The 2018 Signal Separation Evaluation Campaign

Antoine Liutkus[1], Fabian-Robert Stöter[1], and Nobutaka Ito[2]

[1] Inria and LIRMM, University of Montpellier, France
[2] NTT Communication Science Laboratories, NTT Corporation, Japan

Abstract. This paper reports the organization and results for the 2018 community-based Signal Separation Evaluation Campaign (SiSEC 2018). This year's edition was focused on audio and pursued the effort towards scaling up and making it easier to prototype audio separation software in an era of machine-learning based systems. For this purpose, open-source software was developed and released to automatically load, process and report performance on the new MUSDB'18 music separation database. Additionally, a new official Python version for the `BSS Eval` toolbox was released, along with reference implementations for three oracle separation methods: ideal binary mask, ideal ratio mask, and multichannel Wiener filter.

1 Introduction

Source separation is a signal processing problem that consists in recovering individual superimposed *sources* from a *mixture*. Since 2008, the role of the Signal Separation Evaluation Campaign (SiSEC) has been to compare performance of separation systems on a volontary and community-based basis, by defining tasks, datasets and metrics to evaluate methods [18,13,14,1,10,11,9]. Although source separation may find applications in several domains, the focus of SiSEC has always mostly been on audio source separation, with tasks pertaining to both speech and music separation.

This year, we decided to drop the legacy speech separation and denoising tasks UND and BGN, because they are now the core focus of very large and successful other campaigns such as CHiME [3,15,2]. Instead, most of our efforts were spent on music separation, where the SiSEC MUS task is playing an important role, both in terms of datasets and participation. We also maintained the ASY task of asynchronous separation, due to its originality and adequation with the objectives of SiSEC.

While the primary objective of SiSEC is to regularly report on the progress made by the community through standardized evaluations, its secondary objective is also to provide useful resources for research in source separation, even outside the scope of the campaign itself. This explains why the SiSEC data has always been made public, so that it could be used for related publications.

Since 2015, the scope of the SiSEC MUS data was significantly widened, so that it could serve not only for evaluation, but also for the design of music separation system. This important shift in the purpose of the SiSEC data is motivated by the recent development of systems based on deep learning, which now define the state-of-the-art and require important amounts of learning data. This lead to the proposal of the MSD [11] and the DSD100 [9] datasets, that were used in the previous two SiSEC.

This year's SiSEC present several contributions. First, the computation of oracle performance goes further than the usual Ideal Binary Mask (IBM) method to also include Ideal Ratio Mask (IRM) and Multichannel Wiener Filters (MWF). Second, we continued our effort in gathering a dataset for training music separation systems and released the MUSDB'18, that comprises almost 10 h of music with separated stems. Third, we released a new version 4 for the `BSS Eval` toolbox, that handles time-invariant distortion filters, significantly speeding up computations. Pointers to open-source implementations in Python for all these contributions may be found in the SiSEC website[1].

2 Oracle performance for audio separation

We write I as he number of channels of the audio mixture: $I = 2$ for stereo. We write x for the 3-dimensional complex array obtained by stacking the Short-Time Frequency Transforms (STFT) of all channels. Its dimensions are $F \times T \times I$, where F, T stand for the number of frequency bands and time frames, respectively. Its values at Time-Frequency (TF) bin (f, t) are written $x\left(f, t\right) \in \mathbb{C}^I$, with entries $x_i\left(f, t\right)$. The mixture is taken as the sum of the sources *images*: $x\left(f, t\right) = \sum_j y_j\left(f, t\right)$, which correspond to the isolated instruments and are also multichannel.

A filtering method $\mathbf{m}$ usually computes estimates $\hat{y}_j^{\mathbf{m}}$ for the source images linearly from x:

$$\hat{y}_j^{\mathbf{m}}\left(f, t \mid \theta_{\mathbf{m}}\right) = M_j^{\mathbf{m}}\left(f, t \mid \theta_{\mathbf{m}}\right) x\left(f, t\right), \tag{1}$$

where $\theta_{\mathbf{m}}$ are some parameters specific to $\mathbf{m}$ and $M_j\left(f, t \mid \theta_{\mathbf{m}}\right)$ is a $I \times I$ complex matrix called a TF *mask*, computed using $\theta_{\mathbf{m}}$ in a way specific to method $\mathbf{m}$. Once given the filtering strategy $\mathbf{m}$, the objective of a source separation system is to analyze the mixture to obtain parameters $\theta_{\mathbf{m}}$ that yield good separation performance.

