Complex & Intelligent Systems (2022) 8:1539–1550
https://doi.org/10.1007/s40747-021-00604-6

ORIGINAL ARTICLE

Voice adaptation by color-encoded frame matching as a multi-objective optimization problem for future games

Mads Midtlyng[1] · Yuji Sato[2] · Hiroshi Hosobe[2]

Received: 17 April 2021 / Accepted: 26 November 2021 / Published online: 4 January 2022

Abstract

Voice adaptation is an interactive speech processing technique that allows the speaker to transmit with a chosen target voice. We propose a novel method that is intended for dynamic scenarios, such as online video games, where the source speaker's and target speaker's data are nonaligned. This would yield massive improvements to immersion and experience by fully becoming a character, and address privacy concerns to protect against harassment by disguising the voice. With unaligned data, traditional methods, e.g., probabilistic models become inaccurate, while recent methods such as deep neural networks (DNN) require too substantial preparation work. Common methods require multiple subjects to be trained in parallel, which constraints practicality in productive environments. Our proposal trains a subject nonparallel into a voice profile used against any unknown source speaker. Prosodic data such as pitch, power and temporal structure are encoded into RGBA-colored frames used in a multi-objective optimization problem to adjust interrelated features based on color likeness. Finally, frames are smoothed and adjusted before output. The method was evaluated using Mean Opinion Score, ABX, MUSHRA, Single Ease Questions and performance benchmarks using two voice profiles of varying sizes and lastly discussion regarding game implementation. Results show improved adaptation quality, especially in a larger voice profile, and audience is positive about using such technology in future games.

Keywords Voice adaptation · Speech processing · Color-encoding · Multi-objective optimization problems · Video games

Introduction

Voice adaptation (VA) is the speech processing technique [1–5] of translating a spoken message from a source speaker into the voice of a target speaker while retaining prosodic features. Similar, but varying approaches exist as voice conversion and voice transformation. Prosodic information can be divided into many variables, such as the pitch of the voice, loudness, voice quality and more, giving our speech emotion and variance. This process allows a user to communicate with their own voice into a recording device and have it outputted as the voice of another subject. This has rarely seen commercial use due to the difficulty regarding speech quality and processing time, but areas such as online video games can clearly benefit from such a technology if it were to prove performant enough. In online video games, it is elemental for the user to create an alias for their name and an avatar to represent their visual appearance; however, the voice remains unchanged. Benefits include increase of immersion, protection of privacy, character imitation, not to mention how efficient voice-work would be for game studios. It may give players a sense of safety by hiding behind a different voice, while at the same time feeling, they belong in the game world with more credibility, thus providing more enjoyment. VA is often divided into two sub-types, namely, parallel and nonparallel, where the former has been favored and trains two or more subjects correspondingly using similar training parameters and rules for mapping speech characteristics and sometimes physical traits, such as age, gender, models of vocal tracts and speech patterns depending on language. This allows for direct mapping of speech features and

✉ Mads Midtlyng
midtlyng.madsalexander.9c@stu.hosei.ac.jp

Yuji Sato
yuji@k.hosei.ac.jp

Hiroshi Hosobe
hosobe@acm.org

[1] Department of Computer Science, Hosei University, Tokyo, Japan

[2] Faculty of Computer and Information Sciences, Hosei University, Tokyo, Japan

envelope attributes from person $A \rightarrow B$; however, it severely limits the flexibility in using the system without supplemental training and speakers. Nonparallel, however, trains a single subject into a mappable set of data that can be looked up against an unrelated speaker despite varying corpora. This has seen some use in the past by construction of pseudo data sets for pairs of source and target speakers, or transformation of utterings by utilizing existing parallel data sets with separate utterances that are paired by estimation models, or finally by estimating phonemic content correspondingly per active speaker. The crucial factor for nonparallel approaches is the trained model's ability to account for unseen corpora, i.e., speech factors not included in the training material or latent space. One method to account for this is to adjust the scale of sampling so that the voice is broken down to a more rudimentary resolution or increase the training volume substantially to begin with. Models that require reconstruction stages also suffer from quality loss. Speech signals are non-stationary, so it is common to divide them into small frames, because temporal characteristics change quickly, making it difficult to analyze and compare larger segments as most of the analysis is generally based on the frequency domain. Extraction of temporal information like abrupt signal changes is better captured in the time domain, so there is a tradeoff between a better temporal resolution in the frequency domain and the number of samples in a single frame for the time domain. Background noise is also a challenge as it alters the relative energy density, hindering accurate prosodic representation. In this paper, related and contending methods are first presented, then the proposed method and its supporting methods are detailed as well as multi-objective optimization problems, and lastly evaluation and observations.

Related work

Classical methods

Previously considered state of the art approaches were in most cases based on probabilistic [3, 6, 7] as well as rule-based processes. Spectral conversion techniques have been tested for over a decade and to improve the spectral mapping, Gaussian Mixture Model (GMM) was often used which had varying results compared to rule-based approaches due to inconsistent quality. Some methods see the use of pitch shifting to simulate prosody [6, 8], and generally in the field, the adaptation quality was more often than not the ability to adapt some letters of an alphabet from many training data and very specific voice pairings.

Modern methods

Apart from statistical methods as seen in Wu et al. [9] as well as [10], they present an exemplar-based conversion technique with improved output voice quality equal to Maximum Likelihood GMM (ML, GMM). Most commonly used as in [11] is layer-based generative training by Deep Neural Network (DNN), although while having improved results over classical methods, the training procedure is complex and require multiple speakers for data pairing. In [12] we see an interactive evolution in the training stage which considers parameters, such as pitch, power and length which are then used in real-time conversion to simulate prosody. Their results show that evolutionary computation achieved better scores versus a human using trial-and-error learning. [13] performs mapping of the voice's spectral data subsequently stored in a codebook for lookup between speakers. For training they used two speakers from which they generate corresponding vectors using dynamic time warping [14] used VOCALOID's database of various synthetic voices and the creation of voice profile while considering physical parameters of the voice to convert timbre information using GMM. As DNN has become more accessible, more and more research attempt nonparallel VA despite its hurdle of matching unaligned frames, a major type being *CycleGAN* [15, 16] which performs automated training of various translation models, typically for imagery with no pairing examples, as well as general spectral conversion seen in [17]. International contests have been held to judge the state of the technology [18], but VA has yet to see commercial use due do its lack in performance and voice quality.

