FEATURE-INFORMED LATENT SPACE REGULARIZATION FOR MUSIC SOURCE SEPARATION

Yun-Ning Hung and Alexander Lerch

Music Informatics Group
Georgia Institute of Technology
Atlanta, USA
{yhung33, alexander.lerch}@gatech.edu

ABSTRACT

The integration of additional side information to improve music source separation has been investigated numerous times, e.g., by adding features to the input or by adding learning targets in a multi-task learning scenario. These approaches, however, require additional annotations such as musical scores, instrument labels, etc. in training and possibly during inference. The available datasets for source separation do not usually provide these additional annotations. In this work, we explore transfer learning strategies to incorporate VGGish features with a state-of-the-art source separation model; VGGish features are known to be a very condensed representation of audio content and have been successfully used in many music information retrieval tasks. We introduce three approaches to incorporate the features, including two latent space regularization methods and one naive concatenation method. Our preliminary results show that our proposed approaches could improve some evaluation metrics for music source separation. In this work, we also include a discussion of our proposed approaches, such as the pros and cons of each approach, and the potential extension/improvement.

1. INTRODUCTION

Music source separation has been an intensively studied problem due to its numerous applications. By isolating the sound of individual instruments from a mixture of instruments, source separation systems have been used, e.g., for audio-remixing [1], instrument-wise equalization [2], accompaniment generation for Karaoke systems [3], or singer identification [4].

A typical music source separation pipeline often includes input representation (e.g., waveform and extracted features), the machine learning model (e.g., neural network and loss function), and the post-processing algorithm (e.g., Wiener filtering). Although there has been some work on the input representation to improve source separation systems, e.g., investigating waveforms [5] or complex spectrograms [6] instead of the common magnitude spectrograms, most research in recent years has focused on improving source separation models through new, more powerful model architectures [6, 7, 8]. One successful architecture is the U-net [9] that has been adopted and utilized for many source separation studies [5, 10, 11, 12, 13]. While the original U-net is based on a Convolutional Neural Network (CNN) and skip connections, an advanced U-net architecture proposed by Takahashi et al. combines the CNN with a Recurrent Neural Network (RNN) [7]. Different combination of CNN, RNN, and Fully-Connected (FC) layers have been studied by Choi et al. [6]. The popular Open-Unmix and the more recent X-UMX models are based on a combination of RNN and FC [8, 14].

While the increasing model complexity has led to noticeable increases in the quality of the system outputs, it has also led to an increased need for computational resources particularly during training. Researchers often cannot easily meet the GPU and memory requirements for training or inference with modern source separation systems. This resulted in a parallel research direction aiming at improving a source separation model by adding additional information during training, inference, or both [15, 16, 17, 18], which is a contrasting approach compared to increasing model complexity.

However, one drawback of these methods is that they need additional ground truth annotations for training or sometimes even during inference. The available datasets for music source separation —most prominently the MUSDB18 dataset [19]— do not provide these additional annotations. While generating pseudo-annotations with other machine learning models can be one solution, their usefulness depends on the correctness of the model. Furthermore, the computation time will increase with more added features.

In this work, we propose to leverage a feature representation learned from large-scale datasets as additional information. This feature representation includes more generalized features than single task-specific models. To this end, we leverage the well-known VGGish features [20]. The VGGish model is a very deep model trained on a very large audio dataset and provides a condensed feature representation of the input audio file. We investigate three methods to incorporate VGGish features into a state-of-the-art (SOTA) music source separation model, including two latent space regularization methods. In summary, this study explores:

- leveraging the information contained by mid-level features trained on a different task to improve music source separation without the need of additional ground truth annotations, and
- restricting the use of such additional features to the training in order to preserve computational resources during inference.

In the following sections, we first review existing approaches on using VGGish features for music information retrieval (MIR) tasks and incorporating additional features in music source separation. Section 3 provides a detailed overview of the methods we propose for our feature-informed source separation system. Section 4 presents the evaluation results for both, a state-of-the-art

(SOTA) system and our proposed system. The same section also presents a detailed analysis of the latent space after regularization.

2. RELATED WORK

This section discusses two related research directions that inspired our proposed approach. First, we introduce transfer learning methods that leverage representations from models pre-trained on larger datasets to help downstream tasks. Then, the literature on adding additional information that supports a target task is briefly surveyed.

2.1. VGGish features

Traditional hand-crafted features have nowadays been replaced by feature representations learned automatically from the training data. These learned representations can also be used in other tasks in various domains. The pre-trained features from the BERT model [21], for example, have been successfully used in multiple natural language processing tasks such as question answering and language inference. In the image domain, large-scale pre-trained models such as AlexNet [22] and VGG-16 [23] have achieved competitive results in image classification and assist in many visual feature extraction tasks [24]. There exist several pre-trained models in the audio domain, such as L^3-Net [25], VGGish [20] and SoundNet [26], which leverage both audio and visual information provided by video to train an audio feature extraction network.

VGGish features, particularly, have been successfully used in numerous MIR tasks. These tasks include weakly-supervised instrument recognition [27], cross-modal representation learning [28, 29], music auto-tagging [30], music emotion recognition [31], and music genre classification [32, 33, 34]. The VGGish model was pre-trained on a larger dataset (i.e., YouTube-8M [35]) than most other audio feature extractors; this potentially gives this representation higher discriminative power.

