Electronics and Electrical Engineering

https://www.aimspress.com/journal/ElectrEng

AIMS Electronics and Electrical Engineering, 8(3): 292–310.
DOI: 10.3934/electreng.2024013
Received: 04 March 2024
Revised: 22 May 2024
Accepted: 23 May 2024
Published: 03 June 2024

Research article

A hybrid CNN-LSTM model with adaptive instance normalization for one shot singing voice conversion

Assila Yousuf*and David Solomon George

Department of Electronics and Communication Engineering, Rajiv Gandhi Institute of Technology, Kottayam, Kerala, 686501, India (Affiliated to APJ Abdul Kalam Technological University, Kerala)

* **Correspondence:** Email: assilayousuf@rit.ac.in.

Abstract: Singing voice conversion methods encounter challenges in achieving a delicate balance between synthesis quality and singer similarity. Traditional voice conversion techniques primarily emphasize singer similarity, often leading to robotic-sounding singing voices. Deep learning-based singing voice conversion techniques, however, focus on disentangling singer-dependent and singer-independent features. While this approach can enhance the quality of synthesized singing voices, many voice conversion systems still grapple with the issue of singer-dependent feature leakage into content embeddings. In the proposed singing voice conversion technique, an encoder decoder framework was implemented using a hybrid model of convolutional neural network (CNN) accompanied by long short term memory (LSTM). This paper investigated the use of activation guidance and adaptive instance normalization techniques for one shot singing voice conversion. The instance normalization (IN) layers within the auto-encoder effectively separated singer and content representations. During conversion, singer representations were transferred using adaptive instance normalization (AdaIN) layers. This singing voice system with the help of activation function prevented the transfer of singer information while conveying the singing content. Additionally, the fusion of LSTM with CNN can enhance voice conversion models by capturing both local and contextual features. The one-shot capability simplified the architecture, utilizing a single encoder and decoder. Impressively, the proposed hybrid CNN-LSTM model achieved remarkable performance without compromising either quality or similarity. The objective and subjective evaluation assessments showed that the proposed hybrid CNN-LSTM model outperformed the baseline architectures. Evaluation results showed a mean opinion score (MOS) of 2.93 for naturalness and 3.35 for melodic similarity. These hybrid CNN-LSTM techniques allowed it to perform high-quality voice conversion with minimal training data, making it a promising solution for various applications.

Keywords: one-shot singing voice conversion; instance normalization; AdaIN; AGAIN; hybrid CNN-LSTM model

1. Introduction

Singing voice conversion (SVC) is a technique that modifies the singing voice of a reference singer to sound like the voice of the target singer while keeping the phonetic content unchanged [1]. The converted singing voice sounds as if the target is performing the same lyrical composition as the source. SVC finds applications in the entertainment industry.

Voice conversion systems are mainly categorized as parallel [2] and nonparallel voice conversion systems [3,4]. In a parallel voice conversion system, the training data consists of both the reference and target speakers (singers) performing the same lyrical content. In contrast, nonparallel voice conversion systems, do not require both speakers to utter the same sentence set during training. Conventionally, many voice conversion methods rely on parallel training data, which necessitates frame level alignment between source and target utterances. However, collecting perfectly aligned parallel training data in real life scenarios can be extremely challenging. To tackle this problem, recent studies are focused on unsupervised voice conversion techniques that utilize nonparallel training data for voice conversion [5]. The SVC Challenge (SVCC) [6] focuses on comparing and understanding various voice conversion systems for SVC based on a shared dataset.

Recently, neural network (NN) based voice conversion methods such as deep neural network (DNN) [7], recurrent neural network (RNN) [8], and convolutional neural network [9] have been proposed [10, 11]. Additionally, there exist different approaches for style transfer in images. Style transfer involves transferring the style of an input image to another image without changing the content of the former image. Interestingly, these techniques can also be applied to non-parallel voice conversion tasks. Generative adversarial networks (GANs) [12, 13] - originally developed for image translation - can be effectively applied to audio data for voice conversion. Notable nonparallel many-to-many GAN-based approaches include Wasserstein generative adversarial network (WD-GAN) [14], cycle-consistent generative adversarial networks (CycleGAN) [15] and starGAN [16]. In these many-to-many nonparallel techniques, both source and target singing voices must be included in the training process.

Exploring the latest advancements in research focuses on image enhancement tasks, specifically super-resolution and inpainting using deep learning, attention mechanisms, and multi-scale features [17–19]. DNNs play a central role in these papers, learning complex mappings from low-resolution or incomplete images to high-resolution or complete versions [20, 21]. Focusing on image inpainting, Chen et al. [22] proposes a noise-robust voice conversion model. Users can choose whether to retain or remove background sounds during conversion. The model combines speech separation and voice conversion modules. Tomasz et al. [23] discusses techniques for seamlessly transferring a speaker's identity to another speaker while preserving speech content. Another study explores using cosine similarity between x-vector speaker embeddings as an objective metric for evaluating SVC [24]. The system preprocesses the source singer's audio to obtain melody features via F0 contour, loudness curve, and phonetic posteriorgram.

To disentangle singer identity from linguistic information, auto-encoders can be used. Thus, encoder-decoder-based networks are also proposed for unsupervised voice conversion. These auto-encoder-based techniques include variational auto-encoder (VAE) [25], cycle-consistent variational auto-encoder (CycleVAE) [26, 27], and variational auto-encoding Wasserstein generative adversarial network (VAWGAN) [28]. Although these systems generate high-quality singing voices, their major limitation is that they can only be applied to many-to-many voice conversion tasks. These methods are inefficient for SVC when the reference and target singers are absent in the training process. The robust one-shot SVC model [29] relies on GANs to accurately recover the target pitch by matching pitch distributions. Adaptive instance normalization (AdaIN)-skip conditioning further enhances the model's performance. Unlike traditional voice cloning tasks, which modify audio waveform to match a desired voice specified by reference audio, a novel task called visual voice cloning (V2C) [30] bridges the gap between voice cloning and visual information.

Some recent works on voice conversion have focused on one-shot voice conversion. Examples include zero-shot voice style transfer with only auto-encoder loss (AutoVC) [31], AdaGAN-VC [32], and two-level nested U-structure VC (U2VC) [33, 34]. In one-shot voice conversion, either or both speakers that are unseen in the training data can be used for the inference. Recently, voice conversion in the speech domain has utilized one-shot voice conversion techniques. These approaches focus on the separation of speaker and content information. Audio features such as mel-cepstral coefficients (MCEPS), aperiodicities and fundamental frequencies are extracted using vocoders. The voice conversion process involves removing speaker dependent features from the source speaker's utterance and incorporating the target speaker's attributes. AutoVC introduces a vanilla auto-encoder with designed information bottleneck and a pretrained speaker encoder for decoupling the speaker and content information. Meanwhile, a U-Net architecture combining vector quantization (VQ) and instance normalization (IN) (VQVC+) [35] successfully separates linguistic information using vector quantization. However, content leakage remains a significant limitation.