Another popular deep learning technique is variational auto-encoders, or VAE. Especially VAEB (Bayes) is a favored version that is used to learn a specific model p using an encoder q, parametrized by the use of a neural network to generate a directed latent-variable probabilistic model. The model p is parametrized according to (1), where the deterministic vector-valued functions $\vec{\mu}(z)$, $\vec{\sigma}(z)$ receive further parametrization by the neural network, typical with two dense hidden layers. The encoder q is similarly parametrized (2). In [19] they employ *WaveNet* Vocoder that trains a model with limited training data that estimate speech from a multi-speaker data set. It is claimed that a stable, but speaker-dependent vocoder model can be achieved with just 5 min of training data. While they see improved results, there are many cases, where it is similar to conventional vocoder methods. The conditional network (5 layers) output was repeated 80 times and the training went on for 440,000 steps. There is no mention of time spent training, but generally neural networks can take from hours to days to generate decent results, depending on the parameters on the current hardware:

$$p(x|z) = \mathcal{N}\left(x; \vec{\mu}(z), diag(\vec{\sigma}(z))^2\right) \qquad (1)$$

$$p(z) = \mathcal{N}(z; 0, I)$$

$$q(z|x) = \mathcal{N}\left(z; \vec{\mu}(x), \mathrm{diag}(\vec{\sigma}(x))^2\right). \qquad (2)$$

Conceptualization

Our proposed method is a novel nonparallel VA can provide favorable audio quality with features for ease of use and scalability for future hardware. The method is intended to work with extremely limited amount of training data, allowing anyone to train and use this system. We have improved the method drastically compared to our earliest research [20, 21] and introducing both voice as color-based processing with multi-objective optimization problems is a first [22] in VA. Unable to target relevant prosodic information previously, we present solutions as well as increased evaluation in this paper. Our approach has two parts: pre-processing of a voice profile and real-time adaptation. The method does not technically consider language, grammar, or physical traits, but treats speech as a selection of unique sounds that a person can produce. By collecting a subject's unique catalog of sounds that make up the voice, we can construct the requested output from a library of many small frames. For a VA to be considered reliable and performant, we require that it has to match unaligned frames depending on various parameters with a high accuracy, and it must be able to correctly relay prosodic information from the original speaker to the target speaker's frame. Instead of comparing acoustic data directly, we encode grouping of vocal features as colors, thus we can compare frames using distancing functions during a multi-objective optimization problem's execution for high efficiency.

Voice adaptation by color-encoded frames

The method is designed with attention towards real-world use, thus allowing both users and developers to benefit from the simple pipeline. Due to this, it is unlike previously seen work and it sees further novelty by introducing voice as color-encoded frames that are matched by employing versatile multi-objective optimization problems (MOOP) using evolutionary computing. We want to match with ideal target frames considering several objectives at once, not possible using the analytical form with voice data as input for the MOOP, thus we propose to encode the most important features into color channels, a simple but flexible data format. While the method is divided into two parts, they share a lot of common supporting methods that takes care of frame generation, normalization, color conversion and more. The differences are shown in Fig. 1 with *A* representing pre-processing and *B* the real-time adaptation.

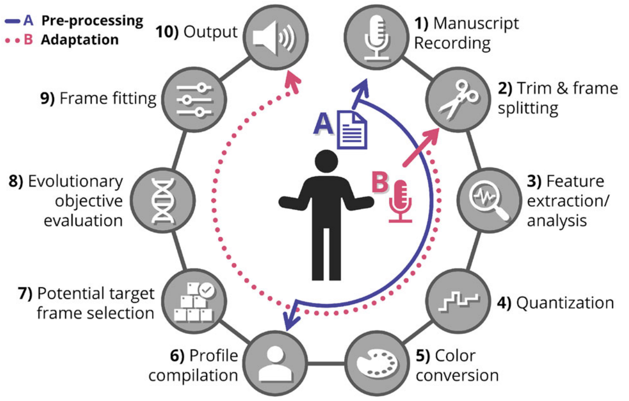

Fig. 1 Pre-processing and adaptation diagram

The reason for encoding speech data as colors is to simplify the format used for processing. The frames that are generated are not used for direct output, but only for matching the inputted speech towards the target voice profile, which is accompanied by its previously created voice recording. In general, for nonparallel VA, the major challenge is to find the ideal unaligned target data, thus related research uses methods such as DNN to train massive amounts of data to statically match data sets $X \rightarrow Y$. We think that the system must not only be easy to use, but fast to train and adjust if it is to see commercial use, hence MOOP is a most suitable technology due to how it mimics natural evolution to reach a set of ideal solutions relatively quickly.

Supporting methods

Manuscript recording

Following Fig. 1's step 1, a voice subject's speech is obtained by recording a manuscript reading which is mostly guided by software. The manuscript is designed to include phonemes and combinations of words, letters, phrases that maximizes the number of unique sounds generated by the subject. The phoneme information includes relevant grapheme, example words and if it is voiced in the case of consonants. The software tells the subject what to say next and at what speech intensity. The length of the manuscript can be customized by specifying a length and the content is automatically generated, and since the adaptation potential rests on the quality of this initial recording we let the algorithm create an ideal manuscript rather than make it by hand. Once steps 1 to 5 are complete the audio data is grouped with a voice profile in step 6 that is completed by mapping various frame-independent data to various points in the obtained audio.

Frame splitting and stylized quantization

Following Fig. 1's steps 2–4, the audio is sampled as 3 ms length segment frames and incrementing with 1.5 ms, all operations henceforth are done on a per-frame basis. The signal is analyzed to extract partial speech features such as fundamental frequency (F_0) and more before it is normalized using a stylized form of quantization. Quantization makes our frame to only take on specific, discrete values and is a base format is intended to cancel out tiny acoustic disparities that would generally impact the frame dependency among speakers. Hence, we can reduce the frame dependency so that Speaker A's components can be recognized in Speaker B's, allowing their frames to be more easily comparable no matter the speaker, and at the same time simplify the acoustic data depending on the resolution of the quantization. The primary parameter we use is pitch and its constraints are the frame's binding to predetermined minimum and maximum fundamental frequencies, in other words the frame is reformed to fit the quantization space to be abstracted. The resolution steps dictate, where the polarizing amplitude points in the frequency maps to, while some coinciding points are discarded. Resolution dictates the new signal's complexity and is enforced by a parameter called alpha. The lower the value, the more abstracted signal is generated, while a higher value is more true to the base input. This quantizer is non-uniform and uses rounding (3) to an alpha step depending on proximity. There is also no reconstruction stage, denoted as Δ, and $\lfloor\rfloor$ denotes a floor function for snapping to the nearest step. Sampling rate is variable, dictated by the adverse values in the current quantization space (Fig. 2) that are considered primary characteristics of the frame in the frequency domain, different for each frame. The normalized frame is used in color-encoding:

$$Q(x) = \Delta \cdot \left\lfloor \frac{x}{\Delta} + \frac{1}{2} \right\rfloor . \tag{3}$$

Color-encoding

Following Fig. 1's step 5, to adjust the format of many speech features into something quantifiable for multi-objective optimization problems, and at the same time consider many crucial speech features rather than just one or a few at once, we group select features into two colors, one for each objective to be evaluated, as shown in Fig. 3. RGBA is an efficient format compared to intricate speech models, and it is easy to compare their likeness; thus, it is an ideal choice for our VA that must be lightweight.

