ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10095654

VQ-CL: LEARNING DISENTANGLED SPEECH REPRESENTATIONS WITH CONTRASTIVE LEARNING AND VECTOR QUANTIZATION.

Huaizhen Tang[1,2], *Xulong Zhang*[1], *Jianzong Wang*[1*], *Ning Cheng*[1], *Jing Xiao*[1*]

[1]Ping An Technology (Shenzhen) Co., Ltd.
[2]University of Science and Technology of China

ABSTRACT

Voice Conversion(VC) refers to converting the voice characteristics of audio to another one as it is said by other people. Recently, more and more studies have focused on disentangle-based VC, which separates the timbre and linguistic content information from an audio signal to effectively achieve VC tasks. However, It's still challenging to extract phoneme-level features from frame-level hidden representations. This paper proposed a novel zero-shot voice conversion framework that utilizes contrastive learning and vector quantization to encourage the frame-level hidden features closer to the phoneme-level linguistic information, called **VQ-CL**. All objective and subjective experiment results show that VQ-CL has better performance than previous studies in separating content and voice characteristics to improve the sound quality of generated speech.

Index Terms— speech synthesis, contrastive learning, voice conversion, vector quantization

1. INTRODUCTION

Voice conversion (VC) also called Voice Style Transfer, which converting an utterance of a source speaker to another utterance of a target person by keeping the content information of the original speech but replacing it with the vocal features from the target speaker. Recently, considerable effort have spent on the topic of VC [1, 2, 3]. These methods can be roughly divided into two categories: parallel VC systems and Non-parallel VC systems [4]. Since it's hard for the parallel VC systems to produce natural speech for a target speaker without enough pair source-target data. Recently, more and more attention have been focused on the non-parallel VC systems.

Recently, many studies have reported that non-parallel VC tasks can be effectively achieved by speech representation disentanglement [5, 6, 7, 8]. Specifically, for each speech, which contains both linguistic information, which we called the content information, and some speaker related information, which we called style information. Obviously, it will be very convenient and flexible for style transfer if we can separate the content and style information and characterize them well. However, learning disentangled speech representations is important but still challenging. The causes of this phenomenon are extremely complex, and we think one of the most critical reasons is that eliminating all style information from the source speech is very challenging.

In recent years, various voice conversion solutions are proposed trying to learn the ideal content features from the input speech. For instance, INVC [9] utlizied Instance Normalization to eliminate the static global information to learn disentangled content features. At the same time, VQ-based models [10, 11] discard the time-invariant information by replacing the hidden frame-level features with the closest discrete coding. Besides, Qian *et,al* proposed AutoVC [12] to get disentangled content embeddings by applying a properly tuned information constraining bottleneck.

Unfortunately, all the above models have their questions. For example, INVC applied a vanilla autoencoder to force the hidden space to approach the Gaussian distribution, which makes it unable to guarantee the distribution-matching between the generated and input data. AutoVC claims they can achieve distribution-matching VC, but they ignored the diversity of the phoneme durations. In addition, in order to disentangle the timbre information, AutoVC introduces very harsh limitations on the dimension of the hidden representations, this would compromise the quality of the converted speech [7, 13]. Relatively speaking, some VQ-based models like [14, 15, 16] can learn well content embeddings from input audio signals. However, in order to improve the disentanglement effectiveness, these methods need to introduce other network structures to combine with VQ, which leads the model very complex.

Recently, the emergence of the paper [17] has given us great inspiration. Like [18], by applying the Montreal forced aligner tool [MFA], we can get the alignment between the phoneme sequences and speech sequences. To take advantage of contrastive learning and this tool. We further proposed a new method to guide the content and style disentanglement with the variance phoneme duration.

In this paper, we proposed a novel voice conversion framework that can be used to learn disentangled speech

[*] These authors are co-corresponding authors: Jianzong Wang, (jzwang@188.com). Jing Xiao, (xiaojing661@pingan.com.cn).

