Reghunath and Rajan
EURASIP Journal on Audio, Speech, and Music Processing (2023) 2023:49
https://doi.org/10.1186/s13636-023-00316-4

EURASIP Journal on Audio, Speech, and Music Processing

EMPIRICAL RESEARCH

Open Access

Predominant audio source separation in polyphonic music

Lekshmi Chandrika Reghunath[1*] and Rajeev Rajan[2]

Abstract

Predominant source separation is the separation of one or more desired predominant signals, such as voice or leading instruments, from polyphonic music. The proposed work uses time-frequency filtering on predominant source separation and conditional adversarial networks to improve the perceived quality of isolated sounds. The pitch tracks corresponding to the prominent sound sources of the polyphonic music are estimated using a predominant pitch extraction algorithm and a binary mask corresponding to each pitch track and its harmonics are generated. Time-frequency filtering is performed on the spectrogram of the input signal using a binary mask that isolates the dominant sources based on pitch. The perceptual quality of source-separated music signal is enhanced using a CycleGAN-based conditional adversarial network operating on spectrogram images. The proposed work is systematically evaluated using the IRMAS and ADC 2004 datasets. Subjective and objective evaluations have been carried out. The reconstructed spectrogram is converted back to music signals by applying the inverse short-time Fourier transform. The intelligibility of separated audio is enhanced using an intelligibility enhancement module based on an audio style transfer scheme. The performance of the proposed method is compared with state-of-the-art Demucs and Wave-U-Net architectures and shows competing performance both objectively and subjectively.

Keywords Predominant, Spectrogram, Time-frequency filtering, Generative adversarial network, Binary masking

1 Introduction

In general, the acoustical environment does not offer sound sources alone. The so-acquired auditory sensory information consists of many sound sources that are likely to overlap in both time and frequency. The basis for the analysis of the acoustic scene is the capacity of human perception to resolve this acoustical mixture. Our human auditory system is capable of distinguishing the constituent sounds in a speech or music mixture, even if they overlap with time and frequency. Since music represents, in general, a multi-source acoustical environment, the here-described data properties indeed represent the main complexity involved in this method. Cherry [1] coined the problem as the cocktail party problem by exemplifying a conversational situation where several voices, overlapping in time, are embedded in a natural acoustical environment including other stationary or dynamic sound sources. The listener, however, can focus on the targeted speech stream and transform the acoustical data into semantic information. But this is a difficult task for computers to automatically do.

Music is viewed as a different issue than other types of source separation. This is because many factors make music uniquely difficult. Music sources are highly correlated and it is mixed and processed non-physically and non-linearly. Reverberation, filtering, and other non-linear signal processing techniques make music separation difficult. This is a problem because you rarely know what processing has been applied to the source or the

*Correspondence:
Lekshmi Chandrika Reghunath
clekshmir04@gmail.com
[1] Department of Electronics and Communication Engineering, College of Engineering Trivandrum, APJ Abdul Kalam Technological University, Trivandrum, India
[2] Department of Electronics and Communication Engineering, Government Engineering College Barton Hill, APJ Abdul Kalam Technological University, Trivandrum, India

mix in an end-to-end system. Since sound source separation is the end goal, listened to by users who can expect high-quality results, it is of utmost importance that the system's results sound good enough for these users. The goal of predominant source separation is to separate prominent instruments including the human voice from a song, without adding any noise. The separation of lead or accompaniment from the polyphonic song, that is the real-world scenario is a relatively difficult task [2].

The ability to interact with the individual sound sources especially lead vocals in a music recording would enable diverse applications such as music up mixing and remixing, automatic karaoke, and object-wise equalization [3]. Also, we can make music recommender systems to listen to the predominant instrument or vocals alone. All these are possible by predominant source separation models.

1.1 Related works

Blind source separation techniques like independent component analysis (ICA) [4], non-negative matrix factorization (NMF) [5], and spectral filtering [6] are some of the initial attempts in sound source separation. NMF is a common method by which the spectrogram of a mixed signal is factored. The problem with this approach, instead of multiplying two non-negative fundamental matrices and a weight matrix, is that we might get a non-negative linear combination of the trained basis vectors as the source signal. Thus, it may degrade the performance of the segregation approach [5]. Spectral filtering techniques such as time-frequency masking are an easy way to separate speech sources, especially when they are sparse and non-overlapping [6]. These methods cannot be directly applied for music source separation because of the peculiar properties of music, and this required the development of specific music algorithms that have prior knowledge about source structure and mixing parameters [7]. Moreover, a typical commercial music mix violates all the classical assumptions of ICA. Li and Wang [8] proposed a comp filtering approach to use a vocal/non-vocal classifier with a predominant pitch detection algorithm for detecting pitch contours from vocals. Ryynanen et al. [9] proposed a method to separate accompaniment from polyphonic music using melody transcription and sinusoidal modeling is employed to estimate, resynthesize, and separate the lead vocals from the musical mixture for karaoke applications. In [10], predominant melodic source is tracked by borrowing concepts from graph theory and computational auditory scene analysis.

Recently, deep neural networks have been applied to speech enhancement, segmentation, and source separation by estimating complex masks corresponding to each source. Huang et al. [11] and Uhlich et al. [12] were the first to propose deep neural networks like recurrent neural networks (RNN) and long short-term memory (LSTM) for the singing voice separation task. A deep neural network-based framework to classify the spectra of the mixed signal into each possible source type is proposed in [13]. The main limitation of the methods using time-frequency masking is that most of them synthesize using masked spectrograms with the original phase [14] of the mixture, which imposes an upper bound on the performance. Luo et al. proposed time-domain audio separation network (TasNet) overcome these limitations [14]. Mimilakis et al. proposed a hybrid structure with skip connections and recurrent inference of time-frequency mask to separate the lead instrument from jazz remix recordings [15].

Generative adversarial network (GAN)-based source separation is proposed in [16], where a Wasserstein GAN generative model is trained to recover sources from mixtures. Luo and Mesgarani [17] propose a generative network that directly models signals in the time domain using a 1-D convolutional encoder/decoder framework and perform source separation on the non-negative encoder outputs. A 1-D convolutional network based on successive downsampling and resampling is proposed in [18] for universal speech separation. SepNet, a DNN, was developed to predict the separation matrix for [19] speech separation. In [20], an adaptation of the U-Net architecture to the one-dimensional time domain to perform end-to-end audio source separation is performed. Recently, adversarial networks have often been used for speech synthesis [21]. In the proposed method Cycle-GAN, an adversarial network is used for separating the predominant source from the polyphonic mix. The proposed method employs a different approach using time-frequency masking, and perceptibility is enhanced by using generative adversarial networks.

1.2 Motivation

Existing techniques for source separation are signal processing models based on prior knowledge about the instrument class, and they faced difficulties when trying to differentiate instruments with similar frequency content and failed to provide a clear instrument identification. Many advances have been made in source separation especially in the last decade, after deep learning techniques started being used. Generative adversarial networks (GANs) have recently been shown to be efficient for speech enhancement [22]. Also, it has been already demonstrated that GANs can effectively be used for suppressing additive noise and improving perceptual quality metrics [23, 24]. GAN explores different ways of learning useful encodings of the music signals that would facilitate the separation without any prior knowledge about music data. It motivated us to employ a GAN-based scheme for the proposed work.

