OPEN ACCESS Freely available online
Y. Liu and C. Jin, "A Similarity-Based Conditioning Method for Controllable Sound Effect Synthesis," *J. Audio Eng. Soc.*, vol. 73, no. 9, pp. 504–518 (2025 Sep.).
https://doi.org/10.17743/jaes.2022.0226.

A Similarity-Based Conditioning Method for Controllable Sound Effect Synthesis

YUNYI LIU,* AND CRAIG JIN
(yunyi.liu@sydney.edu.au) (craig.jin@sydney.edu.au)

Computing and Audio Research Laboratory, Department of Electrical and Information Engineering, The University of Sydney, Sydney, Australia

Generating sound effects with controllable variations is a challenging task, traditionally addressed using sophisticated physical models that require in-depth knowledge of signal processing parameters and algorithms. In the era of generative and large language models, text has emerged as a common, human-interpretable interface for controlling sound synthesis. However, the discrete and qualitative nature of language tokens makes it difficult to capture subtle timbral variations across different sounds. In this research, the authors propose a novel similarity-based conditioning method for sound synthesis, leveraging differentiable digital signal processing. This approach combines the use of latent space for learning and controlling audio timbre with an intuitive guiding vector, normalized within the range [0, 1], to encode categorical acoustic information. By utilizing pretrained audio representation models, this method achieves expressive and fine-grained timbre control. To benchmark their approach, the authors introduce two sound effect datasets—Footstep-set and Impact-set—designed to evaluate both controllability and sound quality. Regression analysis demonstrates that the proposed similarity score effectively controls timbre variations and enables creative applications such as timbre interpolation between discrete classes. This work provides a robust and versatile framework for sound effect synthesis, bridging the gap between traditional signal processing and modern machine learning techniques.

0 INTRODUCTION

Timbre is a musical term used to describe the unique perceptual differences of distinct musical sounds [1], often referred to as "the color or tone of a sound" [2]. In the realm of sound effects that are nonmusical and nonspeech sounds, the authors adopt this terminology more loosely to refer to subtle auditory characteristics that are not easily captured by conventional audio features using temporal or spectral analysis. It serves as a critical audio descriptor, enabling listeners to identify sounds and contributing significantly to the realism and immersive quality of audio experiences[3]. For instance, the sound of footsteps can vary dramatically based on the surface being walked upon. Materials such as wood, gravel, or metal each produce distinct acoustic signatures that reflect their interaction with the environment [4, 5].

The ability to accurately and meaningfully interpolate between these timbres is essential for sound designers aiming to create dynamic and contextually rich soundscapes that mirror varied environmental interactions [6]. This capability is particularly important in industries such as film, gaming, and virtual reality, where nuanced sound design enhances narrative depth and user engagement, ultimately enriching the overall auditory experience [7].

Generating sound effects with controllable variations has been a persistent challenge in audio synthesis [8–10]. Traditional approaches often rely on sophisticated physical models, which demand a deep understanding of signal processing parameters and algorithms. To streamline this process, physically driven neural audio synthesis methods have emerged, leveraging neural networks to reconstruct physical priors from sound, thereby enabling control over the synthesis [11].

While these methods are powerful, they often lack intuitive controls, making them less accessible to nonexperts. Additionally, the perceptual subtleties of audio DSP parameters are not always intuitive, posing challenges even for experienced users. The limited availability of diverse, labeled datasets further hinders the development of flexible and broadly applicable sound synthesis systems, underscoring the need for more user-friendly and adaptable approaches.

In the era of neural audio synthesis, generative models are often conditioned on various modalities, such as class

*To whom correspondence should be addressed, email: yunyi.liu@sydney.edu.au.

labels [12, 13], textual descriptions [14], and visual inputs [15, 16], to guide sound production. While these approaches enable diverse forms of control, they frequently struggle to capture the nuanced timbral differences inherent in distinct environmental contexts. This limitation arises from the reliance on descriptive, hand-crafted labels, which are often insufficient for achieving fine-grained control over sound expression.