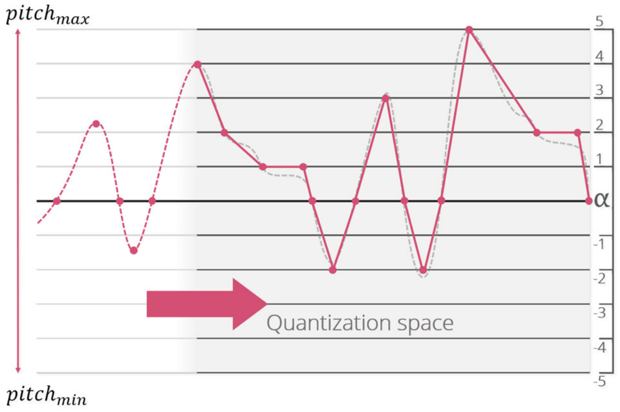

Fig. 2 Resolution alpha dictates the quantized signal complexity

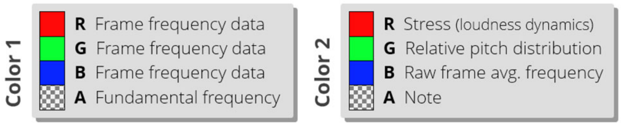
Color 1
🟥 R Frame frequency data
🟩 G Frame frequency data
🟦 B Frame frequency data
▨ A Fundamental frequency
Color 2
🟥 R Stress (loudness dynamics)
🟩 G Relative pitch distribution
🟦 B Raw frame avg. frequency
▨ A Note

Fig. 3 Speech features encoded as color values

Since both sound and color are represented by a frequency; we can convert (4) between them to swap representations of data. $C\upsilon$ is the converted value based on the relationship of the input sound frequency $S\nu_{\text{in}}$ and the ratio of the minimum and maximum relative sound frequencies $S\nu_{\min}, S\nu_{\max}$, and the minimum and maximum relevant light frequencies $L\nu_{\min}, L\nu_{\max}$. While frequency is a common denominator for their values, there are smaller components that play a role. In sound we constraint the relationship of pressure over time while in digital color they are color channels for red, green, blue, and alpha values. The values for each color channel are clamped from a sound frequency converted in (4) to the frequencies associated with visible light, ranging from violet (790 THz) to red (405 THz). While the minimum and maximum hearing range for humans are from 12 to 28,000 Hz, we generally describe it as 20 to 20,000 Hz. In real-time processing and matching of many frames, it is inconvenient to attribute the relationship in sound pressure of the temporal domain for the length of each frame. This would increase voice profile complexity and further disenable the method as voice data volume increases. Using RGBA we only need to consider four simple values for any frame at any given time, and we compare their likeness in color using (5) which is a simplified version of the Euclidean distance function that takes into account the RGB values of two colors which contain a normalized value of the frame's signal, and a prosodic value. For Color 1 the RGB components are obtained from the previously normalized frame, and most other features are easy to extract. Using features such as F_0 allow us to predict if the speaker is a male or female. Some features are based on logarithmic spectrums, others linearly and dictated by real-time adjustments. We can obtain a relative pitch distribution by defining a window size, e.g., 50 ms and apply

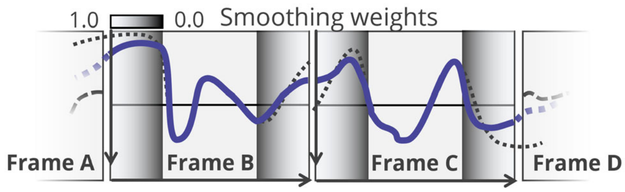

Fig. 4 Frames are smoothed together for a more natural sound

a function such as short-time fourier transform (STFT) (6) which is ideal for our use case that has short, practical signals. The equation is for discrete-time case, where the signal $x[n]$ can be directly considered as frames that overlap each other in a window $w[n]$. Both m and ω are discrete. While not the most accurate method for obtaining a pitch distribution, it is effective and gives a compromised data between time and signal view of the origin. The color components each are valuable insights into the speaker's uniqueness, but by considering several of them it is possible to predict various changes, so they make for good heuristics:

$$Cv = \left(\frac{Sv_{\text{in}} - Sv_{\text{min}}}{Sv_{\text{max}} - Sv_{\text{min}}}\right) \times (Lv_{\text{max}} - Lv_{\text{min}}) + Lv_{\text{min}} \tag{4}$$

$$\text{dist} = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2} \tag{5}$$

$$\text{STFT}\{x[n]\}(m, \omega) \equiv X(m, \omega) = \sum_{n=-\infty}^{\infty} x[n]w[n - m]e^{-j\omega n}. \tag{6}$$

Final frame adjustments

Figure 1's steps 7 and 8 are explained in the next section, but they perform selection of a target frame population and evaluate them against an input frame using evolutionary computing techniques. Following step 9, once a target frame has been selected for output (step 10), it needs to be processed further before it can be used. Target frames are obtained from various unrelated positions in the selected target voice profile, in other words the collected frame's temporal structure does not naturally blend where one frame ends and the next starts. To compensate for this, partial frame smoothing (Fig. 4) is used to stitch the target frames together more naturally. Using weighted smoothing for only parts of the frame we can improve the sound quality. In addition, the voice profile is not intended to store audio of varying speech intensities, rather it is neutral at a conversational volume level. To correctly portray the source speaker's intensity in the output audio, we measure the intensity for each input frame and translate (Fig. 5) it onto the matched target frame. This simple operation further improves sound quality and experience.