For evaluation purposes, it is useful to know how good a filtering strategy can get, i.e. to have some upper bound on its performance, which is what an *oracle* is [17]:

$$\theta_{\mathbf{m}}^{\star} = \underset{\theta_{\mathbf{m}}}{\operatorname{argmin}} \sum_{f,t,j} \left\| y_j\left(f, t\right) - \hat{y}_j^{\mathbf{m}}\left(f, t \mid \theta_{\mathbf{m}}\right) \right\|, \tag{2}$$

where $\| \cdot \|$ is any norm deemed appropriate. In this SiSEC, we covered the three most commonly used filtering strategies, and assessed performance of their respective oracles:

[1] `sisec.inria.fr`.

1. The **Ideal Binary Mask** (*IBM*, [19]) is arguably the simplest filtering method. It processes all (f, t, i) of the mixture independently and simply assigns each of them to one source only: $M_{ij}^{\mathbf{IBM}}(f,t) \in \{0,1\}$. Its oracle performance is obtained if the source with strongest magnitude is picked each time.
2. The **Ideal Ratio Mask** (IRM), also called the α-Wiener filter [7], relaxes the binary nature of the IBM. It processes all (f, t, i) through multiplication by $M_{ij}^{\mathbf{IRM}} \in [0,1]$ defined as:

   $$M_{ij}^{\mathbf{IRM}}(f,t) = \frac{v_{ij}(f,t)}{\sum_{j'} v_{ij'}(f,t)}, \tag{3}$$

   where $v_{ij}(f,t) = |y_{ij}(f,t)|^{\alpha}$ is the fractional power spectrogram of the source image y_{ij}. Particular cases include the *IRM2* Wiener filter for $\alpha = 2$ and the *IRM1* magnitude ratio mask for $\alpha = 1$.
3. The **Multichannel Wiener Filter** (*MWF*, [5]) exploits multichannel information, while IBM and IRM do not. $M_{j}^{\mathbf{MWF}}(f,t)$ is a $I \times I$ complex matrix given by:

   $$M_{j}^{\mathbf{MWF}}(f,t) = C_j(f,t)\, C_x^{-1}(f,t), \tag{4}$$

   where $C_j(f,t)$ is the $I \times I$ covariance matrix for source j at TF bin (f,t) and $C_x = \sum_j C_j$. In the classical local Gaussian model [5], the further parameterization $C_j(f,t) = v_j(f,t)\, R_j(f)$ is picked, with R_j being the $I \times I$ *spatial covariance matrix*, encoding the average correlations between channels at frequency bin f, and $v_j(f,t) \geq 0$ encoding the power spectral density at (f,t). The optimal values for these parameters are easily computed from the true sources y_j [8].

These three oracle systems have been implemented in Python and released in an open-source license[2].

3 Data and metrics

3.1 The MUSDB18 Dataset

For the organization of the present SiSEC, the MUSDB18 corpus was released [12], that comprises tracks from MedleyDB [4], DSD100 [11,9], and other material. In total, it features 150 full-length tracks for approximately 10 h of audio. Its noticeable features are the following.

- All items are full-length tracks, so that the handling of long-term musical structures, and of silent regions in the lead/vocal signal, can be evaluated.
- It only features stereo signals which were mixed using professional digital audio workstations. This results in quality stereo mixes which are representative of real application scenarios.

[2] `github.com/sigsep/sigsep-mus-oracle`

- As for the previous SiSEC official dataset DSD100, all signals are split into 4 predefined categories: bass, drums, vocals, and other. This contrasts with the finer granularity of MedleyDB, but promotes automation of the algorithms.
- Many musical genres are represented in MUSDB, for example, jazz, electro, metal, etc.
- It is split into a development (100 tracks, 6.5 h) and a test dataset (50 tracks, 3.5 h), for the design of data-driven separation methods.

The dataset is freely available for download online, along with Python development tools[3].

3.2 BSS Eval version 4

The BSS Eval metrics, as implemented in the MATLAB toolboxes [6,16] are widely used in the audio separation literature. They quantify the discrepancies between true sources and their estimates through 3 criteria: Source to Distortion, to Artefact, to Interference ratios (SDR, SAR, SIR) and additionally with the Image to Spatial distortion (ISR) for the `BSS Eval v3` toolbox [16].