The apparent popularity of VGGish features combined with the variety of the tasks they have successfully used this input representation implies that these features are able to capture many task-agnostic properties of audio files suitable for a large variety of music-related tasks. Consequently, we choose the VGGish model to extract the feature representation for our experiment.

2.2. Additional features for music source separation

Leveraging additional side information to improve music source separation systems has been proposed in multiple forms. The two most common methods are (i) to add the additional information directly to the input and (ii) to utilize the additional information gained in a multi-task setup.

Taking advantage of the close relation of the music instrument classification task and music source separation, several studies have shown that instrument activity labels as an additional input to a source separation system can improve results. Slizovskaia et al. investigated several methods to add instrument labels to the U-net model [36] and Swaminathan added voice activity labels to improve singing voice separation quality [37]. Instrument activity labels can also be used as a condition to control the output sources by using one model instead of multiple models for each source [15, 38, 39]. In addition to instrument labels, the instrumentation and pitch information provided by musical scores has also been used to guide the learning process and improve separation results [40, 41, 42]. Carabias-Orti et al. learn a timbre model for each instrument they separate and use the trained models as priors for the separation system based on non-negative matrix factorization [43]. Source separation systems can also be informed by incorporating visual features, leveraging the cross-modal information of the video for the separation [44, 45, 46].

As an alternative to adding additional information to the input of a source separation system, this information can also be utilized during the training to improve the internal representation and help the model to generalize better. Hung et al. proposed to combine the training of a frame-level instrument classifier and a source separation system in a multi-task setup and then leverage the instrument predictions during inference for post-processing the result [16]. Manilow et al. were able to improve source separation with a deep clustering model using both separation and transcription as a training tasks [17] and Jansson et al. explored a variety of methods to learn singing voice separation and fundamental frequency (F0) estimation at the same time [18].

3. PROPOSED METHOD

In this work, we investigate three transfer learning approaches to improve a SOTA source separation model via VGGish features. The first approach directly integrates the VGGish information while the second and third approach perform transfer learning indirectly through the regularization of the embedding space based on the VGGish information. Our proposed methods can be categorized as transfer learning since the additional features — which are extracted from a model pre-trained on a larger dataset — provide direct or indirect knowledge transfer. Figure 1 shows a high-level overview of the proposed training pipeline, indicating the use of VGGish features (purple) to modify the latent vector (red).

Our approaches show some similarity to feature-based knowledge distillation methods used in teacher-student learning [47], where the pre-trained representations are used to regularize the embedding space during training. VGGish features might contain information that is useful for separation but not adequately represented in the unregularized latent space. Projecting the latent vectors into the VGGish feature space can help transfer the knowledge from VGGish feature space into latent vectors. Moreover, VGGish features have strong discriminative power. Forcing the latent vectors to be close to VGGish features can lead to more separable latent space representations, and prevent the model from confusing distinct instruments.

3.1. Source separation model

We adopt the X-UMX model as the baseline model [14] since this model achieves very good results on the MUSDB18 dataset and has open-sourced code [1]. The model is based on the Open-Unmix (UMX) architecture [8]. Instead of training one separate model for each instrument, X-UMX uses a bridging network architecture, connecting the paths to cross each source's network by adding two average operators to the original UMX model. The result shows 0.4 dB improvement for the average Source-to-Distortion Ratio (SDR) compared to the baseline UMX architecture [8]. As indicated in Figure 1, the model uses four separate encoders consisting of fully-connected layers and bi-directional recurrent layers to compute the latent vectors $l \in \mathbb{R}^{B \times T_l}$ where B denotes the number of feature bins for the latent vector for each instrument while T_l

