better estimation of $\mathbf{x}$. The second iteration is a simple gradient descent method to find the optimized values for noise parameters.

Algorithm 1 Complete iterative restoration algorithm

```
1: x_u^0 ← Initialized to random values
2: for i ∈ {1, 2, ..., iter1} do                    ▷ First iteration.
3:     R_x = cov[x^i x^i^T]
4:     for j ∈ {1, 2, ..., iter2} do                ▷ Second iteration.
5:         for a ∈ {σ_l, σ_r, ρ} do
6:             a^j ← a^j − λ_a ∂L/∂a^j   ▷ ∂L/∂a^j is given by Equation 8
7:         end for
8:     end for
9:     σ_l^i, σ_r^i, ρ^i ← Optimized by Gradient Descent.
10:    C_m^i ← Computed by Equation 4, 6 , using σ_l^i, σ_r^i, ρ^i.
11:    x_u^i ← Computed by Equation 5, using C_m^i.
12:    x^i ← U x_u^i + K x_k
13: end for
```

III. Experiment and Performance

We compare restoration performance of our interpolator with the baseline method over 100 pieces of both synthetic and real noisy data with duration of 0.1s. For the synthetic data, we generated the noise according to Equation 3 and superimposed it to clean music pieces. For the real world noise, we recorded lead-in part of damaged vinyl discs and superimposed them to clean music pieces. To evaluate the performances quantitatively, we compared the average R.M.S errors of the restored signals by our method and the baseline uni-variate Gaussian model.

TABLE I
Average R.M.S errors over 100 music pieces.

	Synthetic data	Real data
Proposed method	0.06	0.36
Baseline model	0.10	0.43
Noisy data	3.86	4.52

As shown in Tab. I, the unprocessed noisy data have r.m.s. errors 3.86 and 4.52 for synthetic and real data. The baseline uni-variate Gaussian model can reach 0.10 and 0.43, while our proposed method using double-channel data achieved 0.06 and 0.36 for synthetic and real data respectively. The discrepancy between the performance on synthetic and real data is because the real-world signals are always composed of various type of noises with different characteristics. From the noisy data examples in Fig. 1 and Fig. 2, we can visually show that our method have better performance as the interpolated waveform is closer to the underlying signals.

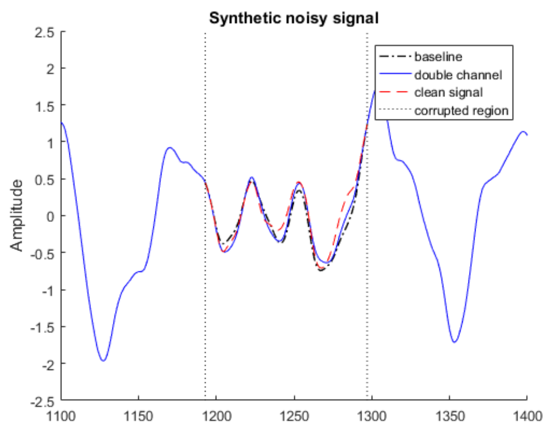

Fig. 1. Waveform comparison for synthetic noisy data.

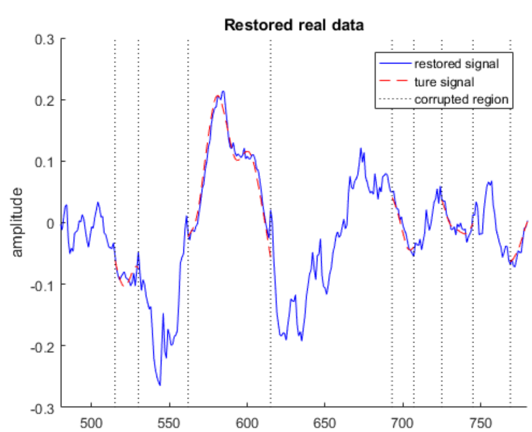

Fig. 2. Waveform comparison for real noisy data.

We also investigate the restoration performance with different noise parameters, which are noise variance σ and channel correlation coefficient ρ. Additional to our method and the baseline model, we also compare the results with a naive MAP estimation that completely ignore the noisy part of the data, which is labelled as 'No noise' in Fig.3 and Fig.4. We used synthetic noisy data with controlled noise parameters. Each data point in Fig. 3 and Fig.4 is the average r.m.s errors over 20 music pieces with duration 0.1s.