1.3 Problem statement

Consider a stereo polyphonic signal $x \in \mathbb{R}^{2\times N}$, where N denotes the number of samples, and let $N \in \mathbb{S}^*$. This signal consists of a combination of source signals $s_j \in \mathbb{R}^{2\times N}$, where j belongs to the set $\{1, 2, \ldots, J\}$, with $J \in \mathbb{S}^*$ being the number of sources. Mathematically, this can be expressed as:

$$x = \sum_{j=1}^{J} s_j$$

Let us denote the predominant signal as s_i, where i takes values from $\{1, 2, \ldots, I\}$, and I signifies the count of predominant signals. The primary objective of the task of predominant audio source separation is to discern the source signals s_i based exclusively on the observed mixture signal x.

The proposed method is discussed in Section 2. Section 3 presents the performance evaluation followed by results and analysis in Section 4. Finally, the paper is concluded in Section 5.

2 Proposed method

In the proposed method, binary masks are generated from the estimates of the pitch tracks. This binary mask is then used to compute the masked predominant spectrogram from the mixed spectrogram. To generate the enhanced spectrum, an image-to-image transformation is applied to the masked dominant spectrograms using CycleGAN. Finally, the spectrum and mixed-signal phase are used to synthesize the dominant source of the constituents. The overall process is shown in Fig. 1.

2.1 Predominant pitch tracking and binary masking

The predominant pitch tracking algorithm is a salience-based melody extraction method originally designed to extract the predominant melody from polyphonic music, as well as monophonic signals [25]. For the current experiment, the predominant pitch is extracted from the audio recordings using MELODIA. This algorithm outputs a time series (sequence of values) with the perceived melody's instantaneous pitch value (in hertz). The approach is based on creating and characterizing pitch contours and time-continuous sequences of pitch candidates grouped using auditory streaming cues [25]. By studying the feature distributions of melodic and non-melodic contours, they defined a set of rules for distinguishing between the contours that form the melody and the contours that should be filtered out. These rules are used to mitigate the challenges related to voice activity detection, the possibility of octave errors, and the presence of strong secondary melodic components [25].

Binary masks are generated based on pitch tracks obtained from predominant pitch tracking algorithms and are then applied to the spectrogram of the polyphonic signal to obtain the dominant masked spectrogram. Spectrograms with enhanced pitch track information are plotted with the *librosa.piptrack* function. The function *librosa.piptrack*(*pitches*, *magnitudes*) returns two 2D arrays with frequency and time axes. The "pitches" array gives the interpolated frequency estimate of a particular harmonic, and the corresponding value in the "magnitudes" array gives the energy of the peak. Figure 2 represents the polyphonic spectrogram of music signal from the dataset with pitch tracks highlighted and corresponding

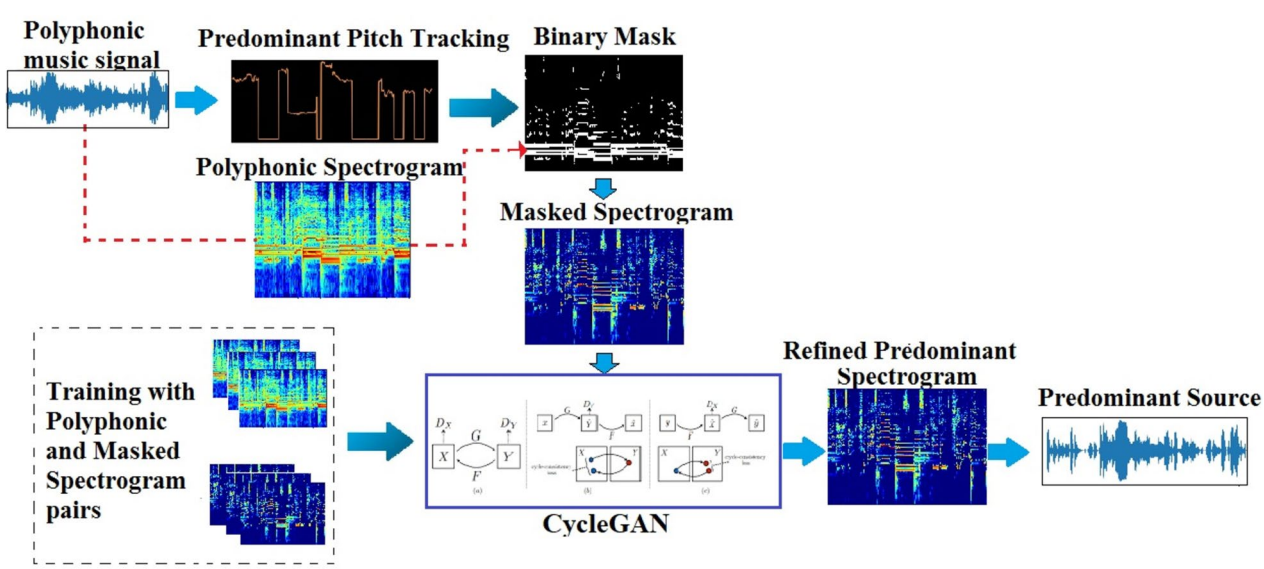

Fig. 1 Schematic diagram of the flow for the proposed CycleGAN-based model

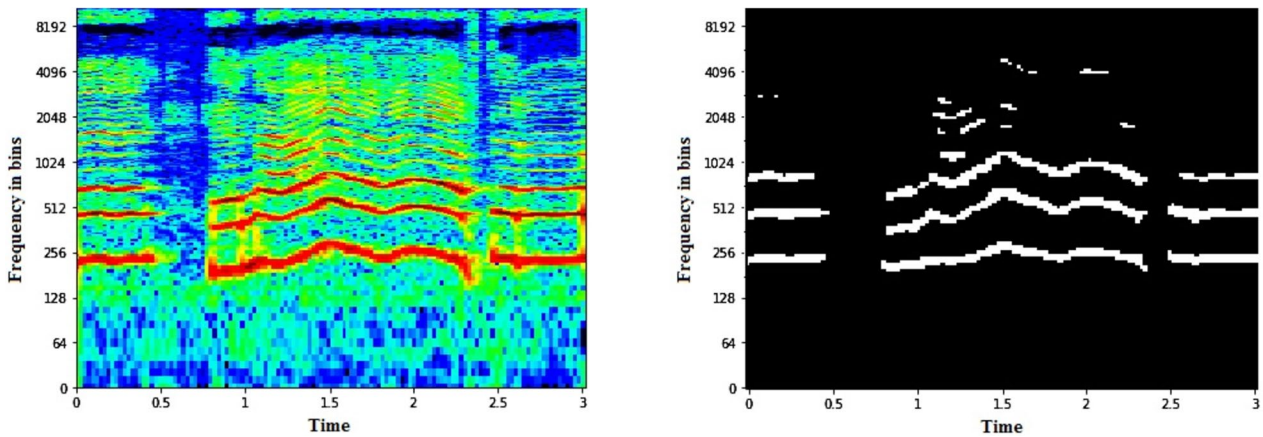

Fig. 2 Spectrogram of polyphonic music signal and corresponding binary mask

binary mask. Phase information is preserved for reconstruction at the end.

The time-frequency masking approach takes advantage of the sparsity of mixed speech signals to perform source separation. The time-frequency representations of the source will no longer overlap, improving overall performance. The sparseness of the signal encourages the concept of using clustering techniques, including mean-shift algorithms. These methods assume that the audio sources are separated in the time-frequency domain. That is, at any given time and any given frequency, only one source is transmitted and the others are zero, a condition known as W-disjoint Orthogonality [6]. Our approach does not claim such constraints and relies on a generative adversarial network to improve the perceptual quality of separated music signals after time-frequency masking.