[1] https://github.com/sony/ai-research-code/tree/master/x-umx

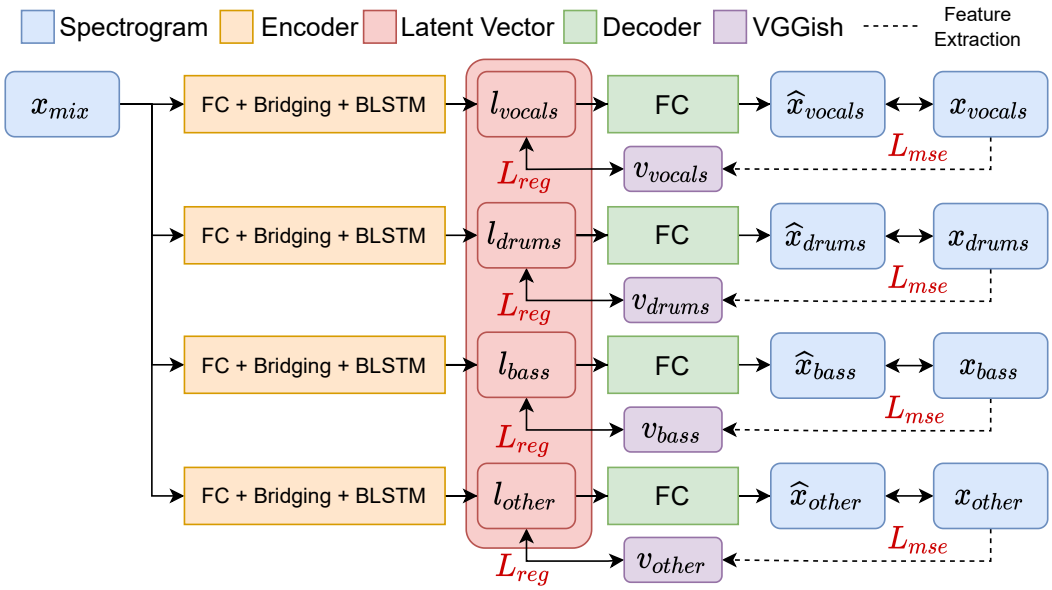


Figure 1: Overview of the X-UMX structure and our proposed training pipeline for the two proposed regularization methods. $\hat{x}$ denotes the estimated target spectrograms; x denotes the ground truth spectrograms; l denotes the latent vectors; v denotes the VGGish features; L_{mse} and L_{reg} denote the Mean Square Error and Regularization loss respectively. FC denotes fully-connected layers. The dotted line represents the feature extraction process by VGGish.

denotes the number of the latent vectors across time. The decoders comprise of fully-connected layers to decode the target spectrogram masks from the latent vectors.

3.2. Input representation and features

Following the setup of the X-UMX model, the input of the source separation model is the magnitude spectrogram $x \in \mathbb{R}^{T_s \times F}$, where T_s represents the duration of the spectrogram while F represents number of frequency bins. The short time Fourier transform is computed with a hop length and block length of 1024 and 4096 samples, respectively.

The additional VGGish features are extracted by the pre-trained VGGish model [20] as 128 -dimensional vector with 0.96 s time resolution (no overlap). The features are PCA transformed (with whitening) and quantized to 8-bits.

One obstacle encountered when incorporating the VGGish features is the difference in time resolution. The VGGish model is originally trained for clip-level tagging so the features have a low time resolution (approx. 1 s). Our source separation model needs frame-wise prediction, and therefore has a much higher resolution (approx. 0.02 s). We address this issue by simply repeating the features n times with n representing the number of time frames in 0.96 s. This approach is based on the assumption that one frame of VGGish features contains information for the entire n latent vector frames and that a slight mis-alignment in time will have a negligible effect on the results.

3.3. Transfer Learning Approaches

3.3.1. Method 1: Concatenation

The first proposed method aims to provide additional information encoded by VGGish directly to the decoder. We simply concatenate

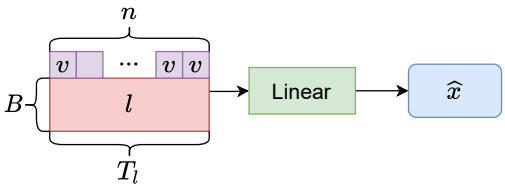


Figure 2: The proposed *Concat* method to concatenate VGGish features with latent vectors.

the VGGish features (of the mixture audio) and the latent vector along feature dimension B, as shown in Figure 2. It should be noted that this approach requires VGGish feature extraction during inference time. The original VGGish model has 72M parameters. Although the VGGish features can be extracted prior to training and thus have only minimal impact on the training time, this approach has the potential to slow down the inference time. Since in practice there is no access to the ground truth for the separated tracks during inference, the VGGish features we use in this method for both training and testing are extracted from the mixture audio. This method is referred as *Concat* in the remainder of this paper.

3.3.2. Method 2: Contrastive Regularization

The second method aims to regularize the latent space with the additional VGGish features. To do so, we utilize the VGGish features extracted from the separate ground truth tracks (e.g., bass latent vectors from the model should be close to VGGish features extracted from the bass track) and add an extra loss term $L_{\text{con-reg}}$ based on cosine similarity to force the latent vectors to be close to

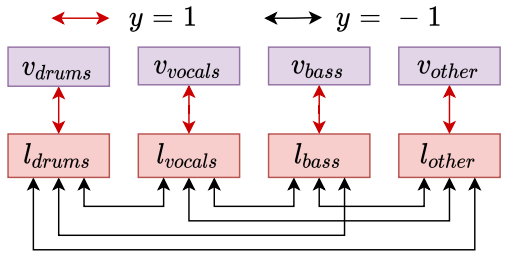


Figure 3: The proposed *Con-Reg* method to regularize latent vectors through VGGish features.

VGGish features:

$$L_{\text{con-reg}} = \begin{cases} 1 - \cos(f(l), v) & \text{if } y = 1 \\ \max(\cos(f(l), v) - \alpha, 0) & \text{if } y = -1, \end{cases} \quad (1)$$

with $y = 1$ when the latent vector l_i and the VGGish features v_i both correspond to the same instrument i (e.g., bass latent vector and bass VGGish features) and $y = -1$ in the case that l_i and v_j represent two different instruments $i \neq j$. The hyperparameter α is the margin of distance (set to $\alpha = 0.2$ after hyperparameter search). A 1D CNN with 1 kernel size (f) is applied to transform the latent vector dimensionality to 128 to match the VGGish feature dimensionality, allowing us to compute the cosine distance between l and v. This method is referred as *Con-Reg* in the remainder of this paper.