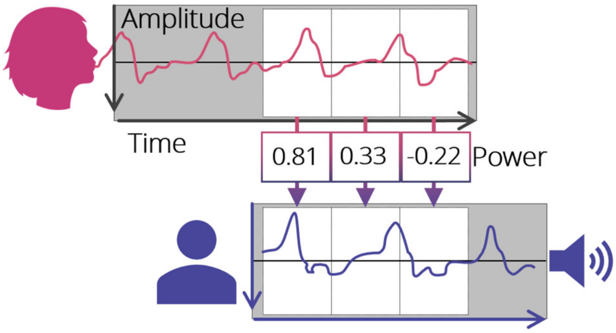

Fig. 5 Source power is translated onto the target frame

Matching by multi-objective optimization problems

Multi-objective optimization problems are a set of single-objective problems that are evaluated according to a scalar function using a set of weight vectors (7). Using a framework MOEA/D [23], we can evaluate several objectives such as temporal and spectral frame alignment at the same time rather than just one. The objective space for m-objectives is normalized according to (8), while $x \in X$. Minimization $(i = 1, 2, \ldots, m)$, x occurs of the ith objective $f_i(x)$ which is a decision vector and X is the feasible region of x in the decision space. A set of weight vectors H, the same as the population size are uniformly distributed the start of the evaluation according to (9). For each single objective, a weight vector similarly has a single solution, and the purpose of the single objectives is to search for the best solution along each weight vector depending on proximity and neighborhood size configuration. Depending on the problem type, solutions move differently. In this case they start randomly spread out distanced from the base point z and arcs uniformly closer to z the more generations they evolve through

$$w = (w_1, w_2, \ldots, w_m) \tag{7}$$

$$\text{Minimize} f(x) = (f_1(x), f_2(x), \ldots, f_m(x)) \tag{8}$$

$$\begin{gathered}\sum_{i=1}^{m} w_i = 1 \text{ and } 0 \le w_i \le 1 \quad \text{for } i = 1, 2, \ldots, m \\ w_i \in \left\{0, \frac{1}{H}, \frac{2}{H}, \ldots, \frac{H}{H}\right\} \quad \text{for } i = 1, 2, \ldots, m.\end{gathered} \tag{9}$$

This tool is often used in economics and engineering fields to present an optimal solution for conflicting multiple objectives. The result can be represented as a Pareto Front (Fig. 6) which contain multiple solutions, each representing a set of values towards multiple objectives. In our case we have two objectives, a normalized frame color and a prosodic frame color (Fig. 7). The normalized frame gives us the essential temporal features in a simplified form, while the prosodic

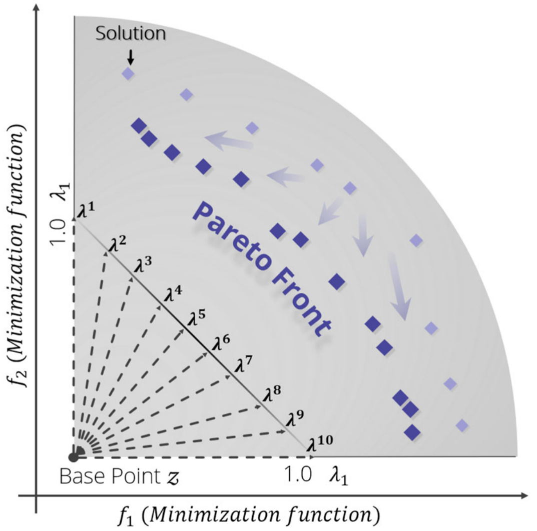

Fig. 6 Generated pareto front with 15 solutions up to 12 generations. The number of pareto optimal solutions increases

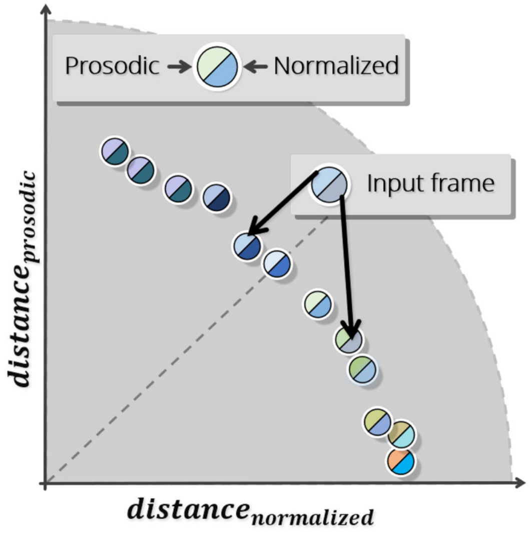

Fig. 7 Weighted line with explanation of solution frame colors

frame represents spectral features. We have seen previously that speech quality is inconsistent by only trying to achieve one objective (e.g., only matching against temporal features). Using MOEA/D we can let evolutionary computing present a set of solutions in a population after n generations which contains ideal target frames from which we select one most ideal frame from, depending on criteria such as weighing what objective is more important. To populate the instance, we can simply select the n closest voice profile frames of the same Note as the input frame, thus instead of arbitrary selection we already have come closer to an ideal target sound by utilizing spectral information encoded in the frame. The way frame colors are evaluated is by a Euclidean distance function in conjunction with the objective function of DTLZ-2 (10), in which we use a base conversion by shifting the values into a 32-bit unsigned integer. The problem type has a continuous search space and is unimodal.

MOEA/D runs in two instances (configuration in Table 1), deliberately lagging the output by 1 frame to allow for initialization. MOEA/D uses random initialization based on a time-generated seed after populating, so the amount of time spent until a target frame match is decided is never quite the same. The population size and max generations are limited to increase performance, and while ideally more generations would give better results as it provides more pareto optimal solutions to the set, we mainly use it to distribute the solutions for selecting the target frame without having to make arbitrary rules. Ultimately in evolutionary computing the goal is to achieve optimal solutions in the least amount of time or generations so that it can be applied to problems classified as real-world problems:

Table 1 MOEA/D configuration

Parameter	Value
Problem type	DTLZ2
Test function	TCHn1
Max generations	12
(or max time allotment)	5 ms
Population size	15
Neighborhood solutions	3
Parallel instances	2 threads

$$\begin{aligned}
&\text{Minimize } f_1(\vec{x}) = (1 + g(\vec{x}))\cos\left(x_1 \frac{\pi}{2}\right)\\
&\text{Minimize } f_2(\vec{x}) = (1 + g(\vec{x}))\sin\left(x_1 \frac{\pi}{2}\right)\\
&g(\vec{x}) = \sum_{x_1 \in \vec{x}} (x_1 - 0.5)^2\\
&\text{subject to } 0 \le x_1 \le 1 \quad \text{for} \quad i = 1, \ldots, n.
\end{aligned} \tag{10}$$

Intended use in video games

The vision behind this research is to prove that a new pillar of interaction for online video games, where players communicate via speech, which is a feature more and more games not only offer, but players take advantage of—could truly be beneficial for all parties. Game designers spend a lot of time creating the world, characters, and story, however, when players take control over those characters and communicate it may break immersion with voices not matching etc. Reinforcing the immersion with VA would give the designed game setting a large boost and allow for new design ideas regarding voice work previously unimaginable. Players with passion for the lore or game setting can be given a chance to become their favourite character, a group of players playing a cooperative game can each become a tailored character with a place in the story, and for other cases it can be a selection between several voices, where the player can choose a voice profile they like or change it in between. With VA, a player can customize their online persona 100% from naming, looks and speech, something that has only been possible with present recordings conventionally and it might be a way for curious prospective players to ease their way into new online experiences.