The differentiable digital signal processing (DDSP) framework [17] offers a promising alternative by conditioning sound synthesis on detailed features, enabling rapid training and high-quality audio generation. In a recent survey, Hayes et al. [18] provided a comprehensive review of the applications and limitations of DDSP in the domain of music and speech processing. By incorporating known signal processing techniques, it was shown that DDSP is able to achieve efficient, interpretable and controllable neural audio synthesis/processing models [19, 20].

However, with the inductive bias, DDSP is usually constrained on its generalization capability on source-agnostic signals such as inharmonic or irregular sounds [21], which are essential for the production of Foley. Further, due to the autoencoder design of the DDSP architecture, many research relies on modifying the latent space via techniques like remapping latent space and timbre space [22] to obtain an interpretable latent space [23]. Addressing these limitations is essential for advancing the flexibility and realism of neural audio synthesis systems.

In this research, two primary objectives are pursued: (1) to generate diverse variations of sounds using a controllable guiding variable that modulates timbre and (2) to synthesize high-quality inharmonic and impulsive sound effects, such as footsteps and gunshots. While most sound effect datasets are categorized by sound type, the authors aim to extract and leverage timbral information both within individual categories and across different sound types. This approach captures the similarity of a sound relative to all categories, preserving essential timbre cues that serve as valuable conditioning information.

To ensure high-quality sound synthesis while maintaining computational efficiency for both training and inference, this method is evaluated using a modified DDSP network specifically tailored for synthesizing impulsive broadband sound effects. This framework enables precise control over timbre while meeting the demands of diverse and dynamic soundscapes.

1 RELATED WORKS

In the domain of neural audio synthesis, the latent space has been extensively explored to control and interpolate the styles of high-level features of the target audio [24–26]. StyleGAN [27] introduces an unsupervised approach that uses intermediate latent vectors to encode different characteristics of the synthesized output, allowing for style mixing. Although originally developed for visual data, this approach can be adapted to sound synthesis to capture and combine diverse stylistic attributes [28]. However, it remains unclear which specific high-level feature is controlled by each sub-latent component of the latent vector.

Given the complexity and intricacy of the latent space, additional effort is often required to establish meaningful control parameters that correspond to specific acoustic attributes. This can be achieved through techniques such as perceptual timbre remapping [29, 22] or adversarial fine-tuning [30]. Audio datasets, particularly in the sound effects domain, often lack detailed descriptive annotations. To address this, many research efforts have focused on extracting meaningful timbral information from audio samples using an analysis-synthesis approach. A straightforward method for achieving a degree of audio control involves using preprocessed audio features, such as pitch [31], loudness [17, 32], or spectral centroid [33], to guide synthesis. These models are typically trained in a supervised learning manner, which often reduces the requirement for extensive datasets.

However, the complexity and diversity of sound effects pose significant challenges. Many variations in sound effects cannot be effectively controlled using only DSP-extracted audio features. For instance, it is exceedingly difficult to transform the timbre of a footstep on a concrete floor into a footstep on snow or sand using solely DSP-based audio descriptors. While class-conditional sound synthesis methods [34, 13, 12] can capture subtle differences between categories, such as different types of footsteps, their capacity for generating new timbres is limited by the discrete nature of one-hot vector representations commonly used in class-conditional models.

To better leverage the information available in sound datasets, Nistal et al. [35] proposed a knowledge distillation approach that enables semantic control over sound categories. This method uses “soft labels” derived from a large audio tagging system to provide richer conditioning information. Furthermore, Liu and Jin [36] introduced an implicit conditioning method that employs continuous vectors sampled from a distribution to represent and condition the generator on each class, enabling smooth timbre interpolation across audio categories. However, since the conditioning vectors are sampled from a standard Gaussian distribution, they are unbounded, which can make them challenging to control effectively.

Unlike pure generative models, DDSP [17] integrates DSP components directly into an autoencoding network. By doing so, it leverages the structured priors provided by DSP principles, which streamline the training process, while also harnessing the neural network’s ability to accurately fit the timbre of target sounds. However, the original DDSP employs a harmonic-plus-noise synthesizer [37], which is primarily designed for harmonic signals and thus inadequate for impulsive sounds, as shown in [32]. For many sound effects that include narrow-band components or transients, the original time-varying finite impulse response (FIR) filter struggles with the trade-off in time-frequency resolution [33].