3.3.3. Method 3: Distance-based Regularization

Similar to method *Con-Reg*, the latent space is regularized with an additional loss. In this case, however, the additional loss term $L_{\text{dis-reg}}$ aims at forcing the distances between pairs of latent vectors to be similar to the distances between two corresponding VGGish features:

$$L_{\text{dis-reg}} = \max\left(D_{\text{latent}}(f(l_i), f(l_j)) - D_{\text{vgg}}(v_i, v_j), 0\right), \quad (2)$$

where D_{latent} represents the cosine distance between two latent vectors and D_{vgg} represents the distance of two VGGish feature vectors from two different instruments i and j where $i \neq j$. This method is similar to the scenario in *Con-Reg* if $y = -1$ in Equation (1). Instead of choosing a fixed value α as the margin, the distance of VGGish feature pairs serve as a 'soft' margin which indicates the lower bound of the distance of two latent vectors. Since the cosine distance is evaluated on two distinct instruments, the latent vectors should be more separated from each other, indicating that D_{latent} should be smaller than D_{vgg}

The loss design is based on the observation that the latent vectors from the same instrument category might not always be close to the same category. For example, 'Electrical Bass' and 'Double Bass' in 'Bass' category might have slightly different features. The same as 'Female Vocals' and 'Male Vocals' in the 'Vocals' categories. Since the VGGish model is pre-trained on the dataset that contains more detailed instrument labels[2], the distance of VGGish features might be able to capture the similarity of each instrument frame. As a result, instead of using the category labels y to determine the absolute distance, we use distance between the corresponding VGGish features to decide the distance of the latent vectors. Since optimizing the model to predict latent vectors matching the distance of VGGish features is not easy, we use the VGGish distance (D_{vgg}) as a margin so that the latent vectors will have a distance (D_{latent}) where $D_{\text{latent}} \leq D_{\text{vgg}}$. Distances smaller than D_{vgg} are also acceptable, since these encourage the latent vectors of two separate instruments to be more separable. This method is referred as *Dis-Reg* in the remainder of this paper.

[2]The ontology of the labels is shown in this website https://research.google.com/audioset/ontology/index.html

3.4. Training Setup

The Adam optimizer [48] is used with a 1e-4 learning rate and 1e-5 weight decay to optimize the model. Early stopping is applied if the validation loss does not decrease for 25 epochs and the learning rate decrease by a factor of 0.3 if the validation loss does not decrease for 10 epochs. The applied data augmentation (e.g., channel swapping and volume adjustment) is identical to the one described in [8]. Each input sample has a length of 6 s and is picked by randomly selecting a starting time in the audio. The standard Mean Square Error (MSE) loss and regularization loss (for *Con-Reg* & *Con-Dis*) are combined with:

$$L_{\text{total}} = L_{\text{mse}} + \lambda L_{\text{reg}}, \quad (3)$$

where L_{reg} is either $L_{\text{con-reg}}$ or $L_{\text{dis-reg}}$. The weight of the regularization loss λ is set after hyperparameter search to $\lambda_{\text{con-reg}} = 1e{-}6$ and $\lambda_{\text{dis-reg}} = 1$, respectively. Since $L_{\text{con-reg}}$ has a 'strong' margin between the latent vector and VGGish features, we found that setting the loss small to gradually influence the model during training can lead to better performance. Note that the multi-domain loss in the X-UMX model is ignored in this study even if it led to an improvement of 0.2 dB [14] for the sake of computational speed.

4. EXPERIMENTS

We train and evaluate the proposed methods on the MUSDB18 dataset [19] and use the 'train,' 'evaluation,' and 'test' split defined in the original dataset. MUSDB18 has a total of 150 full-track songs of different styles. All signals are stereophonic and encoded at 44.1kHz. Each song in the dataset is comprised of four tracks, 'Vocals,' 'Bass,' 'Drums,' and 'Other.' The X-UMX model without any regularization is our baseline for comparison. The same training strategy described above is also used to re-train X-UMX. The `Museval` toolbox is used to calculate evaluation metrics: Signal-to-Distortion Ratio (SDR), Signal-to-Interference Ratio (SIR), and Signal-to-Artifact Ratio (SAR) [49]. These three metrics are commonly used to evaluate the separation quality, the amount of other sources, and the amount of unwanted artifacts in an estimated source. Increasing values indicate better performance.

4.1. Music Source Separation

The results of the source separation experiments per instrument are given in Figure 4. We can observe that, compared to the baseline X-UMX model, our proposed methods show a tendency of suppressing unwanted artifacts and increasing the SAR scores on all instruments. The result supports our assumption that VGGish features contain additional information potentially helpful for separation. Utilizing VGGish features seems to help stabilizing the model and reducing artifacts.