2.2 Cycle generative adversarial network (CycleGAN)

GANs use pretraining [26] to synthesize realistic examples from the dataset. These are generative models, learning to map samples i from a previous distribution $\mathcal{I}$ to samples j from another distribution $\mathcal{J}$. This is one of the training examples. The generator (G) and discriminator (D) are two components of GAN. Generator G performs mapping using adversarial training. Discriminator D classifies input samples as real or fake. Generator G attempts to provide samples from the distribution of interest, and the discriminator attempts to predict whether the samples came from the actual distribution or were produced by the generator. D tries to find realistic features of the input, and G transforms the samples to be more realistic by changing parameters during backpropagation. The generator and discriminator are trained simultaneously. With this, the generator eventually learns to perfectly approximate the underlying distribution, and the discriminator has to guess randomly. However, given enough capacity, the network can map the same set of input images to any random permutation of images in the target domain. In this case, each learned mapping can induce an output distribution that matches the target distribution. For this reason, opponent defeat alone cannot guarantee that the learned function can map each input image to the desired output image. To regularize the model, the authors introduce a cycle consistency constraint. If you transform from the source distribution to the target distribution and back to the source distribution, you need to take samples from the source distribution [27].

A conditional model can be modeled from GANs, if both aforesaid components are conditioned on additional information such as label/ data from other modality y. Pairing is achieved to make input and output share some common features. This is done by inputting y to the components as additional input layers. The prior input noise $p_z(z)$ and y are combined in a joint hidden representation in the generator. In the discriminator, x and y are presented as inputs to a discriminative function [28]. However, to ensure that there is a meaningful relation between these images, they must share some feature, features that may be used to map this output image back to the input image, so there has to be another generator that must be capable of mapping back this output image again to the original input. The job of the discriminator is to distinguish between the original image and the generated image, while the generator would like to ensure that those images get accepted by the discriminator, so it will try to generate images that are very close to the original images. In reality, the generator and discriminator are playing a game whose Nash equilibrium is achieved when the generator's distribution becomes the same as the desired distribution.

Let the CycleGAN use the training patterns $\{a_i\}_1^N \in A$ and $\{b_i\}_1^N \in B$ to convert observation image A and random noise vector N to B . Consider two discriminators D_A and D_B. During training, D_A learns the mapping function $G : A \rightarrow B$ so that the image produced by $G(A)$ is indistinguishable from B. D_A is intended to distinguish between the image a and the translated image $F(b)$. The main purpose of D_B is to distinguish between images b and $G(a)$. For that it learns the inverse function $F : B \rightarrow A$ such that $F(G(A)) = A$. To enforce the conditions $F(G(A)) = A$ and $G(F(B)) = B$, cycle consistency loss is enabled in training.

The objective is [27]:

$$l(G, F, D_A, D_B) = l_{GAN}(G, D_B, A, B) + l_{GAN}(F, D_A, B, A) + \lambda l_{cyc}(G, F), \tag{1}$$

$$l_{GAN}(G, D_B, A, B) = E_{b\sim pdata}(b)\left[logD_{B(b)}\right] + \underset{a\sim pdata(a)}{\mathbb{E}}[log(1 - D_B(G(a))] \tag{2}$$

where,

$$l_{cyc}(G, F) = \underset{a\sim pdata(a)}{\mathbb{E}}[\| F(G(a)) - a \|_1] + \underset{b\sim pdata(b)}{\mathbb{E}}[\| G(F(b)) - b \|_1]. \tag{3}$$

$l_{GAN}(G, D_B, A, B)$ and $l_{cyc}(G, F)$ indicate adversarial loss and consistency loss [27]. G produces an image $G(a)$ that mimics the image in domain B, while D_B distinguishes between translated image $G(a)$ and b . The training process is shown in Fig. 3. In the proposed experiment, a represents the masked predominant spectrogram b denoting the corresponding ground-truth spectrogram in the training phase.

2.3 Cycle GAN architecture

The proposed model comprises 4 components, two for generators and discriminators. The model is designed to process images of size 256 × 256. Discriminators are labeled as Domain-A (original) and Domain-B (predominant). Discriminator resembles PatchGAN where each output prediction of the model maps to a 70 × 70 patch of the input image. Convolutional-InstanceNorm-LeakyReLU layers are used in the discriminator for the processing. InstanceNormalization is used which involves standardizing the values on each output feature map, rather than across features in a batch as in BatchNormalization. The input to the model is 256 × 256 images and outputs a patch of predictions. Least-squares loss (L2)-based optimization is adopted with a weighting parameter of 0.5. Figure 4 represents the internal architecture of cycleGAN with nine resnet blocks. The generator which is based on an encoder-decoder-schema outputs pixel values with the shape as the input and pixel values are in the range [− 1, 1]. The generator first downsamples the input image to a bottleneck layer, then interprets the encoding with several Resnet layers that use skip connections. Later, the process is reversed to the size of the output image. 3 × 3 filters and 1 × 1 stride have been used in the CNN of the Resnet blocks. It is worth noting that the input to the block is concatenated channel-wise to the output of the block. The generator is updated as a weighted average of the 4-loss values, specifically adversarial loss, identity loss, forward cycle loss, and backward cycle loss. The weighting parameter *lambda* has been selected as 10 for the forward and backward cycle loss of adversarial loss, and a fraction of

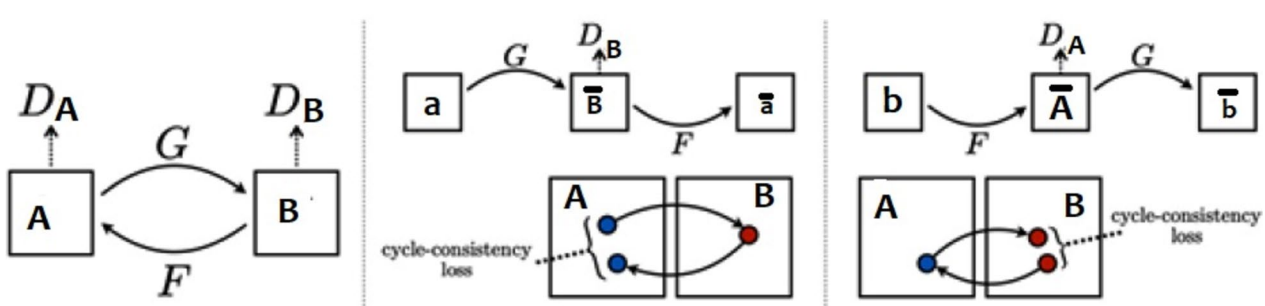

Fig. 3 CycleGAN training with cycle-consistency loss function [27]