Not only improving immersion, but VA in games could benefit various players all around the world. For example, sexism [24] in online video games is so prevalent, many gamers do not want to expose the fact that they are female. Having VA as an option in the game's setting might relieve some players while still being able to communicate and enjoy the game without fearing harassment. In addition, privacy, especially regarding children could benefit from a voice-concealing method, such as VA. According to a study [25] in the US, a high percentage of children ages 8–17 play at least 2 h of video games every day, lacking adult experience regarding their online presence and etiquette. We can assume the figures are similar in other developed countries and with the increase of access to technology such as children with their own smartphones it is a number likely to have grown. It might improve their online wellbeing to have concealed voices to avoid not only harassment but predatory behavior. In addition, it's a technique that does not build on one specific language per say. As long as the voice profile have sufficient unique library of sounds it can work across languages.

Finally, from the developer's side it could be both time and money-saving to bring in a voice subject to create a larger voice profile than record specific voice lines that may need revision later. With a voice profile they can tailor their own voice lines according to a detailed custom character theme at any point without having to do traditional sound engineering work. Since this technology has not seen commercial use, it is likely there are many more areas of use not realized yet.

Experiments and results

The selection of methods is intended to validate not only the speech quality, but the design of the method in terms of usability and computational performance. We have used two voice profiles, *M* (5 min native English) and *L* (8 min for each native English and native Norwegian). The evaluation subjects were from various nationalities (ages 20–29), but possesses native English level, and none have a background related to speech processing. The test environment can be seen in Table 2. The input device (for voice profile recording) is a dynamic microphone distanced about 10–20 cm from the subjects to minimize natural background noise as it is not performed in a professional studio environment.

Table 2 Test environment

Component	Specification
OS	Windows 10 64-bit edition
CPU	Intel Core i5-4690 K 3.5–3.9 GHz
GPU	ASUS STRIX RX480 8 GB
RAM	16 GB
Input device	Dynamic microphone at 44.1 kHz/48 kHz sample rate, 50–15,000 Hz response range
Software	.NET5 custom application

Evaluation methods

1. *Benchmarks*: We measure the time it takes to create the voice profiles and how fast frames can be processed into colors or find an ideal output frame. We tested 40 utterings that amount to almost 35,000 frames. On average the duration for saying "Hello" is about 1 s, and after processing becomes near 500 frames.
2. *MOS*: The *Mean Opinion Score* measures the VA quality and scoring is done using the Absolute Category Rating (ACR), where a score of 1 is labeled *Bad* and 5 is *Excellent*. The scoring is done for each stimuli provided and is calculated as the arithmetic mean for *N* subjects. 50 samples of varying length were presented.
3. *ABX*: Two audio samples, A and B are presented to a listener, usually utterings of the same segment, phrase or words. Finally, sample *X* is presented which can be A or B. The order of playback is random and a low probability means that the subject had difficulties in identifiying which sample was what. For each voice profile there are 20 utterings of various length.
4. *MUSHRA*: The *MUltiple Stimuli with Hidden Reference and Anchor* test can work with fewer participants and still yield statistically favorable results. This test is ideal for testing more difficult items than MOS and 7 stimuli are rated in comparison to a reference item using a scale from 0 to 100. One of the stimuli is a hidden (non-adapted uttering) reference and two others are unlabeled anchors which is are purposely poor examples. This decreases the chance of randomly receiving poor or good scores as listeners have clear examples of good and bad stimuli. Listeners are also told about artifacts to look out for, as it improves statistical impact of the result to have trained listeners. It is recommended to use at least two anchors of 7 kHz and 3.5 kHz each, but this is mostly for the use of classical signal evaluation. Since this is a voice subject we kept the 7 kHz sample and replaced the 3.5 kHz with a poor adaptation example, although

referred to as the ‘Poor Sample Anchor’. The poor sample is a case, where the output stream contains many unfavorable frames (higher than ideal color distance).

5. *SEQ:* The *Single Ease Question* is a 7-point scale that measures the satisfaction after a task is performed. It’s a simple test that works well standalone to give insight into subject’s thoughts without requiring detailed diagnostics. Instead of the classic method of 3 questions per task, we combine 5 questions for a sample size of 12 test subjects.