One particularity of BSS Eval is to compute the metrics after optimally matching the estimates to the true sources through linear *distortion filters*. This arguably allows the criteria to be robust to some linear mismatches. Apart from the optional computation of all possible permutations of the sources, this matching is the reason for most of the computation cost of BSS Eval, especially considering it is done for each evaluation window when the metrics are computed on a framewise basis.

In this SiSEC, we decided to drop the assumption that distortion filters could be varying over time, but considered instead they are fixed for the whole length of the track. First, this significantly reduces the computational cost for evaluation because matching needs to be done only once for the whole signal. Second, this introduces much more dynamics in the evaluation, because time-varying matching filters turn out to over-estimate performance, as we show in Section 4.1. Third, this makes matching more robust, because true sources are not silent throughout the whole recording, while they often were for short windows.

This new 4^{th} version for the `BSS Eval` toolbox was implemented in Python, and allows either time-invariant or time-varying distortion filters. In the latter case, it is fully compatible with earlier MATLAB-based versions up to a tolerance of 10^{-12} dB. It may be found through classical Python package manager or on the dedicated website[4].

4 Separation results

4.1 Oracle performance with `BSS Eval v4`

To the best of our knowledge, the results presented in Figure 1 are the first fair comparison between the different and widely used oracle systems presented in

[3] `https://sigsep.github.io/musdb`

[4] `pip install bsseval` or `bass-db.gforge.inria.fr/bss_eval/`

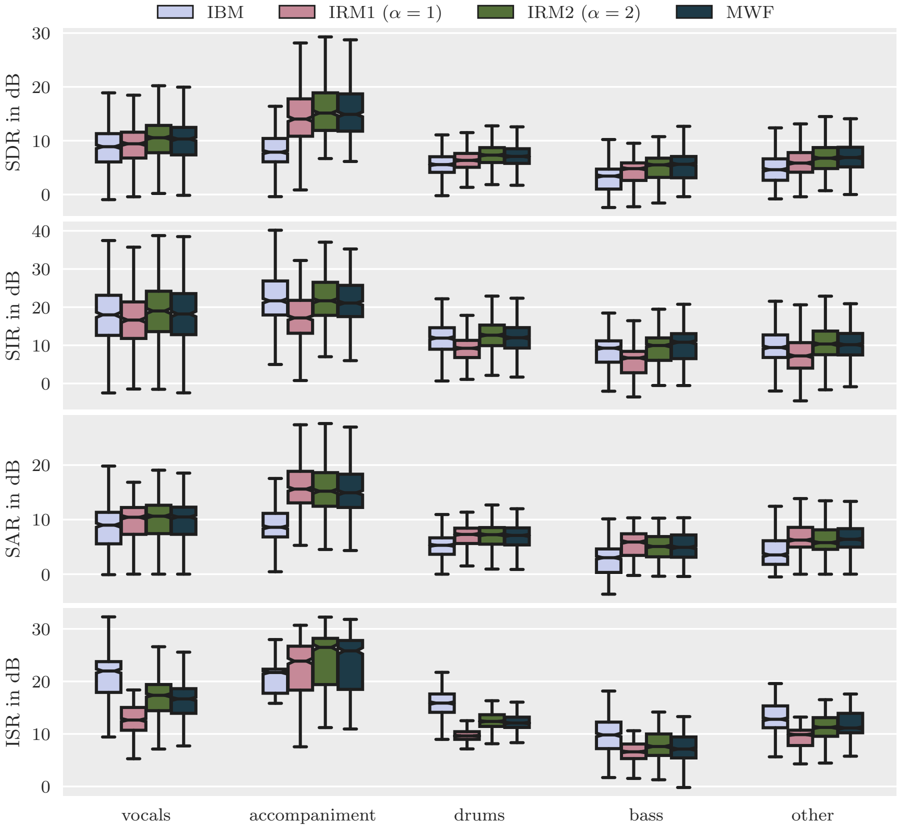

Fig. 1. BSSeval scores obtained by the 4 oracle methods considered over the MUSDB18 test set.

Section 2. On this figure, we can see boxplots of the `BSS Eval` scores obtained by IBM, IRM1, IRM2 and MWF on the 4 sources considered in MUSDB18. The scores were computed on 1 second windows, taken on the whole test-set of MUSDB18.

