Open-Unmix - A Reference Implementation for Music Source Separation

Fabian-Robert Stöter, Stefan Uhlich, Antoine Liutkus, Yuki Mitsufuji

The Journal of Open Source Software

Open-Unmix - A Reference Implementation for Music Source Separation

Fabian-Robert Stöter[1], **Stefan Uhlich**[2], **Antoine Liutkus**[1], **and Yuki Mitsufuji**[3]

1 Inria and LIRMM, University of Montpellier, France **2** Sony Europe B.V., Germany **3** Sony Corporation, Japan

DOI: 10.21105/joss.01667

Software

- Review
- Repository
- Archive

Submitted: 17 August 2019
Published: 08 September 2019

Summary

Music source separation is the task of decomposing music into its constitutive components, e.g., yielding separated stems for the vocals, bass, and drums. Such a separation has many applications ranging from rearranging/repurposing the stems (remixing, repanning, upmixing) to full extraction (karaoke, sample creation, audio restoration). Music separation has a long history of scientific activity as it is known to be a very challenging problem. In recent years, deep learning-based systems - for the first time - yielded high-quality separations that also lead to increased commercial interest. However, until now, no open-source implementation that achieves state-of-the-art results is available. *Open-Unmix* closes this gap by providing a reference implementation based on deep neural networks. It serves two main purposes. Firstly, to accelerate academic research as *Open-Unmix* provides implementations for the most popular deep learning frameworks, giving researchers a flexible way to reproduce results. Secondly, we provide a pre-trained model for end users and even artists to try and use source separation. Furthermore, we designed *Open-Unmix* to be one core component in an open ecosystem on music separation, where we already provide open datasets, software utilities, and open evaluation to foster reproducible research as the basis of future development.

Background

Music separation is a problem which has fascinated researchers for over 50 years. This is partly because, mathematically, there exists no closed-form solution when many sources (instruments) are recorded in a mono or stereo signal. To address the problem, researchers exploited additional knowledge about the way the signals were recorded and mixed. A large number of these methods are centered around "classical" signal processing methods. For a more detailed overview see (Rafii, Liutkus, Stöter, Mimilakis, & Bittner, 2017) and (Cano, FitzGerald, Liutkus, Plumbley, & Stöter, 2019). Many of these methods were hand-crafted and tuned to a small number of music recordings (Araki et al., 2012; Ono, Koldovsky, Miyabe, & Ito, 2013; Vincent et al., 2012). Systematic objective evaluation of these methods, however, was hardly feasible as freely available datasets did not exist at that time. In fact, for a meaningful evaluation, the ground truth separated stems are necessary. However, because commercial music is usually subject to copyright protection, and the separated stems are considered to be valuable assets in the music recording industry, they are usually unavailable.

Nonetheless, thanks to some artists who choose licenses like Creative Commons, that allow sharing of the stems, freely available datasets were released in the past five years and have enabled the development of data-driven methods. Since then, progress in performance has

been closely linked to the availability of more data that allowed the use of machine-learning-based methods. This led to a large performance boost similar to other audio tasks such as automatic speech recognition (ASR) where a large amount of data was available. In fact, in 2016 the speech recognition community had access to datasets with more than 10000 hours of speech (Amodei et al., 2016). In contrast, at the same time, the *MUSDB18* dataset was released (Rafii et al., 2017) which comprises 150 full-length music tracks – a total of just 10 hours of music. To date, this is still the largest freely available dataset for source separation. Nonetheless, even with this small amount of data, deep neural networks (DNNs) were not only successfully used for music separation but they are now setting the state-of-the-art in this domain as can be seen by the results of the community-based signal separation evaluation campaign (SiSEC) (Liutkus et al., 2017; Ono, Rafii, Kitamura, Ito, & Liutkus, 2015; Stöter, Liutkus, & Ito, 2018). In these challenges, the proposed systems are compared to other methods. Among the systems under test, classical signal processing based methods were clearly outperformed by machine learning methods. However they were still useful as a *fast* and often *simple to understand* baseline.

In the following, we will describe a number of these reference implementations for source separation. While there are some commercial systems available, such as *Audionamix XTRAX STEMS*, *IZOTOPE RX 7* or *AudioSourceRE*, we only considered tools that are available as open-source software, and are suitable for research.