Evaluation results

1. Benchmarks (Figs. 8, 9).
2. MOS (Fig. 10).
3. ABX (Fig. 11).
4. MUSHRA (Fig. 12).
5. SEQ (Fig. 13).

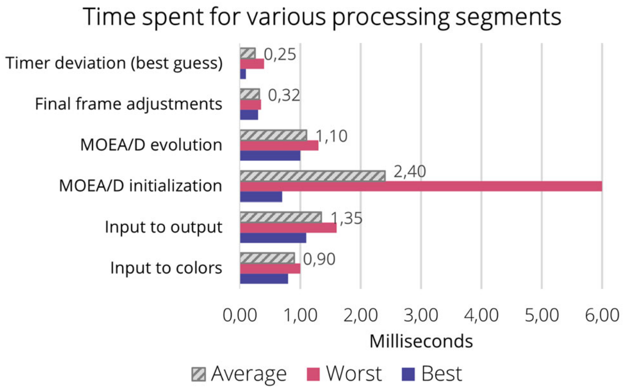

Fig. 8 Various processing speeds for the proposed method

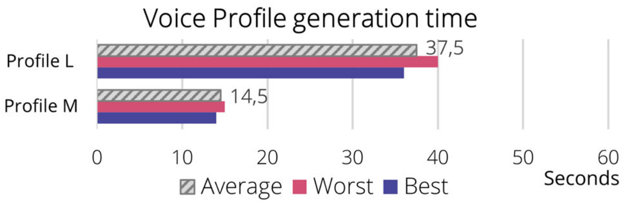

Fig. 9 Processing time to create voice profiles from recordings

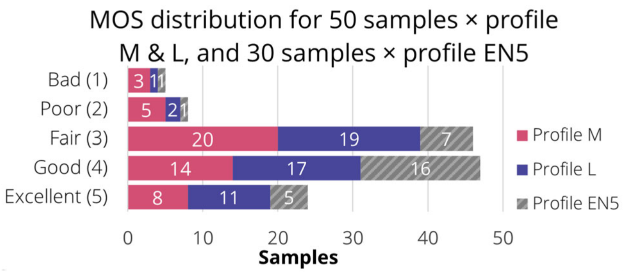

Fig. 10 Absolute category rating used to score samples

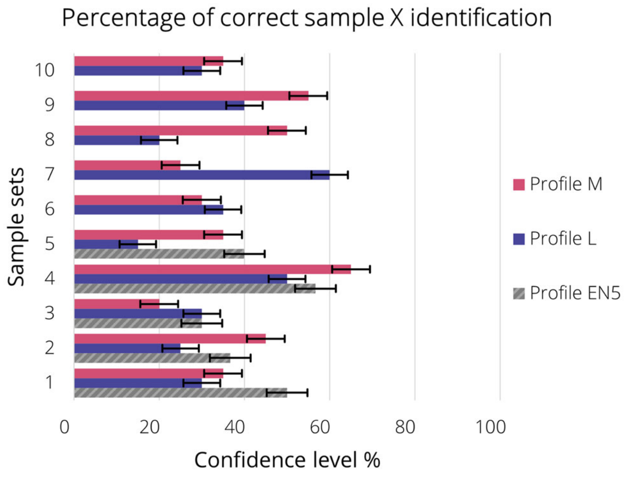

Fig. 11 Lower confidence level is better

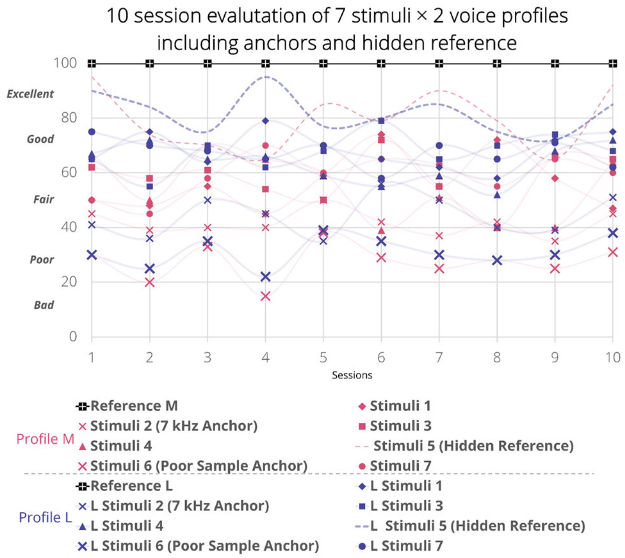

Fig. 12 Comparison of sound quality between two voice profiles

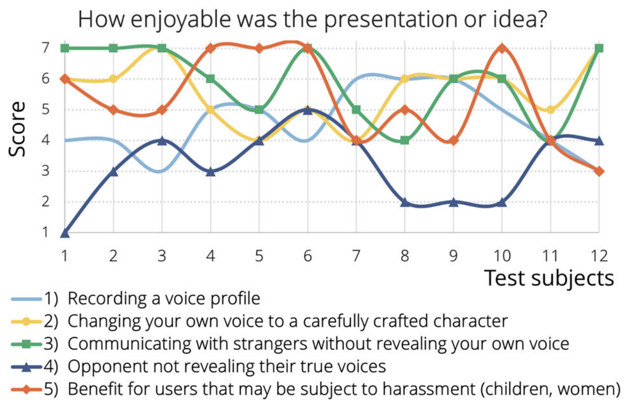

Fig. 13 Higher values indicate more enjoyment for an activity

Discussion

Thoughts regarding evaluation results

We see that compared to our previous work [22], by decreasing the frame length and increasing population in MOEA/D we were able to achieve more favorable results reflected in the MOS test (Fig. 10). Included in the chart is the evaluation from previous work labeled *Profile EN5* (English, 5-min manuscript). Compared to the previous work, the number of samples were increased by over 60%, leading to more statistically significant results. Using MOOP as a way of matchmaking frames resolved such issues in [20, 21], where target frames sometimes could not be resolved, due to better evaluation along multiple objectives rather than just one. Selecting the new population using heuristics such as the encoded color value for Note or Fundamental Frequency contributes to faster selection (Figs. 8 and 9) of solution candidates than previously, allowing for quicker initialization of a population, and additional performance improvements in the time domain due to smaller frame sizes. However, after this initial populating procedure MOEA/D will initialize the solutions by performing 1 generation worth of evolution. This is a part of the total initialization and its time can be seen to be vary too much. As for the blind test, ABX seen in Fig. 11, we see the generally the confidence level is below 50% which means that in most cases the subject had difficulties identifying the adapted stimuli. Profile EN5 was included for reference; however, the number of sample sets and playbacks were different. In overall, the later results achieve lower confidence level which can be attributed to the suggested improvements in this paper such as encoding temporal and spectral features into separate colors. The most impactful evaluation is MUSHRA (Fig. 12) as it includes great reference and poor stimuli so that there is little chance for accidental good evaluations. The test shows that listeners were able to identify good stimuli and the adapted stimuli is generally rated from Fair to Good. We see in all the tests that the larger voice profile L scores higher. This is to be expected as it is not only bigger, but contains sounds from two languages, thus improving its adaptation potential. For a commercial use voice profile, it could be imagined that something like a 15-min recording would yield even better results, although the goal is to have as short a recording as possible. In the SEQ test we see in Fig. 13 that most subjects were positive towards having the ability to either change voice to fit a character, or use it to disguise their personal voice, but at the same time are not quite as positive towards the fact that the same features could be used by an opponent.

There is a tradeoff to having a fast and lightweight method, and that is that voice quality is not always at ideal levels, compared to methods such as neural networks which have quite consistent results, although require a lot more training. We think the current design has sufficient performance, but one way to improve on this aspect would be to have access to more computational power, not only the CPU. By, e.g., running the MOOP instance on the GPU, it would achieve a higher count of generations, which provide more pareto optimal solutions to choose from that can provide us with frames that have better color likeness.

Comparing results to related modern work

In [16] they use a CycleGAN and achieves an MOS score ranging from a little above Poor up to Fair, and an ABX test which scores good, especially compared to other CycleGAN methods; however, since it is a convolutional neural network and a Cycle type, the amount of training is quite high as it cycles between a pair of GANs. Despite being nonparallel in the working method, all voice subjects undergo training to map $G_{X \rightarrow Y}$. NN-based training can have good results, but the mapping can be very static and strict, thus it is difficult to use in a dynamic environment, where you may want to change voice subjects or training data on the fly. Fang et al. [15] presents a CycleGAN method as high quality, and while better than compared GAN methods, the MOS test show under or near average scores on the scale despite substantial training, such as 10 sub-data sets for each speaker. The training of NN is often considerable to the point that it is performed on external services, such as cloud or cluster computing, which utilize both CPU and GPU processing. In [19] using WaveNet vocoder, unlike traditional vocoders it is trainable with DNN. They train multiple speakers to capture shared properties. The matching is mostly based around F_0 when it can be traced. When it is out of range the speech quality degrades, as the system is somewhat prediction-based. They use both large multi-speaker and limited parallel data sets. In an MOS test they see results scored Fair and the ABX test shows good results but compared against deep bidirectional long short-term memory (DBLSTM) neural network. Sekii et al. [26] sees good MOS results for their DNN approach, but execution time is not ideal for real-time use as it requires time to extract, convert and restore features. Kotani et al. [27] while using larger data and time spent training sees typical VA performance and [28] with a route of frequency warping by considering many warping factors achieves decent results compared to the then state-of-the-art statistical methods. Many speaker data sets, layers in NN and long training times across various methods all fall towards the average mark, while [29] (DNN) see mixed results regarding speaker likeness in relation to training volume.