The most striking fact we see on this Figure 1 is that IBM is *not* achieving the best scores on any metric except ISR. Most particularly, we notice that IBM systematically induces a small loss in performance of a few dBs on SDR and SIR compared to soft masks for most sources, and to a significant loss for SAR, that can get as bad as around 5 dB for the accompaniment source. This is in line with the presence of strong *musical noise* produced by IBM whenever the source to separate is *dense* and cannot be assumed stronger in magnitude than all others whenever it is active. This also happens for the bass, which is usually weaker than all other sources at high frequencies, yielding significant distortion with IBM. Furthermore, we suspect the strong scores obtained by IBM in vocals and bass ISR to mostly be due to the zeroing of large amounts of frequency bands in those estimates. Indeed, zero estimates lead the projection filters of BSS eval to totally cancel those frequencies in the reference also, artificially boosting performance.

Now, comparing soft masks, it appears that IRM2 and MWF produce the best overall performance as compared to IRM1. However, this result is expected: `BSS Eval` scores are *in fine* relative to squared-error criteria, which are precisely optimised with those filters. Previous perceptual studies showed that IRM1 may be preferred in some cases [7]. This may be reflected in the slightly better performance that IRM1 obtains for SAR.

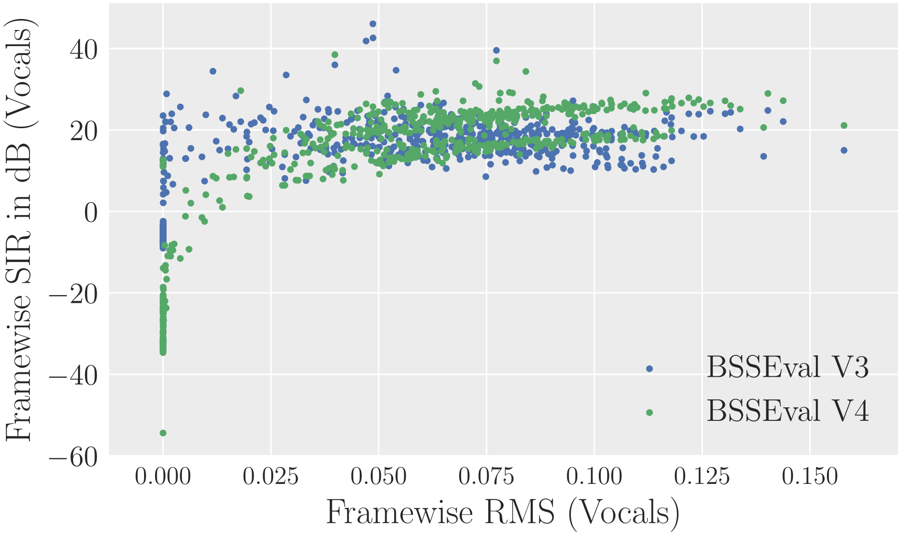

Fig. 2. Vocals SIR score vs vocals energy for BSS eval v3 and v4.

Concerning the discrepancies between`BSS Eval` v3 and v4 (time-invariant distortion filters), we observe several differences. First, computations were 8 times faster for v4 than for v3, which allowed using small 1 s frames and thus get an estimate of the performance along time at a reasonable computing cost. Second, computing distortion filters only once for the whole duration of the signal brings an interesting side-effect that can be visualized on Figure 2.The new v4 brings a much higher dynamics for the scores: we clearly see that lower energy for the true source brings lower performance. However, the marginal distributions for the scores over the whole dataset were not statistically different between v3 and v4, which validates the use of fewer distortion filters to optimize computing time and get to similar conclusions.

4.2 Comparison of systems submitted to SiSEC-MUS 2018

To be completed when the results have been submitted, for the camera-ready version.

4.3 Comparison of systems submitted to SiSEC-ASY 2018

To be completed when the results have been submitted, for the camera-ready version.

5 Conclusion

We reported our work on the organization of SiSEC 2018, that comprised the development of a new Python version 4 for BSS Eval to assess performance, that is fully compatible with earlier MATLAB versions and additionally allows for time-invariant distortion filters, significantly reducing computational load. Furthermore, we presented the new MUSDB'18 dataset, that gathers 150 music tracks with isolated stems, totaling almost 10 h of music. Finally, we also provide open-source implementations of 3 popular oracle methods to provide various upper bounds for performance.

Then, we reported the impact of choosing time-invariant distortion filters for BSS Eval over time-varying ones and quickly summarized the discrepancies in the performance of the proposed oracle methods obtained with BSS Eval v3 and v4.