The first publicly available software for source separation was *openBlissart*, released in 2011 (Weninger, Lehmann, & Schuller, 2011). It is written in C++ and accounts for the class of systems that are based on non-negative matrix factorization (NMF). In 2012, the *Flexible Audio Source Separation Toolbox (FASST)* was presented in (Ozerov, Vincent, & Bimbot, 2011; Salaün et al., 2014). It is written in MATLAB/C++ and is also based on NMF methods, but also includes other model-based methods. In 2016, the *untwist* library was proposed in (Roma, Grais, Simpson, Sobieraj, & Plumbley, 2016). It comprises several methods, ranging from classical signal-processing-based methods to feed-forward neural networks. The library is written in Python 2.7. Unfortunately, it has not been updated since 2017 and many of its methods are not subjected to automated testing. *Nussl* is a very recent framework, presented in (Manilow, Seetharaman, & Pardo, 2018). It includes a large number of methods and generally focuses on classical signal processing methods rather than machine-learning-based techniques. It has built-in interfaces for common evaluation metrics and data sets. The library offers great modularity and a good level of abstraction. However, this also means that it is challenging for beginners who might only want to focus on changing the machine learning parts of the techniques.

The main problem with these implementations is that they do not deliver state-of-the-art results. No open-source system is available today that matches the performance of the best system proposed more than four years ago by (Uhlich, Giron, & Mitsufuji, 2015). We believe that the lack of such a baseline has a serious negative impact on future research on source separation. Many new methods that were published in the last few years are usually compared to their own baseline implementations, thus showing relative instead of absolute performance gains, so that other researchers cannot assess if a method performs as well as state-of-the-art. Also, the lack of a common reference for the community potentially misguides young researchers and students who enter the field of music separation. The result of this can be observed by looking at the popularity of the above-mentioned music separation frameworks on GitHub: all of the frameworks mentioned above, combined, are less popular than two recent deep learning papers that were accompanied by code such as `MTG/DeepConvSep` from (Chandna, Miron, Janer, & Gómez, 2017) and `f90/Wave-U-Net` from (Stoller, Ewert, & Dixon, 2018). Thus, users might be confused regarding which of these implementations can be considered state-of-the-art.

Open-Unmix

We propose to close this gap with *Open-Unmix*, which applies machine learning to the specific tasks of music separation. With the rise of simple to use machine learning frameworks such as *Pytorch*, *Keras*, *Tensorflow* or *NNabla*, the technical challenge of developing a music separation system appears to be very low at first glance. However, the lack of domain knowledge about the specifics of music signals often results in poor performance where issues are difficult to track using learning-based algorithms. We therefore designed *Open-Unmix* to address these issues by relying on procedures that were verified by the community or have proven to work well in the literature.

Design Choices

The design choices made for *Open-Unmix* have sought to reach two somewhat contradictory objectives. Its first aim is to have state-of-the-art performance, and its second aim is to still be easily understandable, so that it can serve as a basis for research to allow improved performance in the future. In the past, many researchers faced difficulties in pre- and post-processing that could be avoided by sharing domain knowledge. Our aim was thus to design a system that allows researchers to focus on A) new representations and B) new architectures.

Framework specific vs. framework agnostic

We choose *PyTorch* to serve as a reference implementation due to its balance between simplicity and modularity (Stöter & Liutkus, 2019a). Furthermore, we already ported the core model to NNabla and plan to release a port for Tensorflow 2.0, once the framework is released. Note that the ports will not include pre-trained models as we cannot make sure the ports would yield identical results, thus leaving a single baseline model for researchers to compare with.

"MNIST-like"

Keeping in mind that the learning curve can be quite steep in audio processing, we did our best for *Open-unmix* to be:

- **simple to extend**: The pre/post-processing, data-loading, training and models part of the code is isolated and easy to replace/update. In particular, a specific effort was done to make it easy to replace the model.
- **not a package**: The software is composed of largely independent and self-containing parts, keeping it easy to use and easy to change.
- **hackable (MNIST like)**: Due to our objective of making it easier for machine-learning experts to try out music separation, we did our best to stick to the philosophy of baseline implementations for this community. In particular, *Open-unmix* mimics the famous MNIST example, including the ability to instantly start training on a dataset that is automatically downloaded.

Reproducible

Releasing *Open-Unmix* is first and foremost an attempt to provide a reliable implementation sticking to established programming practice as were also proposed in (McFee et al., 2018). In particular:

- **reproducible code**: everything is provided to exactly reproduce our experiments and display our results.

- **pre-trained models**: we provide pre-trained weights that allow a user to use the model right away or fine-tune it on user-provided data (Stöter & Liutkus, 2019b, 2019c).
- **tests**: the release includes unit and regression tests, useful to organize future open collaboration using pull requests.