The balance between training volume and speech quality is the primary challenge in the VA field, but this time the method seems to be quite portable, and it is easy to add new voices with few dependencies. Despite this, NN-based solutions are not as dynamic as ideal and the interaction requires some technical insight, while it can achieve good results by mapping voice A to voice B, a voice C cannot simply enter the process without due training so that it can become a mappable set among the rest. Erro et al. [30] surveyed a novel method dealing with frequency warping in conjunction with amplitude scaling by the use of smaller amounts of parallel data. The evaluations saw good results, although subjective, proved that classical methods such as only frequency warping has difficulty in competing with the likes of NN. Among contemporary research is [31] which proposes to utilize Text-to-Speech (TTS) as an approach to training a model via a Recurrent Neural Network (RNN) vocoder for transfer learning. Their results show improvement over simpler techniques, and a slight improvement over similar (TTS-utilizing) methods. It requires a two-step training scheme following a strict framework and NN configuration. In [32] sequence-to-sequence (seq2seq) model is applied, where they attempt to solve the issue of unstable training by allowing pretraining of an RNN. It is said that seq2seq is favorable for handling prosodic representation if the training data is of substantial size. They used TTS and automatic speech recognition (ASR) tasks for pretraining, and from 80 to 932 training utterances used. Mel cepstral distortion (MCD) was used to measure the network. Results show improved speech quality due to pretraining; however, it was tested as a one-to-one (parallel) interface only. Zhou et al. [33] focuses on the prosodic aspect using an emotional speech data set with a method based on a VAE-GAN combination called VAW-GAN (Wasserstein). The evaluation was done with AB, ABX and MOS tests to compare similarity in uttering with focus on emotion (neutral to sad, etc.). Like the above research, results show an improvement in clustering of features to match utterances due more accurately to pretraining. A major challenge with NN is to correctly teach the problem to the network, pretraining makes sense to be trending in various research, and TTS + ASR simplifies creation of training material without real voice subjects; however, it frankly is just layered training which requires more computational resources, tailored parameters, and time, although the ideas are very interesting. It would be interesting to see if similar ideology could be applied to MOOPs to reduce the number of generations required for better performance. It can be said that NN generally achieve favorable results, but at the cost of substantial training not only in size, but time as well. It also requires many external frameworks to operate if it were to be used by game developers, whereas an MOOP by design can be reconfigured without retraining, even during runtime, and voice profiles can be improved upon quickly with additional recording, without having to rely on large computation times.

Benefit of MOOP in VA and future-proofing

The real power of MOOP is that it mimics nature, how genetic information evolves and mutates over generations. However, unlike the real world, where generations can take days to years depending on the organism, frameworks, such as MOEA/D can simulate thousands of generations in seconds on a CPU. In addition, the data used in the evolution of the population (solutions) is not obtained from large data sets. They are randomly initialized, and the evaluation of their objective functions motions the solutions. The framework contains many problem types and configurations, such as functions used during evolution, so if we have a datatype that is compatible with existing problem types it is easy to take advantage of this system. Another option is to create a new problem type; however, it is difficult to troubleshoot issues in genetic algorithms. The reason for using this in conjunction with VA is that the comparison and selection of a target frame based on the input frame is not ideal when you are only evaluating one objective, such as F_0 or other speech features. It will produce good results in some cases and poor in others, thus we need to consider multiple objectives at once. Instead of considering simply F_0 and some other aspect, e.g., MCD, we encode many features into our frame colors, thus we are left with 2 objective values that can be examined in detail when needed, but also compared directly using cheap distance functions. The distance function may also be exchanged with something even more lightweight, such as Manhattan distance, or if the system is on a more powerful computer than is common of the current time a more accurate distance function may be used. This also plays into how MOOP can be configured, as hardware gets better, we can increase parameters, such as population size and max generations which will yield improved results. Evolutionary computing works so that the longer we can simulate the genetic evolution, the better results will become although there is a point when results cease to improve drastically and become marginal, it is far away from the current generations we are currently using for the given problem type.

Considering VA in future game software

In [22] we attempted a small-scale test as a proof of concept using [34–36] and realized that MOEA/D and the proposed VA can be used in real-time applications, such as games without consuming all the computational resources, proving this design is very lightweight. For future work we will attempt a more substantial test based on the same software and hopefully it can become the first prototype VA used in a modern game project. Just how character customization once was novel, we hope to see VA the same, and for two reasons. First, results show that audience are positive towards having such a feature and it might encourage players to communicate more freely without worrying about exposing themselves. Second, for players who want to avoid harassment due to demographic, gender, or age it is an excellent technology that can ease their worry and rather put focus back into enjoying the game while being able to communicate.

Conclusion

We presented a novel method to perform voice adaptation by encoding speech features into colored frames that are used in a multi-objective optimization problem to find an ideal target frame depending on the colors of a given input frame. We demonstrated that this method can be used without lengthy preparation or training which is imperative if VA is to see commercial use, such as online video games. Evaluation methods were traditional MOS, ABX for grading the speech processing and MUSHRA for statistical confirmation of results, as well as benchmarking of the processing speed and finally a SEQ test used indicate the audience's perception of VA as a game technology. The results show that the computational footprint is quite low; however, while utilizing MOEA/D has many benefits, its downside is that it can require more time than ideal to initialize for each new frame. As for future research, we propose that GPU accelerated computing for this aspect of the method could not only solve this issue with long initialization times for the evolutionary computing part but improve performance greater than achievable only using even the highest end CPU for calculations.

Acknowledgements The authors are grateful to Prof. M. Sato from Tokai University for their valuable comments and discussion, as well as insight into evolutionary computing. This work was supported by JSPS KAKENHI Grant No. JP19K12162.