AdaGAN-VC requires adversarial training for the separation of speech attributes, which causes instability problems during the training process. This issue is resolved in AdaIN-VC [36] and activation guidance and adaptive instance normalization (AGAIN)-VC, which utilize instance normalization techniques to removing the speaker information. AdaIN was first introduced in [37] for style transfer in image translation networks. It addresses the limitations of existing normalization methods in deep learning. AdaIN extends the popular normalization technique, IN by incorporating style information from a reference image. Specifically, it maps the normalized mean and standard deviation of the content image to match those of the styled image. AdaIN-VC comprises two encoders with AdaIN for the disentanglement of information and one decoder. Lian et al. [38] proposed arbitrary voice conversion without any supervision, which is achieved using instance normalization and adversarial learning. Meanwhile, masked auto-encoder (MAE-VC) [39] is an end-to-end masked auto-encoder that converts the speaker style of the source speech to that of the target speech while maintaining content consistency. In Activation Guidance and Adaptive Instance Normalization (AGAIN)-VC [40], speaker and content information are disentangled with the help of a single encoder. These approaches are introduced for voice conversion in speech domain. In this paper, one-shot SVC is proposed using a combination of convolutional layers and LSTM architecture. AdaIN and AGAIN techniques are applied for SVC, using them as baseline architectures.

These methods are designed for voice conversion in the speech domain. Only a few works have been proposed for the conversion of singing voice. This paper proposes a hybrid convolutional neural network with long short term memory (CNN-LSTM) model for one-shot SVC system using the AGAIN technique. The proposed AGAIN-SVC method requires a single encoder to separate the vocal timbre of the singer from the phonetic information. Remarkably, without any frame alignment procedures, one's singing voice can be converted into another person's voice using nonparallel data, even if both singers are unavailable during the training process. In this paper, two recent works on voice conversion - AdaIN and AGAIN - are applied for one-shot SVC and also used as baseline models. To improve conversion performance, a combination of convolution layers and LSTM layers is used for the encoder and decoder architecture. Simultaneously achieving high synthesis quality and maintaining singer similarity poses a challenge, as enhancing one often comes at the cost of the other. Many VC systems employ disentanglement techniques to separate singer and linguistic content information. However, some methods reduce the dimension or quantize content embeddings to prevent singer information leakage, inadvertently compromising synthesis quality. Activation guidance serves as an information bottleneck on content embeddings, allowing better control over singer-related features. Additionally, AdaIN dynamically adjusts normalization statistics during training, enhancing the model's flexibility.

The proposed technique significantly improves the delicate balance between synthesis quality and singer similarity. Its one-shot voice conversion relies on learned content features and adaptation mechanisms to transform the source speaker's voice into the desired target speaker's style, even when the target speaker is unseen during training. This simplifies the architecture, making it efficient and practical. Indeed, CNNs excel at extracting local features from audio data, but they fall short in modeling long-term dependencies over extended sequences. Recurrent architectures, such as LSTM and gated recurrent unit (GRU), address this limitation by maintaining hidden states across time steps. By adding them after the CNN blocks, it can introduce the ability to learn contextual information beyond local features. This can improve the model's understanding of speech patterns, intonation, and context.

For this paper, the main contributions are as follows: (i) Rapid voice transformation from an unseen target singer is used to directly convert the source singer's voice to the desired target singer's style. (ii) The one shot voice conversion capability simplifies the architecture, making it efficient and practical. (iii) AdaIN and activation mechanisms achieve a balance between synthesis quality and speaker similarity. (iv) Incorporating recurrent architecture enhances contextual learning and improves temporal modeling.

The paper is organized into four sections. The network architecture of the proposed SVC technique is described in Section 2. Experiment setup and evaluation results are discussed in the following section. Finally, the conclusion is presented in Section 4.

2. Proposed system

The AGAIN technique was first introduced for voice conversion. This paper combines the AGAIN technique with the proposed hybrid CNN-LSTM architecture for SVC.

2.1. Adaptive instance normalization

Internal covariance shift is defined as the variation in the distribution of each network layer input due to the alterations in the parameters of the previous layer. The training of earlier DNNs was affected by internal covariance shift, which in turn slowed down the training process. To alleviate this serious issue, batch normalization (BN) [41,42] was introduced. Consider N is the batch size, C is the number of channels, H is the height of each activation map, and W is the width of activation input. The activation layer input dimensions can be represented as $N * C * H * W$. Generally, normalization can be applied to activation layers by shifting the mean and scaling the variance. In BN, mean and variance for each channel are computed across all samples across both spatial dimensions. Thus, BN normalizes the activations across $N*H*W$ axes. The statistics of BN layers include batch-wise mean and variance. Interestingly, the style of an image can be represented using the BN statistics. Consequently, neural style transfer of two feature maps becomes possible by replacing the batch-wise mean and variance of the source image with those of the target image [43].

Instead of BN, IN was introduced in [44] for feed forward stylization tasks, achieving noticeable improvements in the generated stylized images. Unlike BN, IN can be applied at both training and testing time, ensuring consistency during training, transfer, and testing. IN normalizes across HXW axes by calculating the mean and variance across both spatial dimensions for each channel and each sample.

Let X and Y represent the feature map of reference and target waveform respectively. IN can be computed as follows:

$$IN(X) = \gamma(\frac{X - \mu(X)}{\sigma(X)}) + \beta \tag{2.1}$$

where β and γ are the learned affine parameters. Here, $\mu(X)$ is the mean and $\sigma(X)$ is the standard deviation computed across spatial dimensions independently for each feature channel and each sample.

AdaIN is simply an extension of IN. Unlike IN, it has no learnable affine transformations. While IN normalizes the input style to a single style, AdaIN normalizes the input to an arbitrarily given style. Thus, AdaIN conveys the style of the target as the reference style by simply adjusting the mean and standard deviation of the reference input to match those of the target input. AdaIN can be represented as:

$$AdaIN(X, \mu(Y), \sigma(Y)) = \sigma(Y)(\frac{X - \mu(X)}{\sigma(X)}) + \mu(Y) \tag{2.2}$$

Here, X and Y are content input and style input respectively. The AdaIN layer thus transfers the style of X with style of Y by scaling the normalized content input with the standard deviation of style input and shifts it with $\mu(Y)$.

2.2. *AdaIN-SVC*

AdaIN was proposed in for the arbitrary style transfer in real time. The idea of AdaIN was extended for the voice conversion of speech signals. In audio, the statistics (mean and variance) of the content features might represent phonemes, or musical notes. The style features could correspond to timbres or specific audio effects. By applying AdaIN, it can transfer the style characteristics of one audio signal onto the content of another. In the Eq 2.1 γ parameter adjusts the style features by scaling them whereas the β parameter shifts the content features by adding an offset. AdaIN-SVC employs a VAE with distinct encoders to encode both the singer-specific information and the phonetic content information. AdaIN approach can also be applied for the conversion of singing voices.

Consider the singing voice as the phonetic information sung by a singer with unique features defined as the singer identity information. While the lyrical information changes drastically after each frame, the singer information experiences only slight variations. Since the mean and variance over an entire singing waveform rarely change, singer information is expressed using the same.

AdaIN-SVC employs two encoders to encode the singer information and phonetic content information. The encoder takes a mel-spectrogram of the singing waveform as input frames. IN layers are added in each encoder block. These IN layers effectively separate the singer information from the lyrical content. To achieve this, the channel-wise mean (μ) and standard deviation (σ) is computed. The IN layer detaches the singer representations from the encoded phonetic content information. However, during the decoding process, the detached singer information is reintroduced to the AdaIN layer.

2.3. Proposed AGAIN-SVC

AdaIN-SVC uses a VAE with separate content and speaker encoders. It also leverages AdaIN to separate singer and content representations but relies on separate encoders. It either reduces the dimension or quantizes content embeddings to prevent singer information leakage. However, this strong information bottleneck can harm synthesis quality. AGAIN-SVC introduces a proper activation as an information bottleneck on content embeddings. It simplifies the architecture to an auto-encoder with a single encoder and a decoder. By using activation guidance, it can drastically improve the trade-off between synthesis quality and speaker similarity in converted speech.