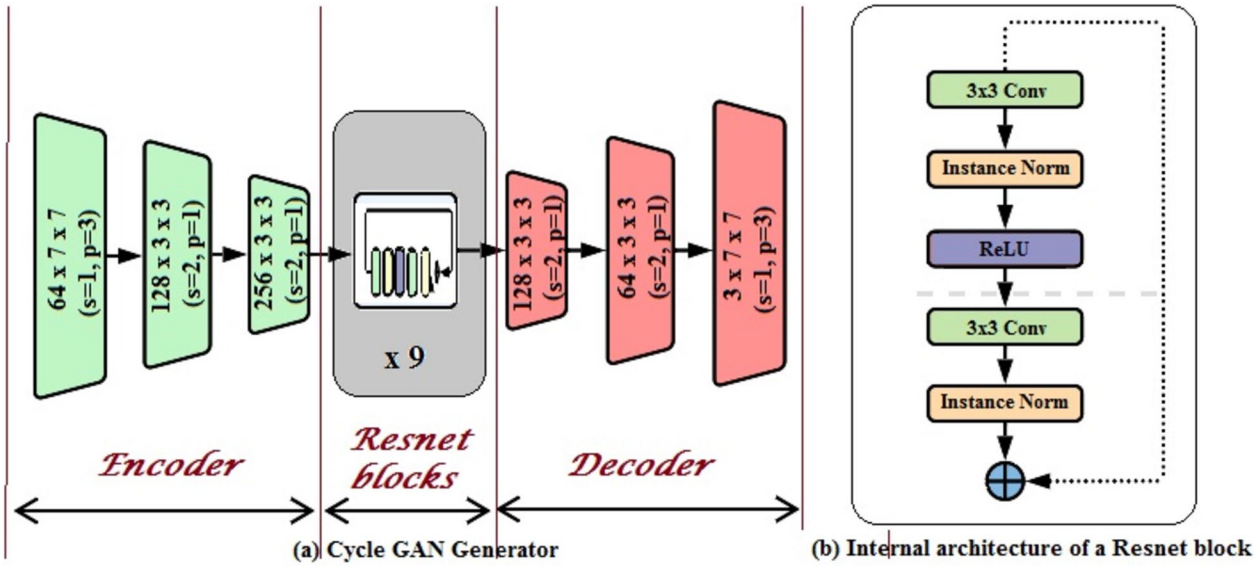

Fig. 4 CycleGAN Generator architecture

Table 1 Proposed architecture for discriminator

Input size	Description
3× 256 × 256	Input spectrogram
64 × 128 × 128	4 × 4 Conv, 64 filters, stride 2, pad 1
64 × 128 × 128	Leaky ReLU (α=0.2)
128 × 64 × 64	4 × 4 Conv, 64 filters, stride 2, pad 1
128 × 64 × 64	Instance normalization
128 × 64 × 64	Leaky ReLU (α= 0.2)
256 × 32 × 32	4 × 4 Conv, 64 filters, stride 2, pad 1
256 × 32 × 32	Instance normalization
256 × x32 × 32	Leaky ReLU (α= 0.2)
512 × 31 × 31	4 × 4 Conv, 512 filters, stride 1, pad 1
1 × 4 × 4	4 × 4 Conv, stride 1, pad 1

Table 2 Proposed architecture for generator

Input size	Description
3 × 256 × 256	Input spectrogram
64 × 256 × 256	7 × 7 Conv, 64 filters, stride 1, pad 3
64 × 256 × 256	Instance normalization
64 × 256 × 256	ReLU
128 × 128 × 128	3 × 3 Conv, 128 filters, stride 2, pad 1
128 × 128 × 128	Instance normalization
128 × 128 × 128	ReLU
256 × 64 × 64	3 × 3 Conv, 256 filters, stride 2, pad 1
256 × 64 × 64	Instance normalization
256 × 64 × 64	ReLU
256 × 64 × 64	9 consecutive Resnet blocks, 256 filters
128 × 128 × 128	3 × 3 Conv, 128 filters, stride 2, pad 1
128 × 128 × 128	Instance normalization
128 × 128 × 128	ReLU
64 × 256 × 256	3 × 3 Conv, 64 filters, stride 1, pad 3
64 × 256 × 256	Instance normalization
64 × 256 × 256	ReLU
3 × 256 × 256	7 × 7 Conv, stride 1, pad 3
3 × 256 × 256	Instance normalization
3 × 256 × 256	Tanh

lambda (0.5) has been selected as a weighting factor for identity loss.

All generators and discriminators are optimized with Adam optimizer with learning rate (2e−4), and the batch size is chosen to be 1. During training for the first 100 epochs, the learning rate is fixed to 2e−4, and for the last 100 epochs, the learning rate is linearly annealed from 2e−4 to 2e−6. Tables 1 and 2 represent the model summary of the discriminator and generator used in the proposed method.

2.4 Intelligent enhancement module

An intelligent enhancement module utilizing style transfer [29, 30] is employed that merges the principles of style transfer with artificial intelligence to elevate the aesthetic and quality of separated audio. The core of style transfer involves transposing the distinct visual or auditory style of one piece of content onto another, all the while preserving the fundamental structure of the content. The module aimed at enhancing intelligibility utilizes a single-layer CNN equipped with 4096 filters, followed by ReLU activation for extracting high-level features. The

initial step involves converting the raw audio into a spectrogram through a short-time Fourier transform (STFT). Each input audio segment is divided into T frames using a Hanning window of n samples (2048) and a hop size of n/2, effectively representing the spectrogram as an image with T channels and n samples per channel. To extract content and style features, a random CNN is employed. The output of this random CNN is used to compute the Gram matrix [29], which describes the style of the audio. Subsequently, an iterative optimization process is applied to gradually minimize both content loss and style loss.

Content loss is determined by measuring the squared error between the content features derived from the content input and the features obtained from the random input. In contrast, style loss is calculated by quantifying the squared error between the Gram matrix computed from the style audio sample and the Gram matrix derived from the features of the random input. The overall loss is then computed as a linear combination of the content loss and the style loss. This total loss guides the optimization process, progressively aligning the output with the desired content and style characteristics.

3 Performance evaluation

3.1 Dataset

IRMAS and ADC 2004 datasets are used for performance evaluation. IRMAS dataset [31] containing separate training and testing sets is used for the evaluation. All audio files in the IRMAS dataset are in a 16-bit stereo .wav format with a sampling rate of 44,100 Hz. The training data are single-labeled and consist of 6705 audio files with excerpts of 3 s from more than 2000 distinct recordings. On the other hand, the testing data are multi-labeled and consist of 2874 audio files with lengths between 5 s and 20 s and contain multiple predominant instruments.

ADC 2004 dataset [32] containing 20 audio clips with a sampling rate of 44,100 Hz and include five genres, namely daisy, jazz, opera, MIDI, and pop, are used for evaluation. This dataset consists of 20 excerpts, with the average length of each song being 20 s.

3.2 Evaluation methodology

The subjective and objective evaluations were carried out. According to this evaluation procedure, a source signal estimation sig_{est} can be decomposed as follows:

$$sig_{est} = sig_{org} + err_{spat} + err_{inter} + err_{artif} \tag{4}$$

where sig_{org} is the orginal clean signal, err_{spat} is the error due to spatial distortions, err_{inter} is the error due to interference and err_{artif} is the error due to artifacts. The relative amounts of spatial distortion, interference, and artifacts are measured using the following three energy ratio criteria expressed in decibels (dB) namely: the source image to spatial distortion ratio (ISR), the source to distortion ratio (SDR), and the sources to artifacts ratio (SAR) [33].

$$SDR = 10log_{10}\frac{\|s\|^2}{\|err_{spat} + err_{inter} + err_{artif}\|^2} \tag{5}$$

$$ISR = 10log_{10}\frac{\|s\|^2}{\|err_{spat}\|^2} \tag{6}$$

$$SAR = 10log_{10}\frac{\|s + err_{spat} + err_{inter}\|^2}{\|err_{artif}\|^2} \tag{7}$$

These metrics were computed using functions in the "Nussl" library in Python upon all the testing data. The class-wise average and maximum values are taken for objective evaluation. SAR is interpreted as the quantity of other sources that can be heard in a source estimate. SDR reveals the overall quality of each estimated source and is interpreted as the amount of unwanted artifacts a source estimate has in relation to the true source [33, 34].