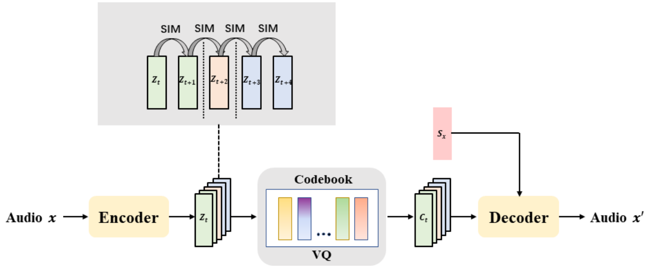


Fig. 1. Framework of VQ-CL. $\boldsymbol{Z_x}$ refer to the frame-level hidden features, $\boldsymbol{C_x}$ denotes the discrete code which is produced by *codebook*. $\boldsymbol{S}_x$ is the style embedding, and it is gengrated from the pretrained style encoder.

representation by applying contrastive learning and vector quantization, named VQ-CL. Specifically, we design a new training method to encourage those frame-level features that correspond to the same phoneme as close, and at the same time, make those frame-level features correspond to different phonemes as far as possible. Experiment results are carried out on AISHELL-3 datasets. Our main contributions can be shown as follows:

- We applied a new training method to guide the content embeddings to contain more pure linguistic information while the style information would be encouraged to be discarded from the encoder output;
- Vector Quantization is also utilized to further eliminate the style information so that we can get the ideal content embedding.

2. METHOD

In this section, we will introduce the core idea of the proposed method. As shown in Figure 1. Our model contains a feature encoder E and a decoder D. In training, an audio x was randomly selected as the input, and the feature encoder will output the hidden frame-level features Z_x from x. After that, the decoder will reconstruct x based on the phoneme-level features C_x and the style embeddings S_x. Noted that S_x is ready-made given by a pretrained style encoder training with GE2Eloss [19], which maximizes the embedding similarity among some utterances which from the same speaker, and minimizes the similarity among different speakers. Now, what we need to do is to find a way to construct the mapping relationship between the frame-level features Z_x and the phoneme-level features C_x.

2.1. Contrast Similarity

First, given a frame-level audio sequence $X = (x_1, x_2, ..., x_T)$. Where T is the length of X. The encoder E can learn a frame-level hidden representations from X, that is, $Z_X = (z_1, z_2, ..., z_T) = E(X)$. And, For each frame hidden feature $z_t \in Z_X$, since we have known the alignment between the phoneme and speech by utilizing MFA Tools, we can easily know whether z_{t+1} contains the same content information with z_t or not. As shown in the upper left of Figure 1, the same colors mean that these hidden features correspond to the same phoneme, and different colors indicate that they correspond to different phonemes. In other words, we can get the boundary frame of two adjacent phonemes.

Obviously, if z_t is not the boundary frame, z_t and z_{t+1} correspond to the same phoneme, we think they contain the same content information, then we expect they should be as closer as possible. At the same time, for those boundary frames between adjacent phonemes, the hidden features z_t and z_{t+1} correspond to different phonemes, we expect they should be as different as possible. In this paper, we use cosine similarity score to evaluate the similarity between a pair of features. It can be formulated as:

$$G(A(x), A(x_1)) = \frac{A^T(x)A(x_1)}{\|A(x)\|_2\|A(x_1)\|_2} \tag{1}$$

Where $G(\cdot, \cdot)$ means the cosine similarity score. $A(\cdot)$ can be any hidden representations extracted from input speech.

As we said above, we train the model to promote a high cosine similarity between similar hidden representations, and a low cosine similarity between the boundary hidden features. Hence, we proposed the similarity contrast loss function to train the model and it can be computed as:

$$\mathcal{L}_{\text{sim}} = \frac{\sum_{t=boundary}(1 + G(E(x_t), E(x_{t+1})))}{\sum_{t\neq boundary}(1 + G(E(x_t), E(x_{t+1})))} \tag{2}$$

Where $E(\cdot)$ means the processing of the feature encoder. Noted that the value of the cosine similarity score $G(\cdot, \cdot) \in (-1, 1)$, so we add a constant 1 to make the training more stable. When $t = boundary$, which indicates that z_t and z_{t+1} contain different content information, otherwise indicates that z_t and z_{t+1} correspond the same phoneme. During training, to minimize the contrast loss $\mathcal{L}_{\text{sim}}$, the feature encoder will be optimized to generate the frame-level hidden features more related to the linguistic information.