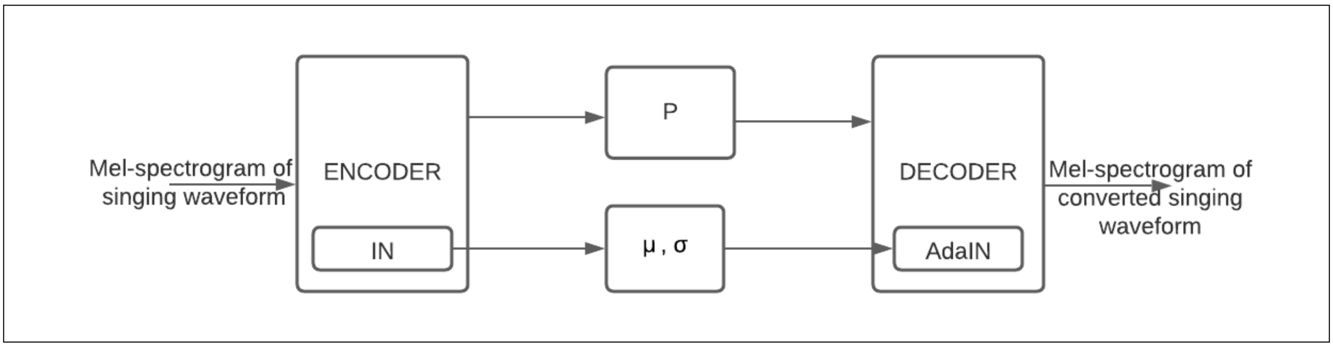


Figure 1. Training phase procedure of AGAIN SVC. Here, μ represents mean and σ is the standard deviation. P denotes linguistic information.

The proposed AGAIN-SVC is an auto-encoder-based model that disentangles the singer identity and phonetic information. The framework comprises a single encoder and a decoder. Unlike AdaIN-SVC, which utilizes separate encoders for phonetic content and singer information, AGAIN-SVC uses only one encoder. The singer information is considered time independent information, whereas the phonetic content information is time dependent. Components, the style transfer of singing data, becomes easier. To prevent information leakage without affecting the conversion performance, an activation guidance is used as an information bottleneck. The training procedure of the proposed AGAIN SVC system is shown in Figure 1. Instead of modeling raw singing audio data, mel-spectrograms are used. These lower resolution representations are easier to model than raw temporal audio and faithfully regenerate back to audio. The IN layer detaches the linguistic content and singer dependent features. Later, the singer embeddings are later added to the AdaIN layer in the decoding section. The evaluation procedure of AGAIN-SVC is illustrated in Figure 2. Given reference X and target Y, the AdaIN layer transfers the singer features (channel-wise mean and variance) from the reference to those of the target singer. Meanwhile, the content of the reference remains unchanged, and only the singer identity is transferred.

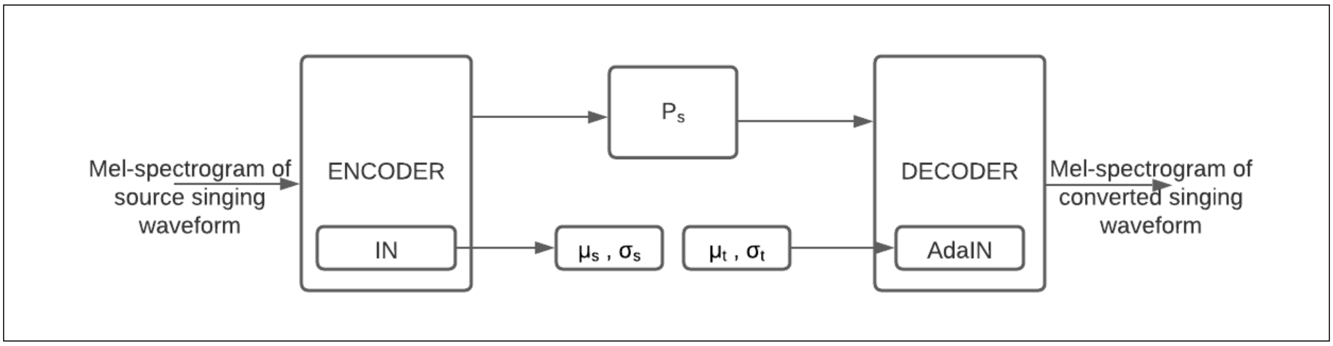

Mel-spectrogram of source singing waveform
ENCODER
IN
Ps
μs , σs
μt , σt
DECODER
AdaIN
Mel-spectrogram of converted singing waveform

Figure 2. Diagram of the evaluation phase of AGAIN SVC. The linguistic content denoted by P_s and the singer representation (μ_s and σ_s) of source data are separated and P_s is given into the decoder without changing. μ_s and σ_s are replaced with the target singer representation (μ_t and σ_t) and fed into the AdaIN layer of the decoder.

2.3.1. Activation guidance

The encoder phase acts as a content encoder, responsible for encoding the phonetic content information from the singing data. However, the mean and variance, which are time independent features, seldom carry any phonetic content information. To address this, IN layers in each encoder block remove the time-invariant information that represents the singer identity. To prevent any leakage of singer information along with the phonetic content embedding, an activation function is included as an information bottleneck. The sigmoid function with the hyperparameter value $\alpha = 0.1$ is chosen. The sigmoid function proves effective in disentangling the singer identity from phonetic content embedding while minimizing the reconstruction error. This activation function ensures that the encoding contains only the phonetic information.

2.3.2. Network architecture

The basic architecture is based on the AGAIN voice conversion system. It employs a U-net architecture, where both encoder and decoder utilize 1D convolutional layers. A CNN [45–47] or ConvNet is a feed-forward neural network. CNN contains a stack of convolutional layers, typically the hidden layer that performs convolutions. The proposed hybrid architecture combines a CNN with an RNN. In DNNs, especially those with many layers, gradients can diminish exponentially as they propagate backward. This phenomenon hinders effective training and can significantly prolong the learning process or even cause convergence failure. The vanishing gradient problem is particularly associated with activation functions like sigmoid. The specialized RNN architectures, LSTM and GRU address the vanishing gradient problem by introducing gating mechanisms and maintaining cell states. LSTMs [48] introduce gates (input, forget, and output gates) that control the flow of information within the cell and maintain them over longer sequences. It maintains a cell state that allows information to flow across time steps without vanishing gradients. GRUs [49] introduce two essential gates: the update gate and the reset gate. It computes a new hidden state by blending the previous hidden state and the updated input. This mechanism allows them to capture temporal dependencies effectively. The hybrid models integrate the RNN (either LSTM or GRU), which acts as a feedback network, with the feed-forward CNN. In these hybrid networks, CNN layers extract complex features, while LSTM or

GRU layers capture long-term dependencies.