A perception test is conducted by sharing a subset comprising 26 audio samples with 20 listeners. All the listeners were presented with a polyphonic signal and the corresponding predominant source was separated for opinion grading. It is measured using five opinion grades, namely excellent (5), very good (4), good (3), poor (2), and very poor (1). Listeners are asked to grade by choosing any of the opinion grades. Direction has been given to listeners to grade each synthesized audio file by considering the effect of cross-talks and break sound effects. Sound quality relates to the number of artifacts or distortions that you can perceive. Interference describes the loudness of the predominant source compared to the loudness of the accompaniments. For example, "strong interference" indicates a strong contribution from accompaniments, whereas "no interference" means that only a predominant source is present in separation. Interference does not include artifacts or distortions that you may perceive. The total mean opinion score (MOS) is computed by taking the average of the scores obtained during the perception test.

3.3 Experimental framework

Spectrograms are computed with a frame size of 50 ms and a hop size of 10 ms. The predominant pitch track is computed using the predominant pitch tracking algorithm using Essentia, and a corresponding binary mask is generated. The spectrogram shown in Fig. 2b highlights the estimated predominant pitch track for a polyphonic music signal. Approximate predominant source spectra

are generated by applying the binary mask to the spectrogram of the mixed signal. A conditional adversarial network is used to enhance the masked predominant spectrogram and is implemented using an image-to-image translation model described in [35]. During the training phase, the original spectrogram of the polyphonic signal is given to the train A folder, while corresponding masked spectrograms are given to the train B folder. Generators learn to translate images from "train A" to "train B" and vice versa. Discriminators evaluate the authenticity of generated images in both domains. A subset of the IRMAS dataset with classes including flute (Flu), acoustic guitar (Gac), saxophone (Sax), trumpet (Tru), and human singing voice (Voice) is used for evaluation. Since it has separate training and testing data, 70% the training data of fixed-length (max. duration of wave file 3 s) of a class is used for training.

The training is performed in a Google Colab Pro GPU environment in 500 epochs with a batch size of 150. Validation of the hyper-parameters is performed using 20% of training spectrograms. The variable-length testing files with a single predominant instrument(max. duration of wave file 20 s) which is never used for training or validation are used for testing. A total of 105 files with acoustic guitar (25), flute (20), saxophone (10), trumpet (14), and voice (36) are used for evaluation. The masked spectrograms obtained from the front end are given to the trained CycleGAN network to enhance the spectrum. Finally, the constituent predominant source is reconstructed using the refined spectrum and mixed-signal phase. Later, objective and subjective evaluation is carried out.

3.4 Performance comparison

The performance of the proposed method is compared with state-of-the-art Demucs [36] and Wave-U-Net architecture [20] which separates leading voice and instrument in the time domain.

The Demucs v2 [36] introduces a encoder/decoder architecture composed of a convolutional encoder, a bidirectional long short-term memory (LSTM), and a convolutional decoder, with the encoder and decoder linked with skip U-Net connections. The U-Net structure allows for efficient feature extraction and integration across different scales of the input signal. The encoder extracts hierarchical representations of the input mixture, while the decoder generates separated sources based on these representations [36].

The Wave-U-Net is an adaptation of the U-Net architecture to the one-dimensional time domain to perform end-to-end audio source separation. Through a series of downsampling and upsampling blocks, which involve 1D convolutions combined with a down-/upsampling process, features are computed on multiple scales/levels of abstraction and time resolution and combined to make a prediction [20]. The pre-trained models M4 and M6 are used to separate the lead voice and instrument [20].

4 Results and analysis

The objective evaluation results showing the overall performance for the IRMAS and ADC2004 datasets are tabulated in Table 3. For ADC an average SAR of 4.24 is observed. The decreased performance of this is due to the fewer training samples available in the dataset. Figure 5 represents the metrics comparison of the proposed method with Wave-U-Net and Demucs architecture. It can be seen that an average SAR of 8.38 and 8.22 have been reported for Demucs and our proposed method respectively. Similar comparable results are obtained for SDR and ISR as shown in Table 4. The inclusion of an intelligibility enhancement module increased the SDR and SAR measures and is reported in Table 4. The idea of using a single model that can specialize in predicting the outputs directly from the inputs allows the development of otherwise highly complex systems that can be considered state-of-the-art. Even though an end-to-end model produces good results, it has some limitations that make it infeasible in some situations. An end-to-end model requires huge training data and is prone to temporal distortions, particularly in a polyphonic environment. Also, an end-to-end model is difficult to validate and often results in wrong outputs.

The results for CycleGAN-based source separation are shown in Fig. 6. The first row of the figure from Fig. 6(a) to (e) represents the CycleGAN-generated spectrograms at various stages of the proposed model from 100 to 500 epochs. The bottom row represents the ground truth spectrogram, and the corresponding CycleGAN generated one for comparison. From the generated spectrogram images, it is clear that near-real reconstruction is achieved in spectral aspects. Phase information from the original mixed audio is added vectorially, and inverse STFT is applied. Figure 7 represents the qualitative analysis of our proposed method with the Demucs model. Figure 7a represents the ground truth spectrogram of the predominant test signal and Fig. 7b represents the corresponding binary mask generated by the predominant extraction algorithm. Figure 7c represents the CycleGAN generated spectrogram, and Fig. 7d represents the spectrogram of Demucs separated predominant signal. From the figure, it is clear that our proposed method using CycleGAN correctly captures the predominant tracks than the state-of-the-art Demucs model.

Figure 8 represents the class-wise performance of our proposed method with enhancement with the

Table 3 Objective evaluation metrics for IRMAS and ADC2004 dataset

	DEMUCS						PROPOSED METHOD					
Class	**SDR (Avg)**	**SDR (Max)**	**SAR (Avg)**	**SAR (Max)**	**ISR (Avg)**	**ISR (Max)**	**SDR (Avg)**	**SDR (Max)**	**SAR (Avg)**	**SAR (Max)**	**ISR (Avg)**	**ISR (Max)**
IRMAS												
Flu	4.28	8.75	5.45	9.77	4.53	6.25	4.53	5.72	3.64	7.64	3.42	5.75
Gac	3.34	5.74	4.74	5.63	3.63	5.55	4.05	4.48	3.25	5.32	2.34	3.75
Sax	3.68	5.44	4.02	5.78	3.84	5.45	2.33	5.16	5.40	10.74	2.39	2.44
Tru	6.54	5.89	4.87	9.67	6.23	7.75	3.54	6.65	2.42	8.91	6.67	5.57
Voice	7.59	9.68	7.43	11.07	6.23	9.45	5.44	7.94	6.01	8.53	5.78	7.90
Average	5.08	7.10	5.30	8.38	4.89	6.89	3.98	5.99	4.14	8.23	4.12	5.08
ADC2004												
Daisy	3.28	4.25	3.33	3.72	2.73	2.95	0.90	1.22	0.43	1.86	1.10	1.46
Jazz	3.34	5.74	4.74	5.63	3.63	5.55	1.88	2.98	2.60	3.92	2.13	2.85
Midi	4.52	3.74	5.02	3.88	3.84	5.05	0.95	1.36	3.00	4.04	1.19	1.36
Opera	5.34	3.68	4.97	6.67	4.63	4.55	2.07	3.36	4.27	6.43	2.52	3.92
Pop	7.12	6.58	6.63	8.07	6.83	7.45	3.19	4.13	3.98	4.93	3.64	4.31
Average	4.72	4.79	4.94	5.59	4.33	5.11	1.79	2.61	2.86	4.24	2.12	2.78