Finally, we provided an overall presentation of the scores obtained by the participants to this year's edition. More detailed analysis and sound excerpts can be accessed online on the SiSEC webpage.

References

1. Shoko Araki, Francesco Nesta, Emmanuel Vincent, Zbyněk Koldovský, Guido Nolte, Andreas Ziehe, and Alexis Benichoux. *The 2011 Signal Separation Evaluation Campaign (SiSEC2011): - Audio Source Separation -*, pages 414–422. 2012.
2. Jon Barker, Ricard Marxer, Emmanuel Vincent, and Shinji Watanabe. The third chimespeech separation and recognition challenge: Dataset, task and baselines. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 504–511. IEEE, 2015.
3. Jon Barker, Emmanuel Vincent, Ning Ma, Heidi Christensen, and Phil Green. The pascal chime speech separation and recognition challenge. *Computer Speech & Language*, 27(3):621–633, 2013.
4. Rachel Bittner, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, , and Juan P. Bello. MedleyDB: A multitrack dataset for annotation-intensive mir research. In *15th International Society for Music Information Retrieval Conference*, Taipei, Taiwan, October 2014.
5. Ngoc Q. K. Duong, Emmanuel Vincent, and Rémi Gribonval. Under-determined reverberant audio source separation using a full-rank spatial covariance model. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(7):1830–1840, September 2010.

6. Cédric Févotte, Rémi Gribonval, and Emmanuel Vincent. Bss_eval toolbox user guide–revision 2.0. 2005.
7. Antoine Liutkus and Roland Badeau. Generalized Wiener filtering with fractional power spectrograms. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, QLD, Australia, April 2015.
8. Antoine Liutkus, Roland Badeau, and Gaël Richard. Low bitrate informed source separation of realistic mixtures. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 66–70. IEEE, 2013.
9. Antoine Liutkus, Fabian-Robert Stöter, Zafar Rafii, Daichi Kitamura, Bertrand Rivet, Nobutaka Ito, Nobutaka Ono, and Julie Fontecave. The 2016 signal separation evaluation campaign. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 323–332. Springer, 2017.
10. Nobutaka Ono, Zbyněk Koldovský, Shigeki Miyabe, and Nobutaka Ito. The 2013 signal separation evaluation campaign. In *2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Sept 2013.
11. Nobutaka Ono, Zafar Rafii, Daichi Kitamura, Nobutaka Ito, and Antoine Liutkus. The 2015 signal separation evaluation campaign. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 387–395. Springer, 2015.
12. Zafar Rafii, Antoine Liutkus, Fabian-Robert Stter, Stylianos Ioannis Mimilakis, and Rachel Bittner. The MUSDB18 corpus for music separation, December 2017.
13. Emmanuel Vincent, Shoko Araki, and Pau Bofill. The 2008 signal separation evaluation campaign: A community-based approach to large-scale evaluation. In *International Conference on Independent Component Analysis and Signal Separation*, pages 734–741. Springer, 2009.
14. Emmanuel Vincent, Shoko Araki, Fabian Theis, Guido Nolte, Pau Bofill, Hiroshi Sawada, Alexey Ozerov, Vikrham Gowreesunker, Dominik Lutter, and Ngoc QK Duong. The signal separation evaluation campaign (2007–2010): Achievements and remaining challenges. *Signal Processing*, 92(8):1928–1936, 2012.
15. Emmanuel Vincent, Jon Barker, Shinji Watanabe, Jonathan Le Roux, Francesco Nesta, and Marco Matassoni. The second chimespeech separation and recognition challenge: Datasets, tasks and baselines. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 126–130. IEEE, 2013.
16. Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte. Performance measurement in blind audio source separation. *IEEE transactions on audio, speech, and language processing*, 14(4):1462–1469, 2006.
17. Emmanuel Vincent, Rémi Gribonval, and Mark D Plumbley. Oracle estimators for the benchmarking of source separation algorithms. *Signal Processing*, 87(8):1933–1950, 2007.
18. Emmanuel Vincent, Hiroshi Sawada, Pau Bofill, Shoji Makino, and Justinian P Rosca. First stereo audio source separation evaluation campaign: data, algorithms and results. In *International Conference on Independent Component Analysis and Signal Separation*, pages 552–559. Springer, 2007.
19. DeLiang Wang. On ideal binary mask as the computational goal of auditory scene analysis. *Speech separation by humans and machines*, pages 181–197, 2005.