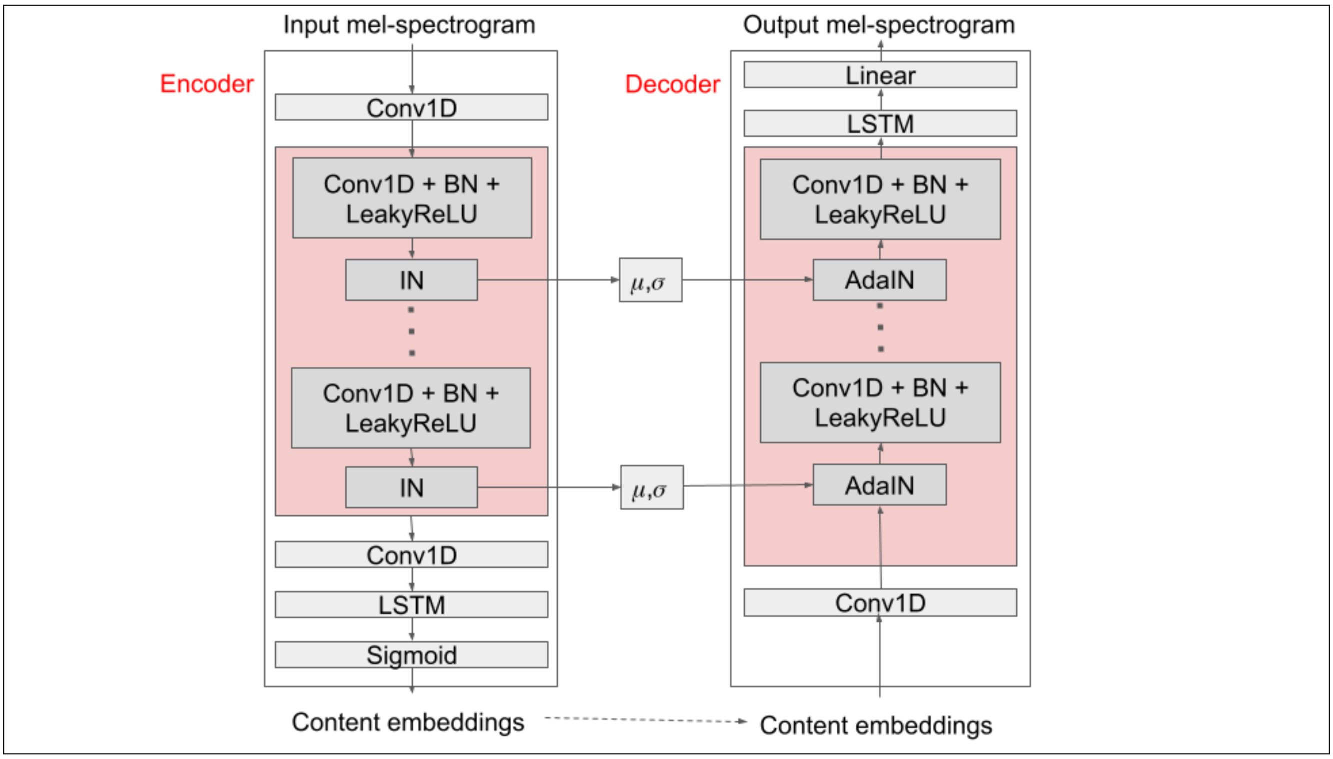


Figure 3. The network architecture of the proposed hybrid CNN-LSTM model.

The encoder and decoder shown in Figures 1 and 2 form a U-net architecture, as depicted in Figure 3. The input mel-spectrogram features pass through 1D convolution layers. Both the encoder and decoder uses 1D convolution blocks, which consist of 1D convolution layers followed by BN and leaky rectified linear unit (LeakyReLU) activation function. Skip connections between the encoder and decoder are formed by transferring speaker embeddings μ and σ between IN layers and AdaIN layers. These skip connections allow encoded features from the encoder to be directly forwarded to the decoder. By connecting the encoder and decoder, information from earlier layers can bypass the bottleneck (latent space) and directly influence the reconstruction process. This helps mitigate the vanishing gradient problem and facilitates better information flow. By allowing gradients to flow more freely, skip connections stabilize training and improve convergence. The content embeddings from the encoder is passed to the decoder, and the converted mel-spectrogram features are extracted from the decoder. LSTM layers are placed after convolution layers at both the encoder and decoder ends. To obtain hybrid CNN-GRU model, LSTM layers are replaced with GRU. Both LSTM and GRU are used in the same way and both of them have somewhat similar effects. The pseudocode algorithm for the proposed system is also included here.

2.4. *MelGAN*

MelGAN [50] is specifically designed for efficient audio waveform synthesis. Unlike autoregressive models (such as WaveNet and WaveRNN), which suffer from high computational complexity, MelGAN focuses on reducing this complexity. It achieves this by decomposing raw audio samples

Algorithm 1 Enhanced AGAIN-VC with ConvLSTM Pseudocode

```
Load preprocessed data (mel-spectrograms) for source and target speakers
Initialize model parameters (encoder, ConvLSTM layers, and decoder)
Encoder:
    Input: Mel-spectrogram of source speaker's speech
    Output: Content embeddings
    Encode the input mel-spectrogram using the encoder (including ConvLSTM layers)
Activation Guidance:
    Apply activation guidance to the content embeddings
    Restrict the flow of information to essential features
Adaptive Instance Normalization (AdaIN):
    Adapt the statistics (mean and variance) of content embeddings
    to match those of the target speaker
    Preserve linguistic content while adjusting for style
Decoder:
    Input: Adapted content embeddings and target speaker's style
    Output: Converted mel-spectrogram
    Decode the adapted content embeddings using the decoder
Loss Functions:
    L1 Loss: Minimize difference between input and output mel-spectrograms
    Perceptual Loss: Encourage perceptual similarity with target speaker's speech
    Total loss: Combination of L1 and perceptual losses
Training:
    Optimize model parameters using backpropagation and gradient descent
    Monitor training progress (e.g., total steps, evaluation steps)
Inference:
    Given a source mel-spectrogram, convert it to target speaker's style
    Output the converted mel-spectrogram
ConvLSTM Layers:
    Incorporate ConvLSTM layers within the encoder and decoder
    Enhance the temporal modeling capabilities by capturing spatio-temporal features
    Adjust hyperparameters (e.g., kernel size, hidden units) as needed
```

into learned basis functions and associated weights. As a result, MelGAN significantly reduces computational demands compared to other GAN-based neural vocoders, such as HiFi-GAN [51]. Moreover, MelGAN consistently produces high-quality audio that rivals other GAN-based vocoders. MelGAN comprises two main components: the generator and the discriminator. The primary objective of MelGAN is to generate high-quality audio waveforms from mel-spectrograms. The generator, being lightweight, efficiently converts mel-spectrograms into raw audio waveforms. Conversely, the discriminator plays the crucial role of distinguishing between real and generated audio. Through an adversarial training process, the generator continually improves the quality of the synthesized audio. Additionally, by conditioning the generator on specific attributes (such as speaker identity or emotion), MelGAN can synthesize audio with desired characteristics. To enable conditional synthesis, MelGAN requires paired data during training, allowing it to learn the mapping from source mel-spectrograms to target audio waveforms.

3. Results and discussion

The model is evaluated using the National University of Singapore (NUS) sung and spoken lyrics corpus (NUS-48E corpus) [52]. The corpus comprises speaking and singing voices from 48 English songs performed by six male and six female subjects, out of which 20 songs are unique. The total length of audio recordings is approximately 169 minutes, with 115 minutes dedicated to singing data and the remaining 54 minutes to speech data. From the NUS-48E database, nine singers are selected for training, while the remaining three singers (two male and one female singer) are reserved for testing. The audio is converted to 22050 Hz, and silent frames are removed. 80 bin mel-scale spectrograms are generated from the audio data as an acoustic feature. To generate the waveform back from the mel-spectrogram, MelGAN is used as the vocoder. MelGAN is a fully convolutional architecture within a GAN setup, employed for conditional audio waveform synthesis.

An adaptive moment estimation (ADAM optimizer) is used for training. The model is trained for 50,000 training steps with a learning rate of 0.0005. The batch size is 32. For performance evaluation of the proposed architecture, AdaIN-SVC and AGAIN-SVC are used as baseline models.