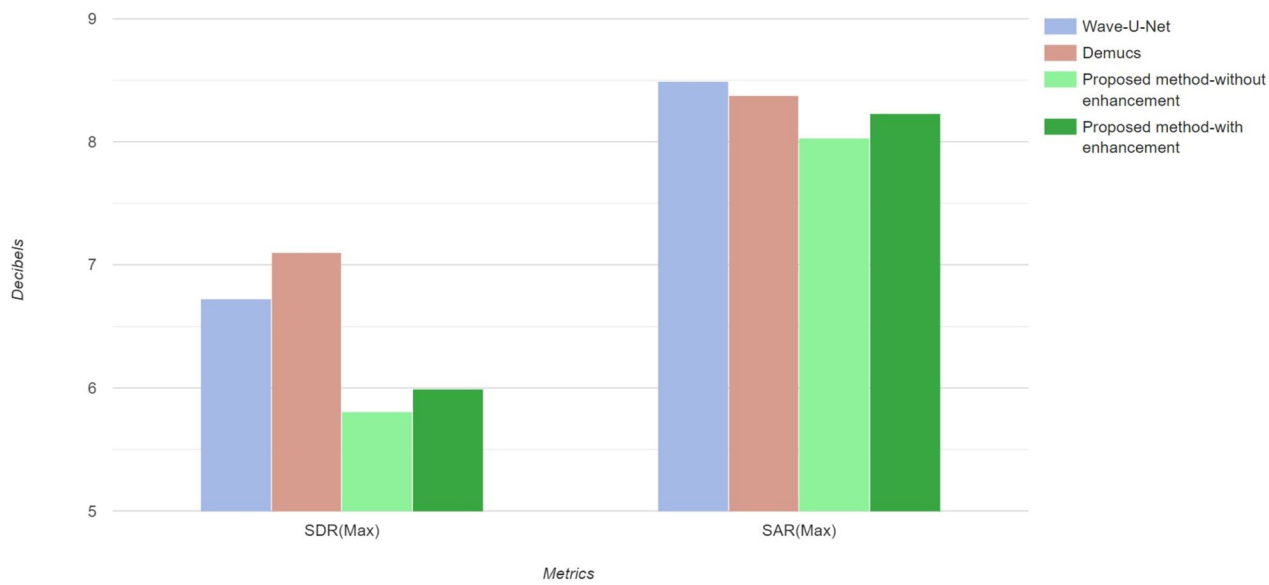

Fig. 5 Metrics comparison of the proposed method with Wave-U-Net and Demucs

Table 4 Comparison of proposed method with and without intelligent enhancement module

Metrics	Demucs	Wave-U-Net	Proposed method without enhancement	Proposed method with enhancement
SDR	7.10	6.72	5.81	5.99
SAR	8.38	8.49	8.03	8.22
ISR	6.89	6.75	5.07	5.08

state-of-the-art Demucs and the Wave-U-Net models. Instrument class saxophone and trumpet show better performance than the Demucs and Wave-U-Net models in both SDR and SAR measures. Also, the instrument class acoustic guitar reports an average SDR of 2.86 and 3.34 for the Wave-U-Net model and Demucs whereas our proposed method reports an average SDR of and 4.05. Moreover, it is worth noting that the performance of the Wave-U-Net and Demucs algorithms is relatively better when the predominant source is a human singing voice. However, our proposed method shows almost similar performance for lead instruments

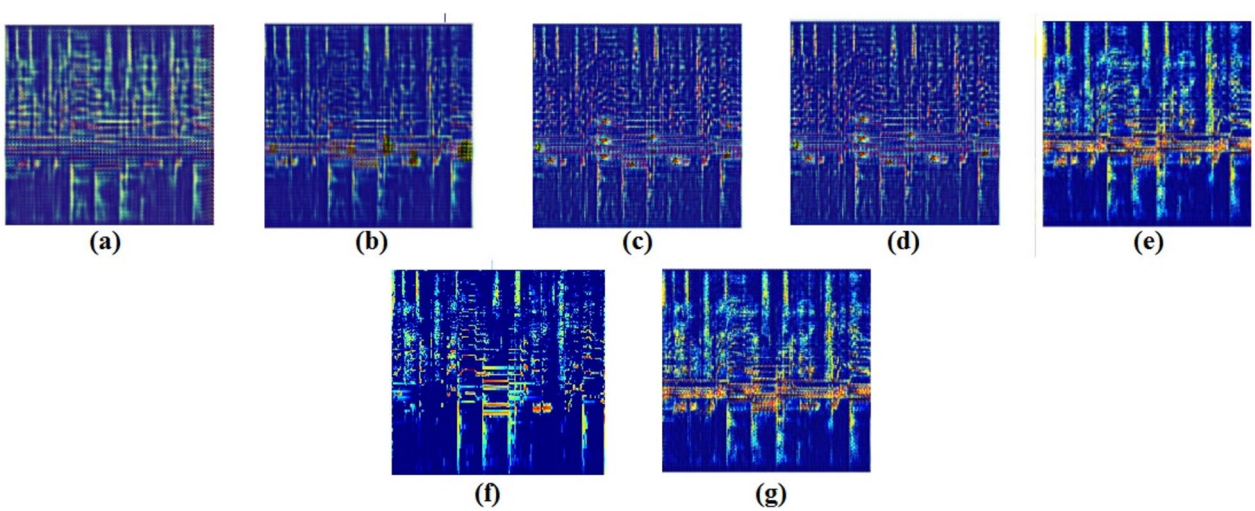

Fig. 6 Result of source separation. Generated spectrograms after **a** 100 epochs, **b** 200 epochs, **c** 300 epochs, **d** 400 epochs, **e** 500 epochs, **f** ground-truth spectrogram of predominant signal, and **g** CycleGAN generated spectrogram of the predominant signal