2.2. Vector Quantization

Vector Quantization(VQ) is an effective data compression technology that can quantify continuous data into the closest discrete data. Specifically, if we define V as a sequence of continuous data, that is, $\boldsymbol{V} = v_1, v_2, ...v_T$. Then $VQ(\boldsymbol{V})$ can be defined as $VQ(\boldsymbol{V}) = q_1, q_2, ...q_T$. That is, for each data $v_i \in \boldsymbol{V}$, the cloest discrete data can be computed as:

$$VQ(v_i) = q_i; \qquad q_i = \arg\min_{q\in Codebook}(\|v_i - q\|_2^2) \tag{3}$$

Recently, many VQ-based models have been proposed to learn discrete speech representations, and previous studies have reported that the quantized discrete data from the utterance is closely related to the phoneme information[20, 21]. Inspired by it, we can use VQ to further eliminate the remaining global style information so that we can get the ideal content embeddings. In training phase, the constrain of the latent-code loss function will minimize the distance between the discrete code and the continuous embeddings and it can be expressed as:

$$\mathcal{L}_{\text{latent}} = \|Z_X - C_X\|_2^2 \tag{4}$$

It's worth mentioning that, unlike those VQ-based models, since the encoder output Z_X are encouraged to be more related with the linguistic information, we can not get the style embeddings from the mean difference between Z_X and the discrete content embeddings C_X. That's why we provided a ready-made style embeddings S_X.

Now, with the content embeddings C_X and the style embeddings S_X, the decoder are encouraged to generate the reconstructed speech x'. We use the reconstruction loss function to constrain this process and it can be formulated as:

$$\mathcal{L}_{\text{recon}} = \|x_i' - x_i\|_1^1 \qquad x_i' = D(c_{x_i}, s_{x_i}) \tag{5}$$

Then the full loss function can be expressed as:

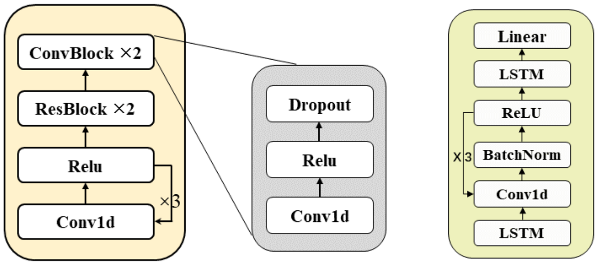
$$L(\boldsymbol{\theta_e}, \boldsymbol{\theta_d}) = \mathcal{L}_{\text{recon}} + \alpha\mathcal{L}_{\text{sim}} + \beta\mathcal{L}_{\text{latent}} \tag{6}$$

Where θ_e and θ_d are the parameters of the feature encoder and the decoder, respectively. α , β refers to the weight of $\mathcal{L}_{\text{sim}}$ and $\mathcal{L}_{\text{latent}}$, respectively.

In the inference phase, with the trained VQ-CL, we can easily get the content embeddings of the source speech and the style embeddings of the tatget speaker, then the decoder will generate a well converted speech.

2.3. Architecture of the proposed method

The architecture of VQ-CL is shown in Figure 2. As illustrated in Figure 2(a), we designed the feature encoder of VQ-CL with three convolution layers, two Conv1dResBlocks, and two ConvBlocks. Among them, Conv1dResBlock refers to the convolution layers with the residual-connect network, and the lower right region of Figure 2(a) shows the details of the ConvBlock. Noted that we introduce dropout to randomly drop the features of some channels, which can be used to encourage the feature encoder more focus on the linguistic-related information and ignore the global style information so that the frame-level hidden features Z_X would be more related to the phoneme information. In addition, as shown in Figure 2(b), we introduce the decoder of AutoVC as our decoder. In training, the style embedding is first copied to the same length as the content embeddings and then concatenated in the channel dimension. The concatenated embedding is passed into the decoder to generate the reconstructed speech.