3.1. Objective evaluation

SVC is primarily associated with the transformation of MCEPS. For evaluation purposes, it is necessary to calculate the distortion between the original and transformed MCEPS features. Mel-cepstral distortion (MCD) [53, 54] is a widely used measure for evaluating synthesized voice. MCD represents the Euclidean distance between the real and converted MCEPS features. It is defined as follows:

$$MCD(dB) = \frac{10}{Nln10} \sum_{n=1}^{N} \sqrt{2 \sum_{d=1}^{D} (m_{c(n,d)} - m_{t(n,d)})^2} \tag{3.1}$$

where $m_{c(n,d)}$ and $m_{t(n,d)}$ are the d^{th} coefficients of the converted and target MCEPS features respectively, at the n^{th} frame. D = 24 is the dimension of the MCEPS. N is the total number of frames. MCD values are tabulated in Table 1 for four possible conversions, male to male (MM), male to female (MF), female to male (FM), and female to female (FF). Conversions with lower MCD values exhibit smaller distortion and better performance. Interestingly, the proposed method for FF conversion yields the best results. Female voices often share more similar acoustic characteristics with each other compared to male voices. These shared characteristics include pitch range, formant frequencies, and overall timbre. When converting from one female voice to another, the transformation is less drastic, that might lead to better results.

For a comparative study with state of the art techniques, experiments are conducted using various machine learning methods in the field of speech conversion, including VAE, VAWGAN, CycleGAN, and cycle-consistent boundary equilibrium generative adversarial network (CycleBEGAN), along with the baseline models (Table 2). The evaluation measures include MCD, root mean square error (RMSE) of mel-spectrogram features from real and converted voices, and log spectral distortion (LSD). LSD

Table 1. MCD results for MM, MF, FM, and FF conversions.

System	MCD (dB)				
	MM	MF	FM	FF	Average
AdaIN-SVC	8.20	7.92	8.30	8.79	8.30
AGAIN-SVC	6.32	6.31	6.37	6.92	6.48
Hybrid CNN-GRU	5.92	5.80	5.95	5.33	5.75
Hybrid CNN-LSTM	5.99	5.56	5.94	4.91	5.60

Table 2. Comparative study with the state of the art techniques.

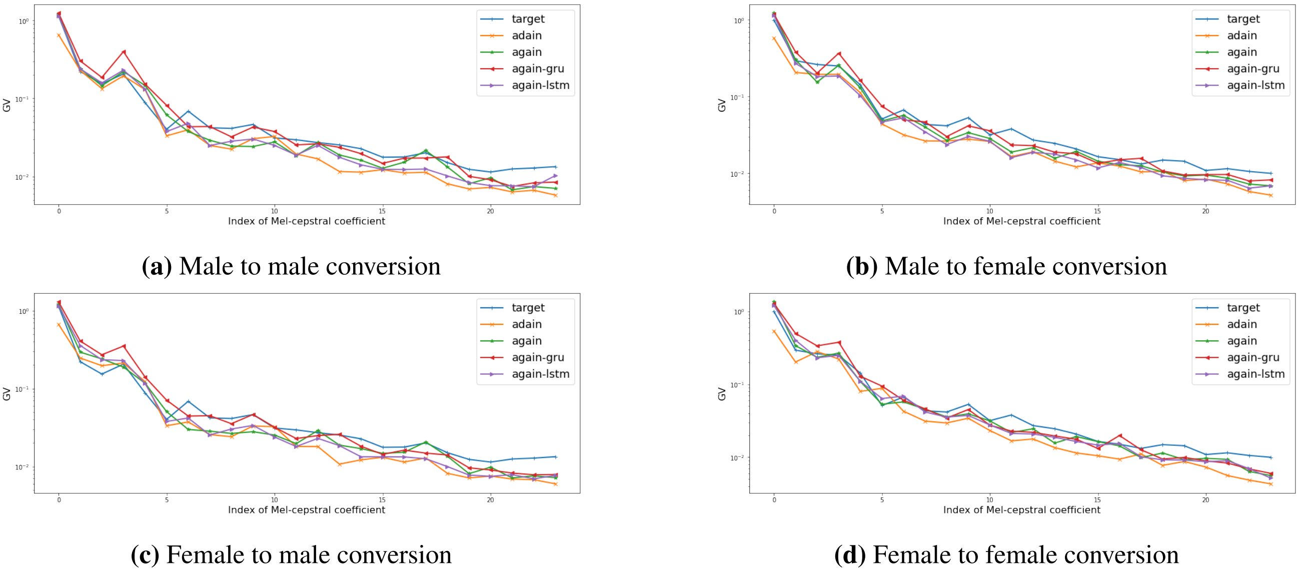
System	MCD(dB)	RMSE(dB)	LSD(dB)
VAE	8.20	3.14	11.13
VAWGAN	6.32	3.36	10.14
CycleGAN	5.92	3.32	10.91
CycleBEGAN	5.99	3.16	9.95
AdaIN-SVC	8.30	4.5	11.24
AGAIN-SVC	6.48	3.14	10.29
Hybrid CNN-GRU	5.75	2.47	10.11
Hybrid CNN-LSTM	5.6	2.04	9.80

computes the difference between the linear prediction coding (LPC) log power spectra from the original and converted singing voices. The converted singing voice retains spectral features similar to the original if the distortion is minimized.

From the results, it can be seen that the hybrid models yield more promising outcomes than the baseline methods. Specifically, hybrid CNN-RNN systems demonstrate superior performance compared to the baseline AdaIN-SVC and AGAIN-SVC models. Although the GRU model is less complex and faster to train, the fusion of LSTM architecture produces a more similar converted voice aligned with the target. This phenomenon arises because the LSTM network requires more trainable parameters, which, in turn, enhance the system's efficiency during training.

Additionally, the global variance (GV) distribution [55] plot for synthesized and target singing voices serves as a valuable measure to assess their closeness. GV visualizes spectral features in terms of variance distribution. For instance, Figure 4 illustrates the variance distribution plot for target singing voice and converted singing voice across four conversions. The results affirm that the converted singing voices closely match the target singing voice.

Incorporating LSTM or GRU with CNN can enhance voice conversion models by capturing both local and contextual features. The choice depends on specific requirements and available resources. LSTM has more parameters and is capable of capturing intricate long-term dependencies, while GRU simplifies the architecture by having fewer gates and parameters. If the system prioritizes accuracy and has sufficient data, LSTM might provide better results. If efficiency is crucial, GRU could be a better choice. GRU typically trains faster than LSTM due to its simpler structure. If you have limited training resources, GRU might be preferable. LSTM might be more suitable for tasks requiring precise modeling of long-range dependencies (e.g., music composition). GRU could be preferable for real-time applications where efficiency matters (e.g., voice-controlled systems).



(**a**) Male to male conversion

(**b**) Male to female conversion

(**c**) Female to male conversion

(**d**) Female to female conversion

Figure 4. Global variance distribution of target and converted singing voice for each frequency index.

3.2. Subjective evaluation

Objective evaluation measures alone have several limitations because the performance of the synthesized singing voice depends on the perceptual abilities of humans. To ensure the quality of the synthesized singing voice, subjective evaluation is also performed. In this subjective evaluation, 24 participants (12 male and 12 female), all without any hearing problems, took part in listening tests. Each participant evaluated a mean opinion score (MOS) for three attributes: naturalness, melodic similarity, and phoneme intelligibility, along with XAB preference tests related to singer similarity. XAB preference testing is a method used to compare two different stimuli (A and B) with an unknown third stimulus (X). During the listening tests, participants were required to listen to two randomly chosen songs from each conversion.