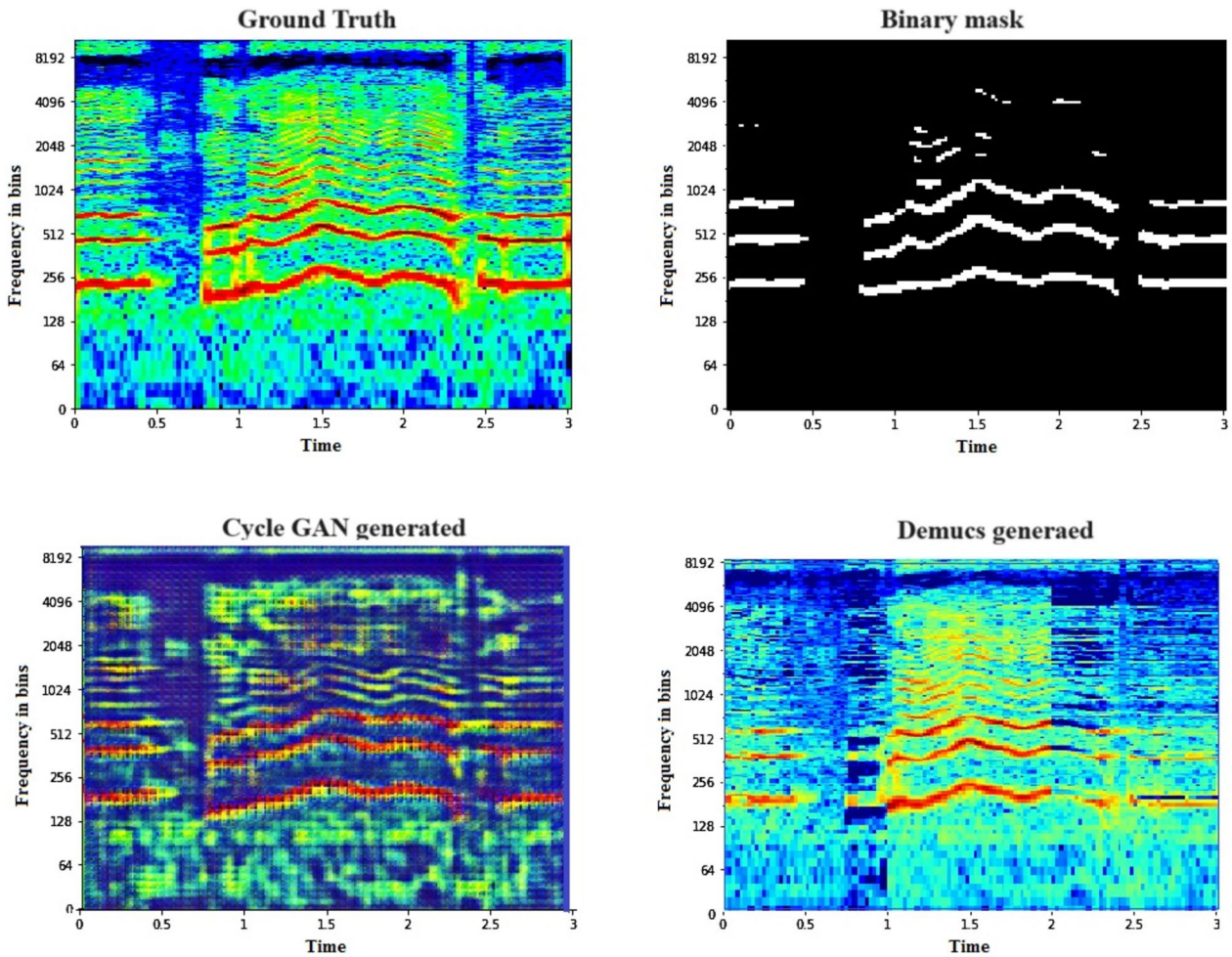

Fig. 7 Spectrogram generated by our proposed method compared with ground truth and by Demucs generated

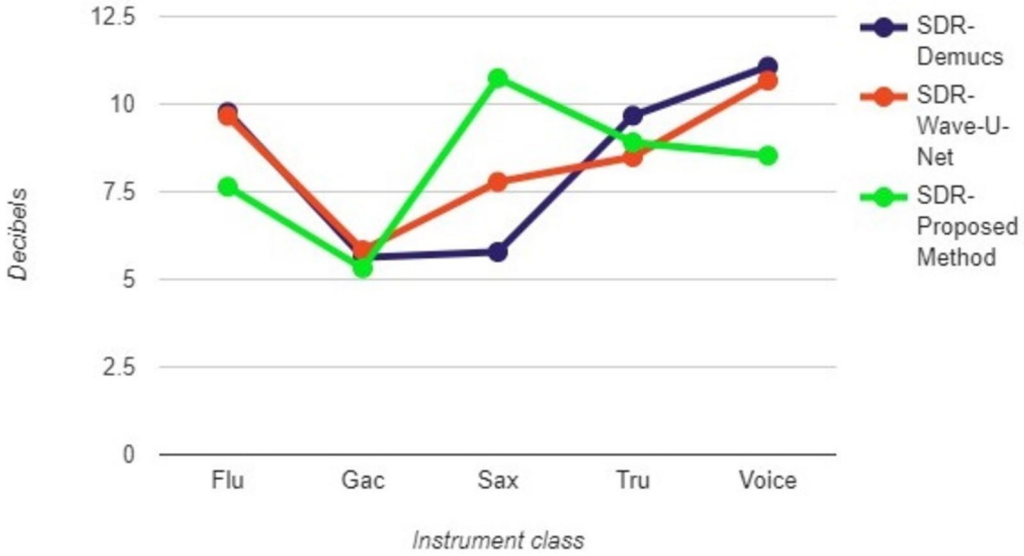

Fig. 8 Class-wise comparison of the proposed method with Wave-U-Net and Demucs

and voice. Since these are preliminary results, we expect further model refinement will lead to significant improvements. Also, our proposed model can effectively suppress noise it can be used for other applications like speech enhancement and speech source separation [34].

The subjective evaluation shows that the perceptibility of separated predominant sources from accompaniments is found promising, and a mean opinion score of 3.24 is obtained by evaluation. We hope that the proposed model may potentially benefit from more training data. It is also important to note that the performance of the model depends on the efficacy of the predominant pitch tracking algorithm in estimating an accurate pitch track. Our model is easily scalable to any number of instruments, including voice, and we can easily customize the architecture for a new speech/music mixture separation task. To summarize, the application of CycleGAN to open set predominant separation is the novelty of the proposed framework.

5 Conclusion

A new approach of CycleGAN-based predominant source separation algorithm is proposed. The predominant pitch track is estimated using the predominant pitch tracking algorithm and is used as the conditional input. The perceptual quality of the separated predominant spectrogram is enhanced using the conditional GAN. The model successfully separates the predominant voice and the leading instrument from accompaniments. The performance of the proposed method is compared with the state-of-the-art Wave-U-Net model and shows competing performance in both subjective and objective measures.

Acknowledgements
The authors would like to acknowledge Juan J. Bosch, Ferdinand Fuhrmann, and Perfecto Herrera (Music Technology Group - Universitat Pompeu Fabra) for developing the IRMAS dataset and making it publicly available.

Authors' contributions
LCR and RR jointly designed, implemented, and interpreted the computer simulations. All authors contributed to writing the manuscript and further read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
The datasets generated and/or analyzed during the current study are available in the Zenodo repository (https://www.upf.edu/web/mtg/irmas) and are publicly available.

Declarations

Competing interests
The authors declare that they have no competing interests.