(a) Feature encoder (b) Decoder

Fig. 2. Architecture of VQ-CL. Noted that **×2** and **×3** denote the number of Resblock, Convblock and Conv1d layers in the feature encoder and the decoder.

3. EXPERIMENTS

3.1. Experiment Configurations

Objective and subjective experiments were conducted on AISHELL-3, a high-fidelity multi-speaker Mandarin speech corpus, to assess the performance of the proposed model in many-to-many VC and Zero-shot VC tasks. The corpus comprises 88035 recordings, totaling roughly 85 hours of speech, from 218 native Chinese Mandarin speakers. The corpus also includes hand-labeled full pinyin annotations. The entire dataset was randomly divided into three sets: 63262 recordings from 174 speakers for training, and the remaining recordings from these speakers were used for testing. Additionally, the voice of some speakers who did not appear in the training set was used for conducting zero-shot VC experiments. In this study, all recordings were resampled at 22.05kHz, and mel-spectrograms were computed using a short-time Fourier transform (STFT) with Hann windowing. The STFT used 1024 for FFT size, 1024 for window size, and 256 for hop size. Furthermore, the STFT magnitude was transformed to the mel scale using an 80 channel mel filter bank spanning 90 Hz to 10.6 kHz.

The VQ-CL model was trained on one NVIDIA V100 GPU for 300k steps, using a batch size of sixteen. A codebook size of 256 was chosen, and the speaker embedding was generated by averaging the embeddings from 10 two-second utterances of the same speaker to the pretrained speaker encoder. The weights in Eq.(6) were set to $\alpha = 0.01, \beta = 0.1$. We used F0-AutoVC, TGAVC, and VQVC+ models as the baseline models, with training following the description in [22, 7, 11], to ensure a fair comparison. Moreover, we used a pretrained Hifi-gan [23] vocoder to convert all the output mel-spectrograms back to the waveform.

To measure the quality of converted speech in our study, we used the Mel-Cepstral Distortion (MCD) metric as an objective evaluation. We also conducted subjective evaluations using MOS and VSS tests with 13 human participants, comprising nine males and four females. For the MOS test, participants rated the naturalness of each converted speech on a

Table 1. Comparison of different models in traditional VC and zero-shot vc

Methods	**Many-to-Many VC**			**zero-shot VC**		
	MCD	MOS	VSS	MCD	MOS	VSS
F0-AutoVC	6.86 ± 0.42	3.50 ± 1.11	3.08 ± 1.29	7.41 ± 0.53	3.35 ± 1.20	3.01 ± 1.33
TGAVC	7.08 ± 0.31	3.66 ± 1.27	3.14 ± 1.08	7.65 ± 0.31	3.54 ± 1.23	3.06 ± 0.97
VQVC+	8.94 ± 0.25	3.23 ± 1.42	3.05 ± 0.84	9.21 ± 0.46	3.02 ± 1.21	3.26 ± 1.17
VQ-CL	**6.23 ± 0.37**	**3.88 ± 1.10**	**3.31 ± 0.97**	**6.38 ± 0.48**	**3.59 ± 136**	**3.26 ± 1.36**

scale of 1-5, with higher scores indicating better quality. In the VSS test, participants rated groups of utterances with a score of 1-5 based on the perceived voice similarity between the converted speech and the ground truth speech. We calculated the score based on the timbre similarity perceived by the participants. A similarity score of 5 indicates that the converted speech is most similar to the ground truth speech, while a score of 1 indicates that the participant does not believe the two speeches come from the same speaker. The results of the evaluations are presented in Table 1.