For the XAB test on singer similarity, participants listened to a target song and their converted singing voice in two systems. They then had to choose their preferred system based on singer similarity with the target. The experiments included intra-gender (MM or FF) and inter-gender (MF or FM) conversions. Initially, the preference test involved converted songs from the AdaIN-SVC and AGAIN-SVC models. Subsequently, comparison was conducted between AdaIN-SVC vs hybrid CNN-LSTM, as well as between AGAIN-SVC vs hybrid CNN-LSTM. The results of the XAB preference test are depicted in Figure 5, Figure 6, and Figure 7.

Additionally, MOS tests were conducted, where each participant provided a score on a scale of 1 to 5 for each system (Figure 8). A score of 1 corresponds to the worst case, while a score of 5 represents the best case. Naturalness reflects how closely the converted singing voice resembles a natural human voice. The melodic similarity attribute is evaluated based on the resemblance in melody between the target voice and the converted voice. Finally, MOS on phoneme intelligibility provides insights into the clarity of phonemes in the converted singing voice.

The singer similarity between AdaIN-SVC and AGAIN-SVC exhibits somewhat equal preferences. However, from the preference test results comparing AdaIN-SVC with the proposed hybrid CNN-

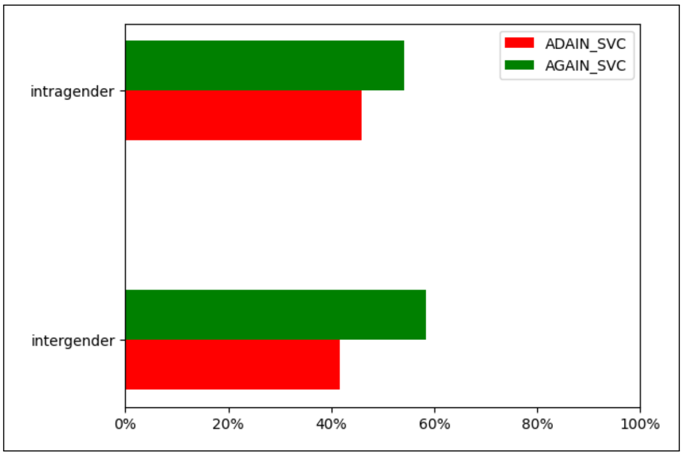


Figure 5. XAB preference test results between AdaIN-SVC and AGAIN-SVC.

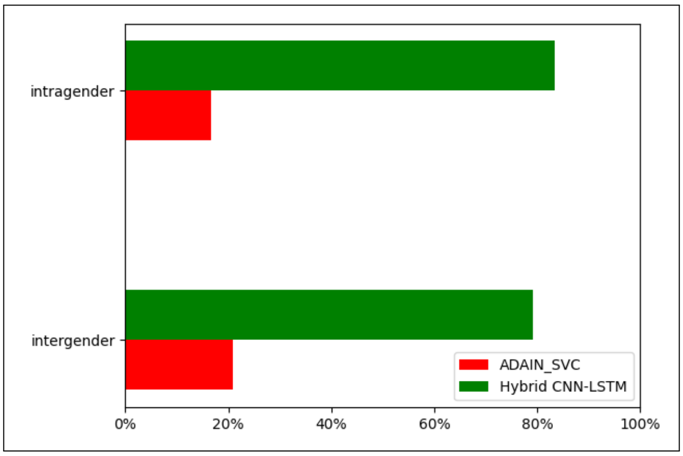


Figure 6. XAB preference test results between AdaIN-SVC and the proposed hybrid CNN-LSTM.

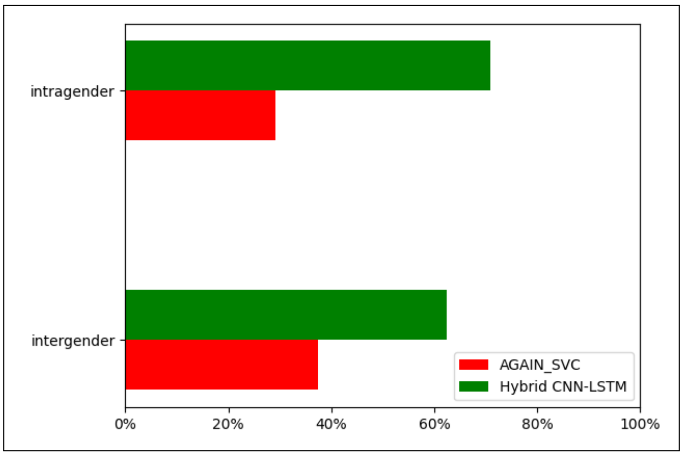


Figure 7. XAB preference test results between AGAIN-SVC and the proposed hybrid CNN-LSTM.

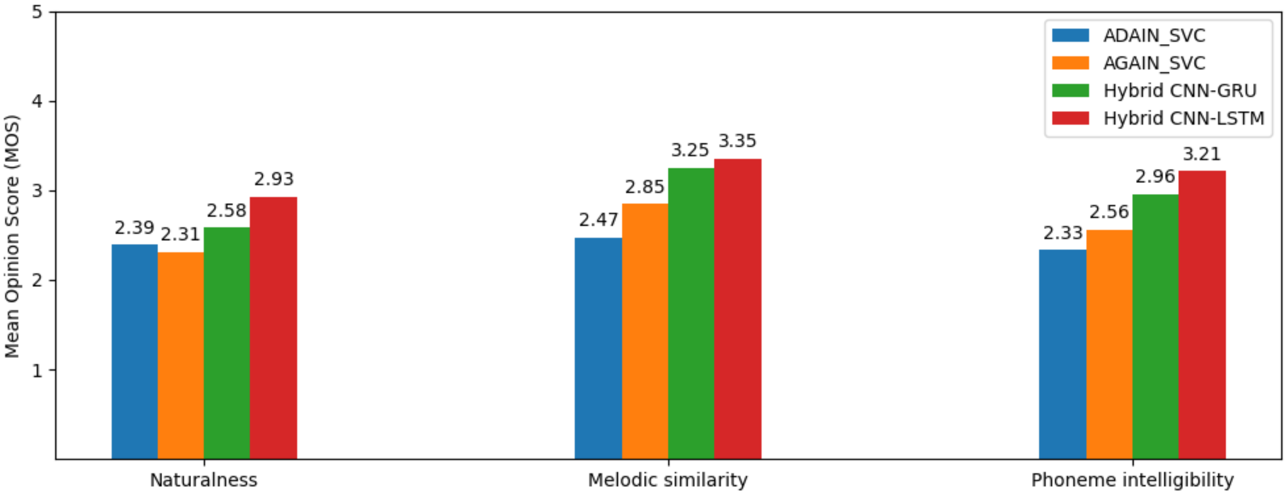


Figure 8. MOS for three attributes; naturalness, melodic similarity, and phoneme intelligibility.

LSTM system, the latter shows greater resemblance to the target singer's voice. Specifically, for inter-gender conversions, the hybrid CNN-LSTM system achieves 86.67% similarity, whereas AdaIN-SVC achieves only 13.3%. For intra-gender conversions, the hybrid CNN-LSTM system achieves 93.3% similarity, while AdaIN-SVC lags behind at 6.7%. Additionally, the hybrid CNN-LSTM model is preferred over AGAIN-SVC. Analyzing the percentage preferences reveals substantial differences: AdaIN-SVC vs. the hybrid model exhibits large disparities, indicating that AdaIN-SVC has the least preference. Meanwhile, AGAIN-SVC vs. the hybrid model comparison shows that the hybrid model is highly preferred. Furthermore, the melodic similarity and phoneme intelligibility of the converted singing voice in the proposed hybrid systems outperform the baseline AdaIN-SVC and AGAIN-SVC models (Figure 8). This improvement is attributed to the activation guidance, which mitigates the phonetic content loss associated with the latter. However, there is room for improvement in the naturalness of the overall system.