Results

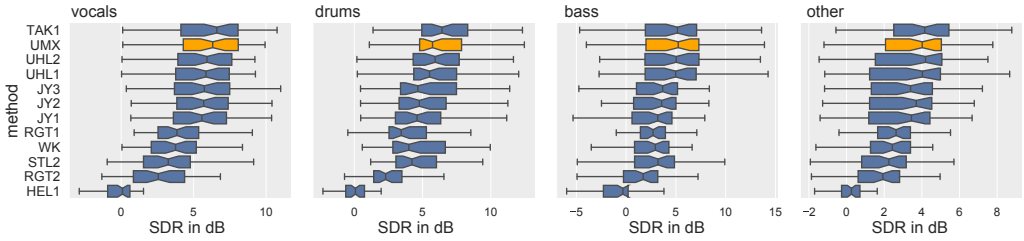

Figure 1: Boxplots of evaluation results of the `UMX` model compared with other methods from (Stöter et al., 2018) (methods that did not only use MUSDB18 for training were omitted)

Open-Unmix is based on the bi-directional LSTM model from (Uhlich et al., 2017) and we compared it to other separation models that were submitted to the last SiSEC contest (Stöter et al., 2018). The results of `UMX` are depicted in 1. It can be seen that our proposed model reaches state-of-the-art results. There is no statistically significant difference between the best method `TAK1` and `UMX`. Because `TAK1` is not released as open-source, this indicates that *Open-Unmix* is the current state-of-the-art open-source source separation system.

Community

Open-Unmix was developed by Fabian-Robert Stöter and Antoine Liutkus at Inria Montpellier. The research concerning the deep neural network architecture as well as the training process was done in close collaboration with Stefan Uhlich and Yuki Mitsufuji from Sony Corporation.

In the future, we hope the software will be well received by the community. *Open-Unmix* is part of an ecosystem of software, datasets, and online resources: the **sigsep** community.

First, we provide MUSDB18 (Rafii et al., 2017) and MUSDB18-HQ (Rafii, Liutkus, Stöter, Mimilakis, & Bittner, 2019) which are the largest freely available datasets; this comes with a complete toolchain to easily parse and read the datasets (Stöter & Liutkus, 2019a). We maintain *museval*, the most used evaluation package for source separation (Stöter & Liutkus, 2019b). We also are the organizers of the largest source separation evaluation campaign such as (Stöter et al., 2018). In addition, we implemented a reference implementation using a multi-channel Wiener filter, released in (Liutkus & Stöter, 2019). The `sigsep` community is organized and presented on its own website. *Open-Unmix* itself can be found on https://open.unmix.app, which links to all other relevant sites and provides further information, such as audio demos.

Outlook

Open-Unmix is a community-focused project. We therefore encourage the community to submit bug-fixes and comments and improve the computational performance. However, we are not looking for changes that only focus on improving the separation performance as this would be out of scope for a baseline implementation. Instead, we expect many researchers

will fork the software as a basis for their research and the documentation explicates several custom options to extend the code (shown here).

References

Amodei, D., Anubhai, R., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Chen, J., et al. (2016). Deep speech 2: End-to-end speech recognition in English and Mandarin. In *ICML* (pp. 173–182).

Araki, S., Nesta, F., Vincent, E., Koldovsky, Z., Nolte, G., Ziehe, A., & Benichoux, A. (2012). The 2011 signal separation evaluation campaign (SiSEC2011): - audio source separation -. In *10th international conference on latent variable analysis and signal separation*. doi:10.1007/978-3-642-28551-6_51

Cano, E., FitzGerald, D., Liutkus, A., Plumbley, M. D., & Stöter, F. (2019). Musical source separation: An introduction. *IEEE Signal Processing Magazine*, *36*(1), 31–40. doi:10.1109/MSP.2018.2874719

Chandna, P., Miron, M., Janer, J., & Gómez, E. (2017). Monoaural audio source separation using deep convolutional neural networks. In *Latent variable analysis and signal separation* (pp. 258–266). doi:10.1007/978-3-319-53547-0_25

Liutkus, A., & Stöter, F.-R. (2019, September). sigsep/norbert: v0.2.1. doi:10.5281/zenodo.3386463

Liutkus, A., Stöter, F.-R., Rafii, Z., Kitamura, D., Rivet, B., Ito, N., Ono, N., et al. (2017). The 2016 signal separation evaluation campaign. In *Proc. Intl. Conference on latent variable analysis and signal separation (lva/ica)* (pp. 323–332). Springer International Publishing. doi:10.1007/978-3-319-53547-0_31

Manilow, E., Seetharaman, P., & Pardo, B. (2018). The northwestern university source separation library. In *ISMIR* (pp. 297–305).