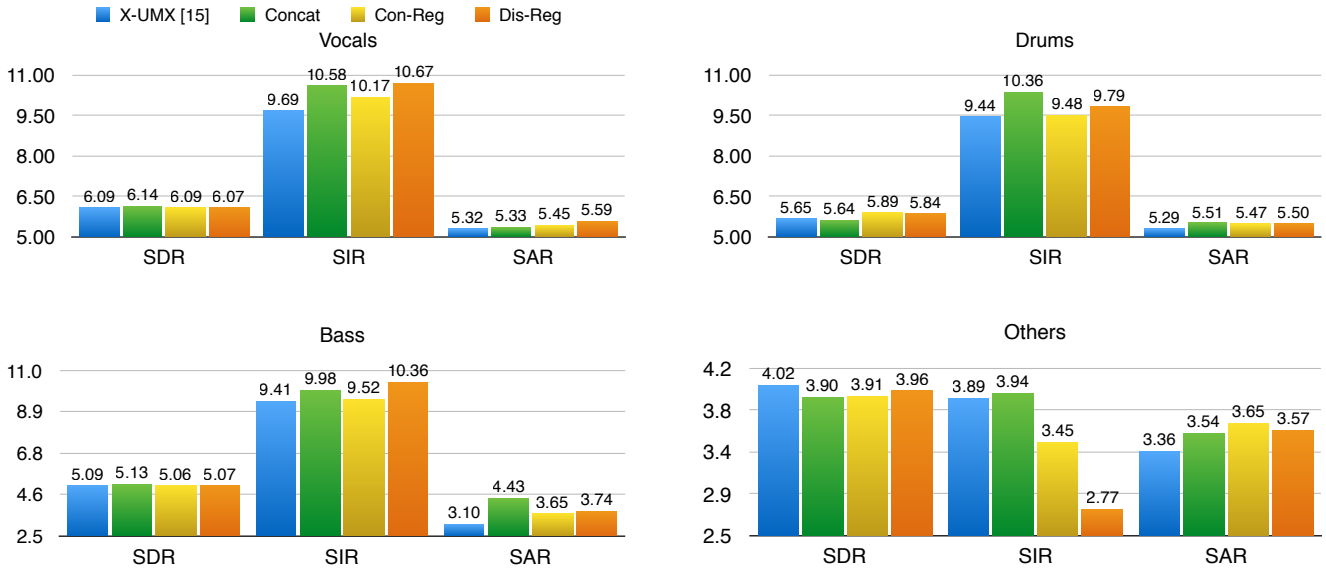
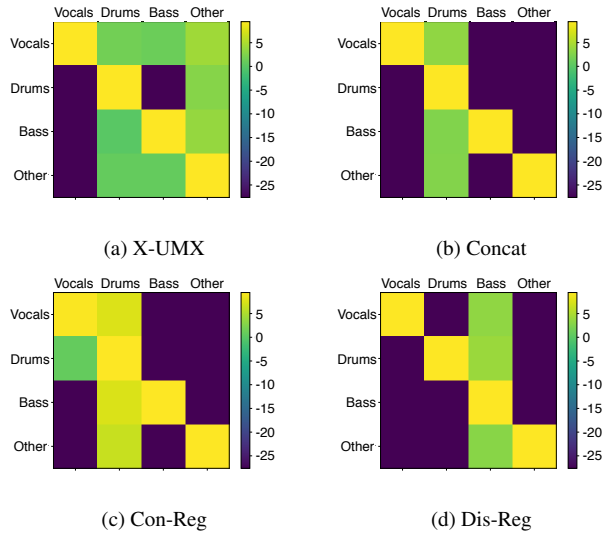


Figure 4: SDR, SIR and SAR scores for X-UMX and our proposed methods across four instruments.

By using the proposed methods, SIR scores improve slightly for 'Vocals', 'Drums,' and 'Bass.' This result suggests that the discriminative power of the VGGish features can benefit separation and decrease the interference of non-targeted sources. As our proposed methods aim to make the latent space more discriminative by either adding additional VGGish features or forcing different instruments' latent vector to be apart from each other, the interference from other instruments decreases. The soft margin employed in *Dis-Reg* generally leads to a higher SIR score than the hard margin in *Con-Reg*. The SIR score of 'Other,' however, decreases after regularization. We speculate that this is due to the non-homogeneity of the 'Other' category as it includes a variety of instruments with large distance variations in the embedding space. For example, if 'Other' contains low pitched instruments such as Tuba, then the distance might be closer to 'Bass' instead of 'Violin,' another instrument in 'Other.' As a result, the distance regularization might actually impede the training and thus lead to the observed poor SIR score. This result also suggests the possibility of improving source separation systems by replacing the 'Other' category with a set of specific "clean" instruments.