4. Conclusions and future work

In the proposed work, a combination of the CNN model and RNN model is used for one shot SVC, employing adaptive normalization layers and activation guidance techniques. IN layers remove singer information from the source singing mel-spectrogram, while AdaIN layers add target singer information to the content representation. Activation guidance prevents singer information leakage, ensuring better control over singer-related features. This technique disentangles singer and content information without leaking singer information into the content embeddings. By incorporating the recurrent architectures such as LSTM and GRU, which excel at capturing long-term dependencies in sequential data, this paper achieves high-quality converted singing voice while maintaining singer characteristics. The one-shot capability simplifies the architecture, making it efficient and practical. The conversion performance of singing voice is assessed through objective and subjective evaluation. For the evaluation, voice conversion techniques such as AdaIN and AGAIN are used as baseline architectures. It is evident that the fusion of CNN and LSTM model consistently yields better results across all experiments. The MCD is least for the proposed hybrid CNN-LSTM model (5.6dB), ensuring superior conversion. The majority of people preferred the proposed conversion technique, with AdaIN-SVC being the least preferred, followed by AGAIN-SVC and hybrid CNN-LSTM. The proposed technique achieves MOSs of 2.93, 3.35, and 3.21 for naturalness, melodic similarity, and phoneme intelligibility respectively.

Although the proposed hybrid CNN-LSTM model aims to balance content preservation with speaker identity transformation, achieving the perfect trade-off remains challenging. Formulating SVC as a multi-objective optimization problem or introducing a latent space and adjusting specific dimensions within this space allows users to control the balance between similarity and quality. One-shot methods are more susceptible to background noise, accent variations, and emotional fluctuations. Robustness to such factors is an ongoing research area. These modifications represent exciting avenues for future research.

Author contributions

Assila Yousuf: Resources, Investigation, Validation, Formal Analysis, Writing – original draft; David Solomon George: Supervision; All authors: Conceptualization, Methodology. All authors have read and approved the final version of the manuscript for publication.

Acknowledgments

The authors would like to thank Directorate of Technical Education (DTE) and A.P.J Abdul Kalam Technological University, Kerala, India for the support provided.

Conflict of interest

All authors declare no conflicts of interest in this paper

References

1. Helander E, Virtanen T, Nurminen J, Gabbouj M (2010) Voice conversion using partial least squares regression. *IEEE/ACM Transactions on Audio, Speech and Language Processing* 18: 912–921. https://doi.org/10.1109/TASL.2011.2165944

2. Saito Y, Takamichi S, Saruwatari H (2017) Voice conversion using input-to-output highway networks. *IEICE T Inf Syst* 100: 1925–1928. https://doi.org/10.1587/transinf.2017EDL8034

3. Yeh CC, Hsu PC, Chou JC, Lee HY, Lee LS (2018) Rhythm Flexible Voice Conversion Without Parallel Data Using Cycle-GAN Over Phoneme Posteriorgram Sequences. *IEEE Spoken Language Technology Workshop (SLT)* 274–281. https://doi.org/10.1109/SLT.2018.8639647

4. Sun L, Wang H, Kang S, Li K, Meng HM (2016) Personalized Cross-Lingual TTS Using Phonetic Posteriorgrams. *Interspeech* 322–326. https://doi.org/10.21437/Interspeech.2016-1043

5. Tian X, Chng ES, Li H (2019) A Speaker-Dependent WaveNet for Voice Conversion with Non-Parallel Data. *Interspeech* 201–205. https://doi.org/10.21437/Interspeech.2019-1514

6. Takahashi N, Singh MK, Mitsufuji Y (2023) Robust One-Shot Singing Voice Conversion. *arXiv:2210.11096v2*. https://doi.org/10.48550/arXiv.2210.11096

7. Hono Y, Hashimoto K, Oura K, Nankaku Y, Tokuda K (2019) Singing Voice Synthesis Based on Generative Adversarial Networks. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 6955–6959. https://doi.org/10.1109/ICASSP.2019.8683154

8. Sun L, Kang S, Li K, Meng H (2015) Voice conversion using deep bidirectional long short-term memory based recurrent neural networks. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 4869–4873. https://doi.org/10.1109/ICASSP.2015.7178896

9. Kaneko T, Kameoka H, Hiramatsu K, Kashino K (2017) Sequence-to-Sequence Voice Conversion with Similarity Metric Learned Using Generative Adversarial Networks. *Interspeech* 2017: 1283–1287. http://dx.doi.org/10.21437/Interspeech.2017-970

10. Freixes M, Alías F, Carrie JC (2019) A unit selection text-to-speech-and-singing synthesis framework from neutral speech: proof of concept. *EURASIP Journal on Audio, Speech, and Music Processing* 2019: 1–14. https://doi.org/10.1186/s13636-019-0163-y

11. Hono Y, Hashimoto K, Oura K, Nankaku Y, Tokuda K (2021) Sinsy: a deep neural network-based singing voice synthesis system. *IEEE/ACM T Audio Spe* 29: 2803–2815. https://doi.org/10.1109/TASLP.2021.3104165

12. Sisman B, Vijayan K, Dong M, Li H (2019) SINGAN: Singing Voice Conversion with Generative Adversarial Networks. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)* 112–118. https://doi.org/10.1109/APSIPAASC47483.2019.9023162

13. Sisman B, Li H (2020) Generative adversarial networks for singing voice conversion with and without parallel data. *Odyssey* 238–244. https://doi.org/10.21437/Odyssey.2020-34

14. Zhao W, Wang W, Sun Y, Tang T (2019) Singing voice conversion based on wd-gan algorithm. *IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)* 950–954. https://doi.org/10.1109/IAEAC47372.2019.8997824

15. Fang F, Yamagishi J, Echizen I, Lorenzo-Trueba J (2018) High-Quality Nonparallel Voice Conversion Based on Cycle-Consistent Adversarial Network. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 5279–5283. https://doi.org/10.1109/ICASSP.2018.8462342

16. Kameoka H, Kaneko T, Tanaka K, Hojo N (2018) StarGAN-VC: non-parallel many-to-many Voice Conversion Using Star Generative Adversarial Networks. *IEEE Spoken Language Technology Workshop (SLT)* 266–273. https://doi.org/10.1109/SLT.2018.8639535

17. Chen Y, Xia R, Yang K, Zou K (2023) MICU: Image Super-resolution via Multi-level Information Compensation and U-net. *Expert Syst Appl* 245: 123111. https://doi.org/10.1016/j.eswa.2023.123111

18. Chen Y, Xia R, Yang K, Zou K (2023) MFMAM: Image Inpainting via Multi-Scale Feature Module with Attention Module. *Comput Vis Image Und* 238: 103883. https://doi.org/10.1016/j.cviu.2023.103883

19. Chen Y, Xia R, Yang K, Zou K (2023) GCAM: Lightweight Image Inpainting via Group Convolution and Attention Mechanism. *Int J Mach Learn Cyb* 15: 1815–1825. https://doi.org/10.1007/s13042-023-01999-z

20. Chen Y, Xia R, Yang K, Zou K (2024) DNNAM: Image Inpainting Algorithm via Deep Neural Networks and Attention Mechanism. *Appl Soft Comput* 111392. https://doi.org/10.1016/j.asoc.2024.111392