Declarations

Conflict of interest On behalf of all authors, the corresponding author wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

Informed consent Consent is obtained from the participants aiding the study which resulted in the presented evaluation data. Technical implementation and primary researcher is the corresponding author, while research supervision is contributed to the 2nd author and manuscript quality assurance from the 3rd author.

References

1. Eason Y, Stylianou (2009) Voice transformation: a survey. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Taipei, pp 3585–3588
2. Erro D, Moreno A (2007) Weighted frequency warping for voice conversion. In: 8th Annual Conference of the International Speech Communication Association INTERSPEECH, Antwerp, pp 1965–1968
3. Stylianou Y, Cappé O, Moulines E (1998) Continuous probabilistic transform for voice conversion. IEEE Trans Speech Audio Process 1:285–288
4. Toda T, Saruwatari H, Shikano K (2001) Voice conversion algorithm based on Gaussian mixture model with dynamic frequency warping of STRAIGHT spectrum. In: Proc. ICASSP, pp 841–844
5. Moulines E, Sagisaka Y (1995) Voice conversion: state of the art and perspectives. Speech Commun 16(2):125–126 (**Special Issue**)
6. Kain A, Macon MW (1998) Spectral voice conversion for text-to-speech synthesis. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seattle, pp 285–299
7. Ye H, Young S (2006) Quality-enhanced voice morphing using maximum likelihood transformations. IEEE Trans Audio Speech Lang Process 14(4):1301–1312
8. Chen Y, Chu M, Chang E, Liu J, Liu R (2003) Voice conversion with smoothed GMM and map adaptation. In: 8th European Conference on Speech Communication and Technology (Eurospeech 2003—Interspeech 2003), Geneva, pp 2413–2416
9. Wu Z, Virtanen T, Chng ES, Li H (2014) Exemplar-based sparse representation with residual compensation for voice conversion. IEEE Trans Audio Speech Lang Process 22(10):1506–1521
10. Takashima R, Takiguchi T, Ariki Y (2012) Exemplar-based Voice conversion in noisy environment. In: IEEE Spoken Language Technology Workshop (SLT), Miami, pp 313–317
11. Villavicencio F, Bonada J (2014) Voice conversion using deep neural networks with layer-wise generative training. IEEE/ACM Trans Audio Speech Lange Process (TASLP) J 22:1859–1872
12. Sato Y (2004) Voice quality conversion using interactive evolution of prosodic control. Appl Soft Comput J 5:181–192
13. Abe M, Nakamura S, Shikano K, Kuwabara H (1988) Voice conversion through vector quantization. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 1, New York, pp 655–658
14. Villavicencio F, Bonada J (2010) Applying voice conversion to concatenative singing-voice synthesis. In: 11th Annual Conference of the International Speech Communication Association (INTERSPEECH), Chiba, pp 2162–2165
15. Fang F, Yamagishi J, Echizen I, Lorenzo-Trueba J (2018) High-quality nonparallel voice conversion based on cycle-consistent adversarial network. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, pp 5279–5283
16. Kaneko T, Kameoka T, Tanaka K, Hojo N (2019) Cyclegan-VC2: improved cyclegan-based non-parallel voice conversion. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, pp 6820–6824
17. Hsu C-C, Hwang H-T, Wu Y-C, Tsao Y, Wang H-M (2016) Voice conversion from non-parallel corpora using variational auto-encoder. In: Asia–Pacific Signal and Information Processing

Association Annual Summit and Conference (APSIPA), Jeju, pp 1–6

18. Lorenzo-Trueba J, Yamagishi J, Toda T, Saito D, Villavicencio F, Kinnunen T et al. (2018) The voice conversion challenge 2018: promoting development of parallel and nonparallel methods, Odyssy 2018
19. Liu L, Ling Z, Jiang Y, Zhou M, Dai L (2018) WaveNet Vocoder with limited training data for voice conversion. In: Proc. Interspeech 2018, pp 1983–1987
20. Midtlyng M, Sato Y (2016) Real-time voice adaptation with abstract normalization and sound-indexed based search. In: IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, pp 60–65
21. Midtlyng M, Sato Y (2018) Voice adaptation from mean dataset voice profile with dynamic power. In: IEEE International Conference on Systems, Man, and Cybernetics (SMC), Shizuoka, pp 2037–2042
22. Midtlyng M, Sato Y (2020) Lightweight multi-objective voice adaptation for real-time speech interaction applied in games. In: IEEE Conference on Games (CoG), Osaka, pp 237–243
23. Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 11(6):712–731
24. Fox J, Tang WY (2014) Sexism in online video games: the role of conformity to masculine norms and social dominance orientation. Comput Hum Behav 33:314–320
25. Rideout V (2015) The common sense census: media use by tweens and teens. Analysis & Policy Observatory, Common Sense Media
26. Sekii Y, Orihara R, Kojima K, Sei Y, Tahara Y, Ohsuga A (2017) Fast many-to-one voice conversion using autoencoders. In: International Conference on Agents and Artificial Intelligence (ICAART), Porto, pp 164–174
27. Kotani G, Saito D, Minematsu N (2017) Voice conversion based on deep neural networks for time-variant linear transformations. In: Asia–Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Kuala Lumpur, pp 1259–1262
28. Tamura M, Morita M, Kagoshima T, Akamine M (2011) One sentence voice adaptation using GMM-based frequency warping and shift with a sub-band basis spectrum model. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, pp 5124–5127
29. Li Y, Lee KA, Yuan Y, Li H, Yang Z (2018) Many-to-many voice conversion based on bottleneck features with variational autoencoder for non-parallel training data. In: Asia–Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hawaii, pp 829–833
30. Erro D, Navas E, Hernáez I (2013) Parametric voice conversion based on bilinear frequency warping plus amplitude scaling. IEEE Trans Audio Speech Lang Process 21(3):556–566
31. Zhang M, Zhou Y, Zhao L, Li H (2021) Transfer learning from speech synthesis to voice conversion with non-parallel training data. IEEE/ACM Trans Audio Speech Lang Process 29:1290–1302
32. Huang W-C, Hayashi T, Wu Y-C, Kameoka H, Toda T (2021) Pretraining techniques for sequence-to-sequence voice conversion. IEEE/ACM Trans Audio Speech Lang Process 29:745–755
33. Zhou K, Sisman B, Liu R, Li H (2021) Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset. In: ICASSP 2021—2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp 920–924
34. Microsoft .NET5 SDK (2020) [Online]. Available: https://dotnet.microsoft.com/download/dotnet/current
35. Microsoft WebView2 web rendering (2020) [Online]. Available: https://docs.microsoft.com/en-us/microsoft-edge/webview2/
36. Unity3D, Unity Technologies. Accessed on: January 1. 2019 [Online]. Available: https://unity.com/