The results indicate that our VQ-CL model outperforms other baseline models in both subjective and objective tests in traditional many-to-many VC tasks. This suggests that the speech generated by our model is superior to that of the baseline models. Furthermore, the VSS test reveals that our method improves the conversion effect by enabling better learning of timbre and prosodic features compared to F0-AutoVC, TGAVC, and VQVC+. The results of natural evaluation demonstrate that our proposed method is still superior to the baselines, even for unseen speakers. Additionally, many people find that the speech synthesized using our method is more similar to the ground truth than the baseline synthesized speech, highlighting the competence of VQ-CL in zero-shot conversion, as shown in the right part of Table 1.

3.2. Ablation studies

In this section, we first evaluate the effectiveness of the similarity contrast loss $\mathcal{L}_{sim}$ and VQ technology, respectively. Specifically, we retrain the proposed model without $\mathcal{L}_{sim}$, called 'M1', or without VQ, called 'M2'. To evaluate the quality of different converted speech more accurately, in addition to the above MCD test, we add another objective experiment. To compare the similarity of 8 unknown speeches (5 real ones and 3 fakes generated from M1, M2, and VQ-CL), we conducted a fake speech detection test using the Resemblyzer open-source speech detection toolkit (`https://github.com/resemble-ai/Resemblyzer`) against ground truth reference audio. For each converted speech, The toolkit will automatically give a Detection score between 0 to 1 against the ground truth reference audio. The higher the score, the more similar the converted speech and the target voice. We repeated the test on 20 groups of converted speeches and the results are shown in Table 2. In conclusion, with the similarity contrast loss and VQ technology, the performance of the model will be greatly improved.

Besides, we also focused on the choice of codebook sizes, that is, how many discrete vectors the codebook contains are the best. For this, we reuse *Resemblyzer* to conduct many comparison experiments to find the proper number of codebook sizes. The results are summarized in Table 3, and you can find one of the Comparison results in Figure 3. From the result, 256 is selected as an appropriate number for the codebook sizes.

Table 2. Results of the ablation experiments.

Score	VQ-CL	M1	M2
MCD	6.23 ± 0.37	7.07 ± 0.41	6.49 ± 0.19
Detection	0.77 ± 0.16	0.66 ± 0.31	0.72 ± 0.09

Table 3. Comparison of different codebook sizes .

Codebook Size	64	128	256	512
Detection Mean Score	0.663	0.731	0.760	0.738

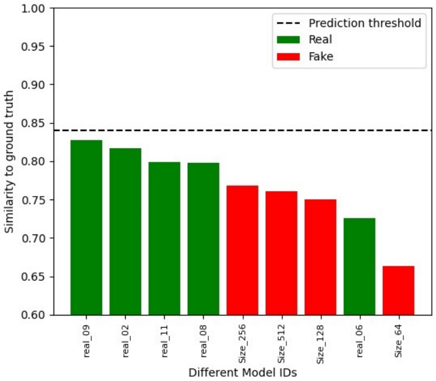


Fig. 3. Detection Score under different codebook sizes.

4. CONCLUSION

In this paper, we proposed the similarity contrast loss to improve the disentanglement between the content and style information. At the same time, VQ is also applied to further eliminate the residual style information to improve the performance of VQ-CL in VC tasks. All objective and subjective experiments show that VQ-CL overperformed previous works in VC tasks.

5. ACKNOWLEDGEMENT

This paper is supported by the Key Research and Development Program of Guangdong Province under grant No.2021B-0101400003. Co-corresponding authors are Jianzong Wang (jzwang@188.com) and Jing Xiao (xiaojing661@pingan.com.cn).

6. REFERENCES

[1] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, “Stargan-vc2: Rethinking conditional methods for stargan-based voice conversion,” *Proc. Interspeech 2019*, pp. 679–683, 2019.

[2] Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min Wang, “Voice conversion from non-parallel corpora using variational auto-encoder,” in *2016 APSIPA*. IEEE, 2016, pp. 1–6.

[3] Kaizhi Qian, Yang Zhang, Shiyu Chang, Jinjun Xiong, Chuang Gan, David D. Cox, and Mark Hasegawa-Johnson, “Global rhythm style transfer without text transcriptions,” *CoRR*, vol. abs/2106.08519, 2021.