McFee, B., Kim, J. W., Cartwright, M., Salamon, J., Bittner, R. M., & Bello, J. P. (2018). Open-source practices for music signal processing research: Recommendations for transparent, sustainable, and reproducible audio research. *IEEE Signal Processing Magazine*, *36*(1), 128–137. doi:10.1109/MSP.2018.2875349

Ono, N., Koldovsky, Z., Miyabe, S., & Ito, N. (2013). The 2013 signal separation evaluation campaign. In *Proc. IEEE international workshop on machine learning for signal processing (MLSP)* (pp. 1–6). doi:10.1109/MLSP.2013.6661988

Ono, N., Rafii, Z., Kitamura, D., Ito, N., & Liutkus, A. (2015). The 2015 signal separation evaluation campaign. In *Proc. Intl. Conference on latent variable analysis and signal separation (lva/ica)*. Liberec, Czech Republic, doi:10.1007/978-3-319-22482-4_45

Ozerov, A., Vincent, E., & Bimbot, F. (2011). A general flexible framework for the handling of prior information in audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, *20*(4), 1118–1133. doi:10.1109/TASL.2011.2172425

Rafii, Z., Liutkus, A., Stöter, F.-R., Mimilakis, S. I., & Bittner, R. (2017, December). MUSDB18, a corpus for audio source separation. doi:10.5281/zenodo.1117372

Rafii, Z., Liutkus, A., Stöter, F.-R., Mimilakis, S. I., & Bittner, R. (2019, August). MUSDB18-hq - an uncompressed version of musdb18. doi:10.5281/zenodo.3338373

Roma, G., Grais, E. M., Simpson, A., Sobieraj, I., & Plumbley, M. D. (2016). Untwist: A new toolbox for audio source separation. In *Extended abstracts for the late-breaking demo session of the 17th international society for music information retrieval conference, ismir* (pp. 7–11).

Salaün, Y., Vincent, E., Bertin, N., Souviraà-Labastie, N., Jaureguiberry, X., Tran, D. T., & Bimbot, F. (2014, May). The Flexible Audio Source Separation Toolbox Version 2.0. ICASSP. Retrieved from https://hal.inria.fr/hal-00957412

Stoller, D., Ewert, S., & Dixon, S. (2018). Wave-u-net: A multi-scale neural network for end-to-end audio source separation. *arXiv preprint arXiv:1806.03185*.

Stöter, F.-R., & Liutkus, A. (2019a, August). sigsep/open-unmix-pytorch: Initial release of Open-Unmix. doi:10.5281/zenodo.3382104

Stöter, F.-R., & Liutkus, A. (2019b, August). Open-unmix-pytorch umx. doi:10.5281/zenodo.3370486

Stöter, F.-R., & Liutkus, A. (2019c, August). Open-unmix-pytorch umx-hq. doi:10.5281/zenodo.3370489

Stöter, F.-R., & Liutkus, A. (2019a, July). sigsep/sigsep-mus-db: v0.1.7. doi:10.5281/zenodo.3271451

Stöter, F.-R., & Liutkus, A. (2019b, June). sigsep/sigsep-mus-eval: v0.3.0. doi:10.5281/zenodo.3261102

Stöter, F.-R., Liutkus, A., & Ito, N. (2018). The 2018 signal separation evaluation campaign. In *Latent variable analysis and signal separation: 14th international conference, lva/ica 2018, surrey, uk* (pp. 293–305). doi:10.1007/978-3-319-93764-9_28

Uhlich, S., Giron, F., & Mitsufuji, Y. (2015). Deep neural network based instrument extraction from music. In *Icassp* (pp. 2135–2139). doi:10.1109/ICASSP.2015.7178348

Uhlich, S., Porcu, M., Giron, F., Enenkl, M., Kemp, T., Takahashi, N., & Mitsufuji, Y. (2017). Improving music source separation based on deep neural networks through data augmentation and network blending. In *Icassp*. New Orleans, LA, USA. doi:10.1109/ICASSP.2017.7952158

Vincent, E., Araki, S., Theis, F. J., Nolte, G., Bofill, P., Sawada, H., Ozerov, A., et al. (2012). The signal separation evaluation campaign (2007-2010): Achievements and remaining challenges, *92*(8), 1928–1936. doi:10.1016/j.sigpro.2011.10.007

Weninger, F., Lehmann, A., & Schuller, B. (2011). OpenBliSSART: Design and evaluation of a research toolkit for blind source separation in audio recognition tasks. In *Proc. IEEE intl. Conf. On acoustics, speech and signal processing (icassp)* (pp. 1625–1628). doi:10.1109/ICASSP.2011.5946809