21. Chen Y, Xia R, Yang K, Zou K (2023) DARGS: Image Inpainting Algorithm via Deep Attention Residuals Group and Semantics. *J King Saud Univ-Comput* 35: 101567. https://doi.org/10.1016/j.jksuci.2023.101567

22. Chen L, Zhang X, Li Y, Sun M, Chen W (2024) A Noise-Robust Voice Conversion Method with Controllable Background Sounds. *Complex Intell Syst* 1–14. https://doi.org/10.1007/s40747-024-01375-6

23. Walczyna T, Piotrowski Z (2023) Overview of Voice Conversion Methods Based on Deep Learning. *Applied sciences* 13: 3100. https://doi.org/10.3390/app13053100

24. Liu EM, Yeh JW, Lu JH, Liu YW (2023) Speaker Embedding Space Cosine Similarity Comparisons of Singing Voice Conversion. *The Journal of the Acoustical Society of America (JASA)* 154: A244–A244. https://doi.org/10.1121/10.0023424

25. Hsu CC, Hwang HT, Wu YC, Tsao Y, Wang HM (2016) Voice conversion from non-parallel corpora using variational auto-encoder. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)* 1–6. https://doi.org/10.1109/APSIPA.2016.7820786

26. Tobing PL, Wu YC, Hayashi T, Kobayashi K, Toda T (2019) Non-Parallel Voice Conversion with Cyclic Variational Autoencoder, *Interspeech* 674–678. https://doi.org/10.21437/Interspeech.2019-2307

27. Yook D, Leem SG, Lee K, Yoo IC (2020) Many- to-many voice conversion using cycle-consistent variational autoencoder with multiple decoders. *Odyssey* 215–221. https://doi.org/10.21437/Odyssey.2020-31

28. Hsu CC, Hwang HT, Wu YC, Tsao Y, Wang HM (2017) Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks. *arXiv preprint arXiv:1704.00849*. https://doi.org/10.48550/arXiv.1704.0084

29. Huang WC, Violeta LP, Liu S, Shi J, Toda T (2023) The Singing Voice Conversion Challenge 2023. *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)* 1–8. https://doi.org/10.1109/ASRU57964.2023.10389671

30. Chen Q, Tan M, Qi Y, Zhou J, Li Y, Wu Q (2022) V2C: Visual Voice Cloning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 21242–21251.

31. Qian K, Zhang Y, Chang S, Yang X, Hasegawa-Johnson M (2019) Autovc: Zero-shot voice style transfer with only autoencoder loss. *International Conference on Machine Learning* 5210–5219.

32. Patel M, Purohit M, Parmar M, Shah NJ, Patil HA (2020) Adagan: Adaptive gan for many-to-many non-parallel voice conversion.

33. Liu F, Wang H, Peng R, Zheng C, Li X (2021) U2-VC: one-shot voice conversion using two-level nested U-structure. *EURASIP Journal on Audio, Speech, and Music Processing* 2021: 1–15. https://doi.org/10.1186/s13636-021-00226-3

34. Liu F, Wang H, Ke Y, Zheng C (2022) One-shot voice conversion using a combination of U2-Net and vector quantization. *Appl Acoust* 199: 109014. https://doi.org/10.1016/j.apacoust.2022.109014

35. Wu DY, Lee HY (2020) One-shot voice conversion by vector quantization. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 7734–7738. https://doi.org/10.1109/ICASSP40776.2020.9053854

36. Chou JC, Lee HY (2019) One-Shot Voice Conversion by Separating Speaker and Content Representations with Instance Normalization. *Interspeech* 664–668. https://doi.org/10.21437/Interspeech.2019-2663

37. Huang X, Belongie S (2017) Arbitrary style transfer in real-time with adaptive instance normalization. *IEEE International Conference on Computer Vision (ICCV)* 1501–1510. https://doi.org/10.1109/ICCV.2017.167

38. Lian J, Lin P, Dai Y, Li G (2022) Arbitrary Voice Conversion via Adversarial Learning and Cycle Consistency Loss. *International Conference on Intelligent Computing* 569–578. https://doi.org/10.1007/978-3-031-13829-4_49

39. Gu Y, Zhao X, Yi X, Xiao J (2022) Voice Conversion Using learnable Similarity-Guided Masked Autoencoder. *International Workshop on Digital watermarking* 13825: 53–67. https://doi.org/10.1007/978-3-031-25115-3_4

40. Chen YH, Wu DY, Wu TH, Lee HY (2021) AGAIN-VC: A one-shot voice conversion using activation guidance and adaptive instance normalization. *IEEE International Conference on Acoustics, Speech, and Signal Processing* 5954–5958. https://doi.org/10.1109/ICASSP39728.2021.9414257

41. Ulyanov D, Lebedev V, Vedaldi A, Lempitsky VS (2016) Texture networks: Feed-forward synthesis of textures and stylized images. *Proceedings of the 33nd International Conference on Machine Learning* 1349–1357.

42. Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proceedings of the 32nd International Conference on International Conference on Machine Learning* 37: 448–456.

43. Li Y, Wang N, Shi J, Liu J, Hou X (2016) Revisiting batch normalization for practical domain adaptation. *arXiv preprint arXiv:1603.04779*.

44. Ulyanov D, Vedaldi A, Lempitsky V (2017) Improved Texture Networks: Maximizing Quality and Diversity in Feed-Forward Stylization and Texture Synthesis. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 4105–4113. https://doi.org/10.1109/CVPR.2017.437

45. Liu J, Han W, Ruan H, Chen X, Jiang D, Li H (2018) Learning Salient Features for Speech Emotion Recognition Using CNN. *First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)* 1–5. https://doi.org/10.1109/ACIIAsia.2018.8470393

46. Lim W, Jang D, Lee T (2016) Speech emotion recognition using convolutional and Recurrent Neural Networks. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)* 1–4. https://doi.org/10.1109/APSIPA.2016.7820699

47. Hajarolasvadi N, Demirel H (2019) 3D CNN-Based Speech Emotion Recognition Using K-Means Clustering and Spectrograms. *Entropy (Basel)* 21: 479. https://doi.org/10.3390/e21050479

48. Graves A (2012) Long Short-Term Memory Supervised Sequence Labelling with Recurrent Neural Networks. *Studies in Computational Intelligence* 385: 37–45. https://doi.org/10.1007/978-3-642-24797-2

49. Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

50. Kumar K, Kumar R, de Boissiere T, Gestin L, Teoh WZ, Sotelo J, et al. (2019) Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in Neural Information Processing Systems* 14910–14921.

51. Kong J, Kim J, Bae J (2020) HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. *Proceedings of the 34th International Conference on Neural Information Processing Systems* 33: 17022–17033.

52. Duan Z, Fang H, Li B, Sim KC, Wang Y (2013) The NUS sung and spoken lyrics corpus: A quantitative comparison of singing and speech. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference* 1–9. https://doi.org/10.1109/APSIPA.2013.6694316

53. Kubichek R (1993) Mel-cepstral distance measure for objective speech quality assessment. *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing* 1: 125–128. https://doi.org/10.1109/PACRIM.1993.407206

54. Kobayashi K, Toda T, Nakamura S (2018) Intra-gender statistical singing voice conversion with direct waveform modification using log spectral differential. *Speech Commun* 99: 211–220. https://doi.org/10.1016/j.specom.2018.03.011

55. Toda T, Tokuda K (2007) A speech parameter generation algorithm considering global variance for hmm-based speech synthesis. *IEICE T Inf Syst* 90: 816–824. https://doi.org/10.1093/ietisy/e90-d.5.816