Received: 2 February 2023 Accepted: 7 November 2023
Published online: 24 November 2023

References

1. J.H. McDermott, The cocktail party problem. Curr. Biol. **19**(22), 1024–1027 (2009)
2. D. Stoller, S. Ewert, S. Dixon, *in IEEE Int. Conf. on Acoustics, Speech and Signal Processing*. Adversarial semi-supervised audio source separation applied to singing voice extraction (ICAASP-2018), IEEE, pp. 2391-2395. https://doi.org/10.1109/ICASSP.2018.8461722
3. Z. Rafii, A. Liutkus, F.-R. Stöter, S.I. Mimilakis, D. FitzGerald, B. Pardo, An overview of lead and accompaniment separation in music. IEEE/ACM Trans. Audio Speech Lang. Process. **26**(8), 1307–1335 (2018)
4. A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications. Neural Netw. **13**(4–5), 411–430 (2000)
5. B. Nasersharif, S. Abdali, in *The Int. Symposium on Artificial Intelligence and Signal Processing (AISP)*. Speech/music separation using non-negative matrix factorization with combination of cost functions (IEEE, 2015), pp. 107–111. https://doi.org/10.1109/AISP.2015.7123491
6. O. Yilmaz, S. Rickard, Blind separation of speech mixtures via time-frequency masking. IEEE Trans. Signal Process. **52**(7), 1830–1847 (2004)
7. E. Vincent, N. Bertin, R. Gribonval, F. Bimbot, From blind to guided audio source separation: how models and side information can improve the separation of sound. IEEE Signal Process. Mag. **31**(3), 107–115 (2014)
8. Y. Li, D. Wang, in *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005*. Detecting pitch of singing voice in polyphonic audio, vol 3 (IEEE, 2005), pp. iii/17-iii/20. https://doi.org/10.1109/ICASSP.2005.1415635
9. M. Ryynanen, T. Virtanen, J. Paulus, A. Klapuri, in *2008 IEEE International Conference on Multimedia and Expo*. Accompaniment separation and karaoke application based on automatic melody transcription (IEEE, 2008), pp. 1417–1420
10. M. Lagrange, L.G. Martins, J. Murdoch, G. Tzanetakis, Normalized cuts for predominant melodic source separation. IEEE Trans. Audio Speech Lang. Process. **16**(2), 278–290 (2008)
11. X. Li, K. Wang, J. Soraghan, J. Ren, in Proc. of 9th International conference on Artificial Intelligence in Music, Sound, Art and Design. Fusion of hilberthuang transform and deep convolutional neural network for predominant musical instruments recognition (2020), pp. 80–89. https://doi.org/10.1007/978-3-030-43859-3_6
12. S. Uhlich, F. Giron, Y. Mitsufuji, in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*. Deep neural network based instrument extraction from music (IEEE, 2015), pp. 2135–2139. https://doi.org/10.1109/ICASSP.2015.7178348
13. E.M. Grais, M.U. Sen, H. Erdogan, in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*. Deep neural networks for single channel source separation (IEEE, 2014), pp. 3734–3738. https://doi.org/10.1109/ICASSP.2014.6854299
14. Y. Luo, N. Mesgarani, in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*. Tasnet: time-domain audio separation network for real-time, single-channel speech separation (IEEE, 2018), pp. 696–700. https://api.semanticscholar.org/CorpusID:4923261
15. S.I. Mimilakis, K. Drossos, J.F. Santos, G. Schuller, T. Virtanen, Y. Bengio, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Monaural singing voice separation with skip-filtering connections and recurrent inference of time-frequency mask (IEEE, 2018), pp. 721–725. https://doi.org/10.3390/sym12061051
16. Y.C. Subakan, P. Smaragdis, in IEEE Int. Conf. on Acoustics, Speech and Signal Processing. Generative adversarial source separation (IEEE, 2018), pp. 26–30
17. Y. Luo, N. Mesgarani, in Interspeech. Real-time single-channel dereverberation and separation with time-domain audio separation network (2018), pp. 342-346. https://doi.org/10.21437/Interspeech.2018-2290
18. E. Tzinis, Z. Wang, P. Smaragdis, in *IEEE 30th Int. Workshop on Machine Learning for Signal Processing (MLSP)*. Sudo rm-rf: efficient networks for universal audio source separation (IEEE, 2020), pp. 1–6. https://doi.org/10.1109/MLSP49062.2020.9231900
19. S. Inoue, H. Kameoka, L. Li, S. Makino, in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*. Sepnet: a deep separation matrix prediction network for multichannel audio source separation (IEEE, 2021), pp. 191–195. https://doi.org/10.1109/ICASSP39728.2021.9414884

20. D. Stoller, S. Ewert, S. Dixon, Wave-u-net: a multi-scale neural network for end-to-end audio source separation. (2018). arXiv preprint arXiv:1806.03185
21. C. Donahue, J. McAuley, M. Puckette, Adversarial audio synthesis. (2018). arXiv preprint arXiv:1802.04208
22. H. Phan, I.V. McLoughlin, L. Pham, O.Y. Chén, P. Koch, M. De Vos, A. Mertins, Improving gans for speech enhancement. IEEE Signal Process. Lett. **27**, 1700–1704 (2020)
23. C. Donahue, B. Li, R. Prabhavalkar, in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*. Exploring speech enhancement with generative adversarial networks for robust speech recognition (IEEE, 2018), pp. 5024–5028. https://doi.org/10.1109/ICASSP.2018.8462581
24. S. Pascual, A. Bonafonte, J. Serra, Segan: Speech enhancement generative adversarial network. (2017). arXiv preprint arXiv:1703.09452
25. J. Salamon, E. Gómez, Melody extraction from polyphonic music signals using pitch contour characteristics. IEEE Trans. Audio Speech Lang. Process. **20**(6), 1759–1770 (2012)
26. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets. Adv. Neural Inf. Process. Syst. **63**(11), 27 (2014). https://doi.org/10.1145/3422622
27. J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, in *Proc. of the IEEE Int. Conf. on computer vision*. Unpaired image-to-image translation using cycle-consistent adversarial networks (2017), pp. 2223–2232. https://doi.org/10.1109/ICCV.2017.244
28. M. Mirza, S. Osindero, Conditional generative adversarial nets. (2014). arXiv preprint arXiv:1411.1784
29. E. Grinstein, N.Q. Duong, A. Ozerov, P. Pérez, in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. Audio style transfer (IEEE, 2018), pp. 586-590. https://doi.org/10.1109/ICASSP34228.2018
30. D. Ulyanov, V. Lebedev, Audio texture synthesis and style transfer. (2016). https://dmitryulyanov.github.io/audio-texture-synthesis-and-style-transfer. Accessed 3 Mar 2023
31. F. Fuhrmann, P Herrera, in *Proc. of 13th Int. Conf. on Digital Audio Effects DAFx10*. Polyphonic instrument recognition for exploring semantic similarities in music (2010), pp. 1–8. http://mtg.upf.edu/files/publications/ffuhrmann_dafx10_final_0.pdf.
32. S. Jo, C.D. Yoo, in Proceedings of International Society for Music Information Retrieval (ISMIR). Melody extraction from polyphonic audio based on particle filter (Citeseer, 2010), pp. 357–362
33. D. Barry, G. Kearney, in Audio Engineering Society Conf.: 35th Int. Conf.: Audio for Games. Localization quality assessment in source separation-based upmixing algorithms. Audio Engineering Society, 2009, (AES 35th International Conference, London, UK, 2009), pp. 11–13
34. S. Joseph, R. Rajan, in *Circuits, Systems, and Signal Processing*. Cycle gan-based audio source separation using time-frequency masking (2022), pp. 1–18. https://doi.org/10.1007/s00034-022-02178-1a
35. P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, in *Proc. of the IEEE Conf. on computer vision and pattern recognition*. Image-to-image translation with conditional adversarial networks (2017), pp. 1125-1134. https://doi.org/10.1109/CVPR.2017.632
36. A. Défossez, N. Usunier, L. Bottou, F. Bach, Music source separation in the waveform domain. (2019). arXiv preprint arXiv:1911.13254

Publisher's Note