(a) X-UMX (b) Concat

(c) Con-Reg (d) Dis-Reg

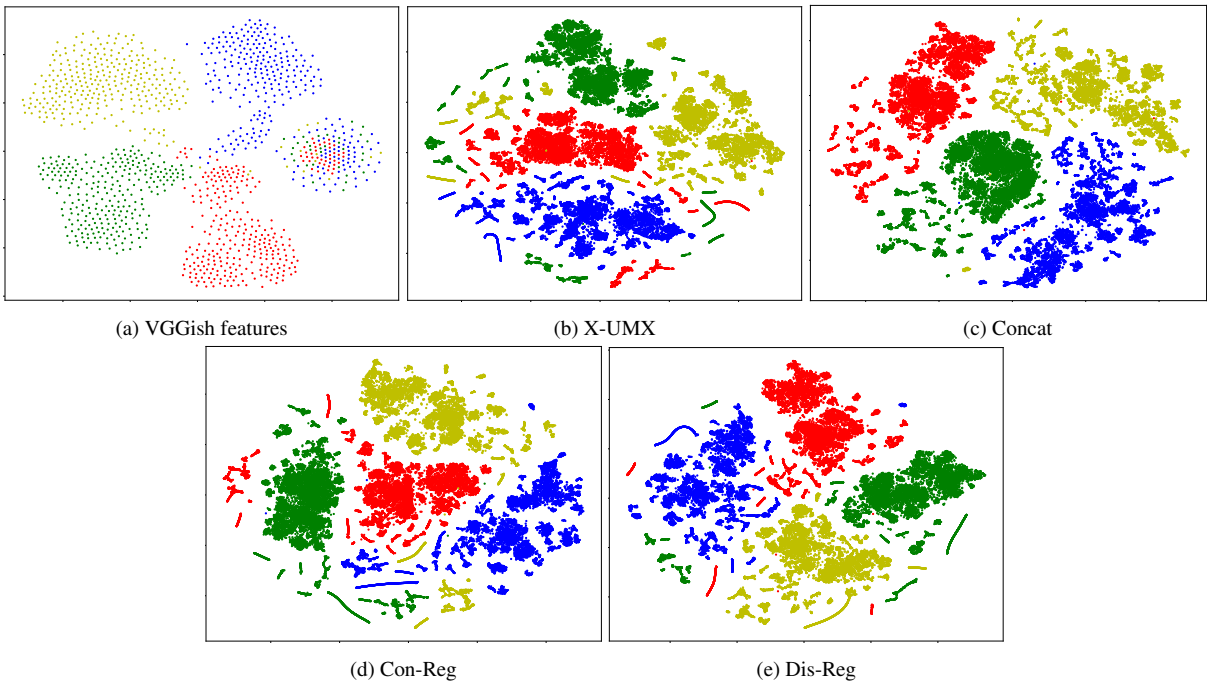
Figure 5: Confusion matrix of latent vector classification.

Comparing *con-Reg* and *Dis-Reg*, we can observe that *Dis-Reg* generally outperforms *Con-Reg*. The distance-based regularization appears to help the model maintain a structure targeted at source separation while adding relevant information. The SDR score stays mostly constant for most of the instruments except for 'Drums,' where we can observe improvement when using regularization methods. The general lack of improvement in SDR scores could potentially be related to much lower time resolution of VGGish features.

To summarize, while the *Concat* method performs better than *Dis-Reg* and *Con-Reg* on most of the metrics, it needs to incorporate the VGGish model during inference, which increases computation time and required memory. In contrast, *Dis-Reg* and *Con-Reg*, although they do not achieve the same improvement as *Concat* on some of the metrics, still show improvement over X-UMX for SIR and SAR and do not require the additional features during inference.

4.2. Latent Space Classification

To measure the discriminative power of the latent vectors after the proposed latent space modification, we compute the latent vectors from the test set for our four models: X-UMX, *Concat*, *Con-Reg*, and *Dis-Reg*. Each audio input has a $T_l \times C$ latent features for C represents the dimension of latent vectors after modification. Since all the tracks have same length, the total number of frames across classes is equally distributed. For *Con-Reg* and *Dis-Reg*, C is equal to 128 . C is equal to (B+128) for *Concat* and C is equal to B for X-UMX. We perform K-means algorithm on all the latent vectors across all audio inputs for clustering. The latent vectors from the



(a) VGGish features (b) X-UMX (c) Concat

(d) Con-Reg (e) Dis-Reg

Figure 6: The t-SNE visualization of the VGGish features as well as the latent vectors of the baseline X-UMX model and the proposed methods from an audio sample in test set. 'Blue': vocals, 'Red': drums, 'Green': bass, 'Yellow': other.

same instruments should be in the same cluster.

The confusion matrix of the proposed models are shown in Figure 5. The matrix is processed through log function for better visualization. 'Vocals' has the best performance for X-UMX latent vectors, while 'Other' tends to be confused with other instruments. The confusion is eased by using latent vectors from proposed methods. Surprisingly, when using *Concat* and *Con-Reg*, 'Drums' tends to get confused with other instruments and 'Bass' tends to to get confused when using *Dis-Reg*. The consistent confusion of a specific instrument might be caused by the silence or low volume frames. K-means algorithm directly assigns the closest instrument to those frames.

4.3. Latent Space Visualization

We randomly choose a sample from the test set and visualize a t-SNE projection [50] of the extracted latent vectors from the VGGish features, the baseline X-UMX model, *Concat*, *Dis-Reg*, and *Con-Reg*. The results are shown in Figure 6. We can observe from Figure 6b that — although the latent vectors are clearly clustered — several small clusters can be found to be far from their corresponding main cluster (top-left yellow and bottom-right red). This implies that model might confuse one instrument with another in these cases. When using regularization in Figure 6d and Figure 6e, the clusters tend to be more tightly packed and the spurious small clusters decrease in both number and size. Directly concatenating the VGGish features with latent vectors yields the most separable latent space, as shown in Figure 6c. As with classification, we observe that some of the clusters separated from the main clusters are from low volume audio frames or silence. Since they contain minimal instrument information, their latent vectors are difficult to cluster into one of the four groups. Moreover, we can observe that *Concat* and *Dis-Reg* achieve slightly better clustering results than X-UMX and *Con-Reg*. The result is aligned with our separation result in Figure 4, where *Dis-Reg* and *Concat* tend to outperform *Con-Reg*.