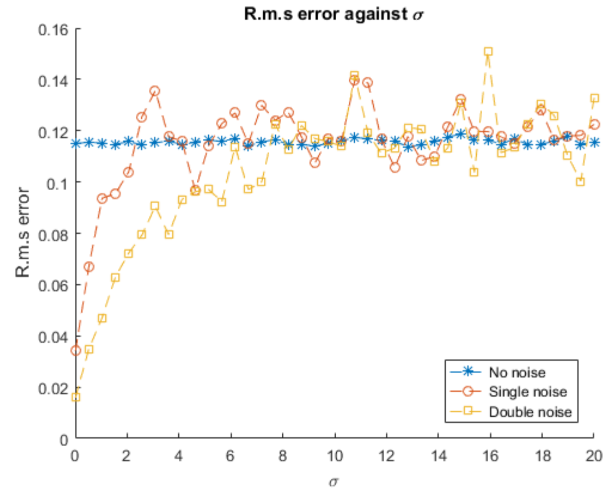

Fig. 3. Comparison of r.m.s errors against σ with a controlled ρ.

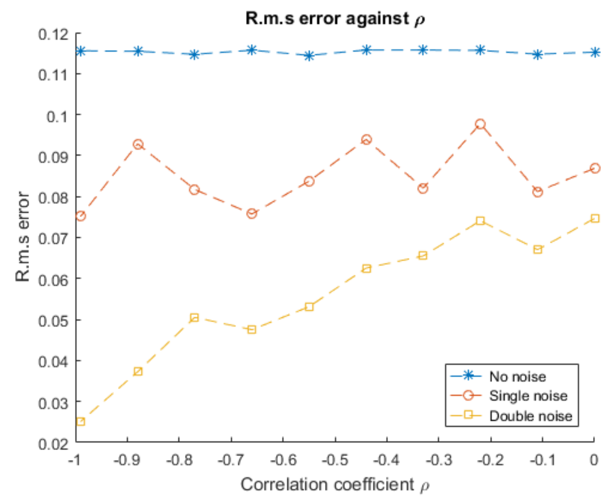

Fig. 4. Comparison of r.m.s errors against ρ with a controlled σ.

In Fig.3, the r.m.s errors are shown against the noise variance σ (Here we use the same value for both channels, $\sigma_l = \sigma_r$), while the value of ρ is controlled as -0.3. As shown in graph, the restoration performance gets worse for both methods as the σ increases. When $\sigma \leq 8$, our proposed method can produce constantly better result than the baseline model. However, when $\sigma > 8$, as the noises have too high amplitudes than the underlying signals, all methods cannot achieve good restoration performances.

In Fig.4, the r.m.s errors are shown against the noise correlation ρ, while σ is kept as 3. As shown, our model can achieve better performance for noises with strong correlation, which means we successfully take advantage of the multi-channel information. When ρ is close to -1, we can restore with almost zero estimation errors, because there is nearly complete information to exploit from two channel noises. When ρ is close 0 side, we still have better performance than the baseline methods, which can be supported by the theory that two independent noisy channels could still provide more information.

IV. Conclusion

There has been previous research on restoring single-channel noisy audio data using Gaussian model. However, dual mono audio is also common in many sound systems and contains extra information that can be exploited by a multi-channel model. In this paper, we used multi-variate Gaussians to model the double-channel noises and the underlying music signals separately. Based on this, we derived the MAP estimation of the corrupted underlying music signals. Then we also derived the ML estimation of the parameters of the noise model. The r.m.s restoration errors of our model and the baseline model have been compared and it showed that double-channel restoration model has generally better performances, especially when there are strong correlation between noises from two channels.

References

[1] S. Godsill and P. Rayner, "Digital audio restoration," *Department of Engineering, University of Cambridge*, pp. 101–126, 1997.

[2] H. Lin and S. Godsill, "The multi-channel ar model for real-time audio restoration," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005.*, Oct 2005, pp. 335–338.

[3] J. J. Moré and D. J. Thuente, "Line search algorithms with guaranteed sufficient decrease," *ACM Transactions on Mathematical Software (TOMS)*, vol. 20, no. 3, pp. 286–307, 1994.

[4] R. J. Barnes, "Matrix differentiation," *Department of Civil Engineering, University of MinnesotaMinneapolis, Minnesota, USA*, 2006.