[4] Yannis Stylianou, Olivier Cappé, and Eric Moulines, “Continuous probabilistic transform for voice conversion,” *IEEE Transactions on speech and audio processing*, vol. 6, no. 2, pp. 131–142, 1998.

[5] Kaizhi Qian, Yang Zhang, Shiyu Chang, Mark Hasegawa-Johnson, and David Cox, “Unsupervised speech decomposition via triple information bottleneck,” in *ICML*. PMLR, 2020, pp. 7836–7846.

[6] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed, “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[7] Huaizhen Tang, Xulong Zhang, Jianzong Wang, Ning Cheng, Zhen Zeng, Edward Xiao, and Jing Xiao, “Tgavc: Improving autoencoder voice conversion with text-guided and adversarial training,” in *ASRU*. IEEE, 2021, pp. 938–945.

[8] Huaizhen Tang, Xulong Zhang, Jianzong Wang, Ning Cheng, and Jing Xiao, “Avqvc: One-shot voice conversion by vector quantization with applying contrastive learning,” in *ICASSP*. IEEE, 2022, pp. 4613–4617.

[9] Ju-Chieh Chou and Hung-yi Lee, “One-shot voice conversion by separating speaker and content representations with instance normalization,” in *Interspeech 2019*. 2019, pp. 664–668, ISCA.

[10] Da-Yi Wu and Hung-yi Lee, “One-shot voice conversion by vector quantization,” in *ICASSP*. IEEE, 2020, pp. 7734–7738.

[11] Da-Yi Wu, Yen-Hao Chen, and Hung-yi Lee, “Vqvc+: One-shot voice conversion by vector quantization and u-net architecture,” *Proc. Interspeech 2020*, pp. 4691–4695, 2020.

[12] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson, “Autovc: Zero-shot voice style transfer with only autoencoder loss,” in *ICML*. PMLR, 2019, pp. 5210–5219.

[13] Zhiyuan Tan, Jianguo Wei, Junhai Xu, Yuqing He, and Wenhuan Lu, “Zero-shot voice conversion with adjusted speaker embeddings and simple acoustic features,” in *ICASSP 2021*. 2021, pp. 5964–5968, IEEE.

[14] Benjamin van Niekerk, Leanne Nortje, and Herman Kamper, “Vector-quantized neural networks for acoustic unit discovery in the zerospeech 2020 challenge,” in *Interspeech 2020*. 2020, pp. 4836–4840, ISCA.

[15] Shaojin Ding and Ricardo Gutierrez-Osuna, “Group latent embedding for vector quantized variational autoencoder in non-parallel voice conversion.,” in *Interspeech*, 2019, pp. 724–728.

[16] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *NeurIPs*, vol. 33, pp. 12449–12460, 2020.

[17] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola, “What makes for good views for contrastive learning?,” *NeurIPs*, vol. 33, pp. 6827–6839, 2020.

[18] Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu, “Fastspeech 2: Fast and high-quality end-to-end text to speech,” *arXiv preprint arXiv:2006.04558*, 2020.

[19] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, “Generalized end-to-end loss for speaker verification,” in *2018 ICASSP*. IEEE, 2018, pp. 4879–4883.

[20] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu, “Neural discrete representation learning,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 6309–6318.

[21] Alexander H Liu, Tao Tu, Hung-yi Lee, and Lin-shan Lee, “Towards unsupervised speech recognition and synthesis with quantized speech representation learning,” in *ICASSP*. IEEE, 2020, pp. 7259–7263.

[22] Kaizhi Qian, Zeyu Jin, Mark Hasegawa-Johnson, and Gautham J Mysore, “F0-consistent many-to-many non-parallel voice conversion via conditional autoencoder,” in *ICASSP*. IEEE, 2020, pp. 6284–6288.

[23] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis,” *NeurIPs*, vol. 33, pp. 17022–17033, 2020.