5. CONCLUSION

In this work, we propose three methods to incorporate VGGish features into a SOTA music source separation system. The first method simply concatenates features with the latent vectors, while the other two methods regularize the latent space through an additional loss function during training. Our proposed methods show the potential of reducing the artifacts and the interference created by the model and improving the SAR and SIR scores on 'Vocals,' 'Drums,' and 'Bass'. Latent space visualization confirms that the latent space has slightly better discriminative properties after regularization.

However, further experimentation is required to verify our approaches in different settings. For example, we plan to include other features into our proposed training strategies, such as the L3-Net embedding features [25]. Other model architectures, feature resolutions, and instrument categories, as well as the application to other audio-related tasks will also be studied in the future.

6. REFERENCES

[1] Len Vande Veire and Tijl De Bie, "From raw audio to a seamless mix: creating an automated DJ system for drum and bass," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2018, no. 1, pp. 1–21, 2018.

[2] Tülay Adalı, Christian Jutten, Arie Yeredor, Andrzej Cichocki, and Eric Moreau, "Source separation and applications," *IEEE Signal Processing Magazine*, 2014.

[3] Hideyuki Tachibana, Yu Mizuno, Nobutaka Ono, and Shigeki Sagayama, "A real-time audio-to-audio karaoke generation system for monaural recordings based on singing voice suppression and key conversion techniques," *Journal of Information Processing*, vol. 24, no. 3, pp. 470–482, 2016.

[4] Adam Berenzweig, Daniel P.W. Ellis, and Steve Lawrence, "Using voice segments to improve artist classification of music," in *Proceedings of AES International Conference on Virtual, Synthetic, and Entertainment Audio*, 2002.

[5] Daniel Stoller, Sebastian Ewert, and Simon Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," in *Proceedings of International Society for Music Information Retrieval Conference*, 2018, pp. 334–340.

[6] Woosung Choi, Minseok Kim, Jaehwa Chung, Daewon Lee, and Soonyoung Jung, "Investigating U-Nets with various intermediate blocks for spectrogram-based singing voice separation," in *Proceedings of International Society for Music Information Retrieval Conference*, 2020, pp. 192–198.

[7] Naoya Takahashi, Nabarun Goswami, and Yuki Mitsufuji, "MMDenseLSTM: An efficient combination of convolutional and recurrent neural networks for audio source separation," in *Proceedings of International Workshop on Acoustic Signal Enhancement*, 2018, pp. 106–110.

[8] Fabian-Robert Stöter, Stefan Uhlich, Antoine Liutkus, and Yuki Mitsufuji, "Open-Unmix a reference implementation for music source separation," *Journal of Open Source Software*, 2019.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*, 2015, pp. 234–241.

[10] Andreas Jansson, Eric Humphrey, Nicola Montecchio, Rachel Bittner, Aparna Kumar, and Tillman Weyde, "Singing Voice Separation with Deep U-Net Convolutional Networks.," in *Proceedings of International Society for Music Information Retrieval Conference*, 2017, pp. 745–751.

[11] Sungheon Park, Taehoon Kim, Kyogu Lee, and Nojun Kwak, "Music source separation using stacked hourglass networks," in *Proceedings of International Society for Music Information Retrieval Conference*, 2018, pp. 289–296.

[12] Weitao Yuan, Shengbei Wang, Xiangrui Li, Masashi Unoki, and Wenwu Wang, "A skip attention mechanism for monaural singing voice separation," *IEEE Signal Processing Letters*, vol. 26, no. 10, pp. 1481–1485, 2019.

[13] Alexandre Défossez, Nicolas Usunier, Léon Bottou, and Francis Bach, "Music source separation in the waveform domain," *arXiv preprint arXiv:1911.13254*, 2019.

[14] Ryosuke Sawata, Stefan Uhlich, Shusuke Takahashi, and Yuki Mitsufuji, "All for one and one for all: Improving music separation by bridging networks," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

[15] Olga Slizovskaia, Leo Kim, Gloria Haro, and Emilia Gomez, "End-to-end sound source separation conditioned on instrument labels," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 306–310.

[16] Yun-Ning Hung and Alexander Lerch, "Multitask learning for instrument activation aware music source separation," in *Proceedings of International Society for Music Information Retrieval Conference*, 2020, pp. 748–755.

[17] Ethan Manilow, Prem Seetharaman, and Bryan Pardo, "Simultaneous separation and transcription of mixtures with multiple polyphonic and percussive instruments," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 771–775.

[18] Andreas Jansson, Rachel M Bittner, Sebastian Ewert, and Tillman Weyde, "Joint singing voice separation and f0 estimation with deep U-net architectures," in *Proceedings of European Signal Processing Conference*, 2019, pp. 1–5.

[19] Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner, "The MUSDB18 corpus for music separation," 2017.

[20] Shawn Hershey, Sourish Chaudhuri, Daniel P.W. Ellis, Jort Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson, "CNN architectures for large-scale audio classification," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics*, 2019, pp. 4171–4186.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[23] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[24] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[25] Jason Cramer, Ho-Hsiang Wu, Justin Salamon, and Juan Pablo Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 3852–3856.

[26] Yusuf Aytar, Carl Vondrick, and Antonio Torralba, "Soundnet: Learning sound representations from unlabeled video," in *Proceedings of International Conference on Neural Information Processing Systems*, 2016, p. 892–900.

[27] Siddharth Gururani, Mohit Sharma, and Alexander Lerch, "An Attention Mechanism for Musical Instrument Recognition," in *Proceedings of International Society for Music Information Retrieval Conference*, 2019, pp. 83–90.

[28] Yi Yu, Suhua Tang, Francisco Raposo, and Lei Chen, "Deep cross-modal correlation learning for audio and lyrics in music retrieval," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 15, no. 1, pp. 1–16, 2019.

[29] Donghuo Zeng, Yi Yu, and Keizo Oyama, "Audio-visual embedding for cross-modal music video retrieval through supervised deep cca," in *IEEE International Symposium on Multimedia*, 2018, pp. 143–150.

[30] Khaled Koutini, Hamid Eghbal-zadeh, and Gerhard Widmer, "Receptive field regularization techniques for audio classification and tagging with deep convolutional neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1987–2000, 2021.

[31] Eunjeong Koh and Shlomo Dubnov, "Comparison and analysis of deep audio embeddings for music emotion recognition," *arXiv preprint arXiv:2104.06517*, 2021.

[32] Jaime Ramírez and Julia Flores, "Machine learning for music genre: multifaceted review and experimentation with audioset," *Journal of Intelligent Information Systems*, vol. 55, no. 3, pp. 469–499, 2020.

[33] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho, "Transfer learning for music classification and regression tasks," in *Proceedings of International Society for Music Information Retrieval Conference*, 2017, pp. 141–149.

[34] Jort Gemmeke, Daniel P.W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 776–780.

[35] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan, "Youtube-8M: A large-scale video classification benchmark," *arXiv preprint arXiv:1609.08675*, 2016.

[36] Olga Slizovskaia, Gloria Haro, and Emilia Gómez, "Conditioned source separation for music instrument performances," *arXiv preprint arXiv:2004.03873*, 2020.

[37] Rupak Vignesh Swaminathan and Alexander Lerch, "Improving singing voice separation using attribute-aware deep network," in *Proceedings of International Workshop on Multilayer Music Representation and Processing*, Milan, Italy, 2019.

[38] Prem Seetharaman, Gordon Wichern, Shrikant Venkataramani, and Jonathan Le Roux, "Class-conditional embeddings for music source separation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 301–305.

[39] Venkatesh Kadandale, Juan Montesinos, Gloria Haro, and Emilia Gómez, "Multi-channel u-net for music source separation," in *IEEE International Workshop on Multimedia Signal Processing*, 2020, pp. 1–6.

[40] Sebastian Ewert, Bryan Pardo, Meinard Muller, and Mark Plumbley, "Score-informed source separation for musical audio recordings: An overview," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 116–124, 2014.

[41] Marius Miron, Julio Carabias-Orti, Juan Bosch, Emilia Gómez, and Jordi Janer, "Score-informed source separation for multichannel orchestral recordings," *Journal of Electrical and Computer Engineering*, vol. 2016, 2016.

[42] Marius Miron, Jordi Janer, and Emilia Gómez, "Monaural Score-Informed Source Separation for Classical Music Using Convolutional Neural Networks.," in *Proceedings of International Society for Music Information Retrieval Conference*, 2017, pp. 55–62.

[43] Julio Carabias-Orti, Máximo Cobos, Pedro Vera-Candeas, and Francisco Rodríguez-Serrano, "Nonnegative signal factorization with learnt instrument models for sound source separation in close-microphone recordings," *EURASIP Journal on Advances in Signal Processing*, vol. 2013, no. 1, pp. 1–16, 2013.

[44] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba, "The sound of pixels," in *Proceedings of European Conference on Computer Vision*, 2018, pp. 570–586.

[45] Li-Chia Yang and Alexander Lerch, "Remixing music with visual conditioning," in *Proceedings of IEEE International Symposium on Multimedia*, 2020, pp. 181–188.

[46] Chuang Gan, Deng Huang, Hang Zhao, Joshua B Tenenbaum, and Antonio Torralba, "Music gesture for visual sound separation," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10478–10487.

[47] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[48] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *Proceedings of International Conference for Learning Representations*, 2015.

[49] Fabian-Robert Stöter, Antoine Liutkus, and Nobutaka Ito, "The 2018 signal separation evaluation campaign," in *Proceedings of International Conference on Latent Variable Analysis and Signal Separation*, 2018, pp. 293–305.

[50] Laurens Van der Maaten and Geoffrey Hinton, "Visualizing data using t-SNE.," *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.

[51] Jordi Pons, Oriol Nieto, Matthew Prockup, Erik Schmidt, Andreas Ehmann, and Xavier Serra, "End-to-end learning for music audio tagging at scale," in *Proceedings of International Society for Music Information Retrieval Conference*, 2018, pp. 637–644.