Consequently, several approaches have been proposed to extend DDSP for modeling inharmonic and impulsive sound effects. Lundberg [38] explored vehicle engine noise

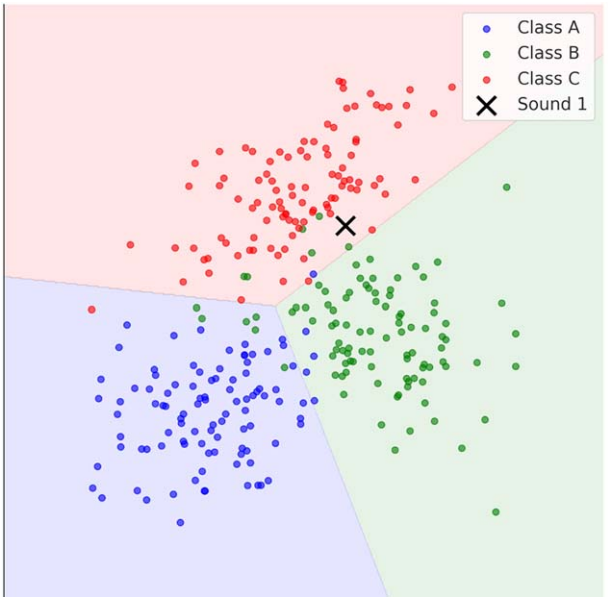


Fig. 1. A *t*-distributed stochastic neighbor embedding visualization of a three-class embedding clusters. Notice that depending on the distance from each cluster's centroid, a sound might possess the sound characteristics of multiple classes. The closer the sound is from a centroid of a class, the higher the chance that it possess the feature associated with that class.

modeling by integrating a transient synthesizer [39] into DDSP. Liu et al. [32] expanded this idea to model rigid-body impact sounds such as gunshots and footsteps. Serrano [40] focused specifically on modeling footsteps using the original time-varying FIR filter, conditioned on a ground reaction force curve. Meanwhile, Diaz et al. [41] developed a differentiable infinite impulse response filter bank for modeling rigid-body sounds conditioned on the shape and material of objects. For a more generalized approach, Barahona-Ríos and Collins [33] introduced a series of multirate filterbanks to synthesize source-agnostic sounds with fine-grained control.

2 METHODS

2.1 Similarity Score

This method for conditioning the generative model draws inspiration from statistical evaluation metrics used in deep generative models. For instance, Fréchet Audio Distance (FAD) [42] employs large pretrained audio representation models, such as VGGish[43] and PANNs [44], to extract meaningful timbre information from both the reference audio dataset and the generated audio. FAD measures the Wasserstein distance between two multivariate Gaussians—one representing real audio embeddings and the other representing generated audio embeddings.

To illustrate, consider a footstep dataset with three categories, each exhibiting a unique timbre, as shown in Fig. 1. For each sound category, embeddings were computed for all audio samples, obtaining the mean and covariance for the category. To extract these embeddings, the widely used Contrastive Language-Audio Pretraining (CLAP) model [45] was leveraged. CLAP was designed for obtaining representations of general audio agnostic to sources, and the model was trained on diverse range of sound effects [46].

Given an audio signal, such as "Sound 1" in Fig. 1, its statistical similarity can be quantified relative to each sound category. Inspired by anomaly detection techniques, where similarity scores are computed to identify outliers, distances were calculated using the Mahalanobis distance (MD) [47]. The MD is a multivariate measure that assesses the distance between a point and a distribution. In this context, for an audio signal x and a group of audio samples characterized by mean μ and covariance matrix Σ, the MD is defined as

$$MD(x) = \sqrt{(x-\mu)^T \Sigma^{-1}(x-\mu)}. \tag{1}$$

Since the calculation of the MD requires the covariance matrix, Σ, to be invertible, Σ was regularized by adding a small value, a process known as diagonal loading: $\Sigma \leftarrow \Sigma + \varepsilon I$, where I is the identity matrix of the same dimension as Σ. This ensures the determinant is nonzero. The computed MD can be interpreted as a measure of similarity to a class (e.g., how similar a gunshot recording is to a balloon explosion).

For a dataset with three categories, the MD for a given sound can be computed relative to all three categories, resulting in a three-dimensional vector, as illustrated in Eq. (2). However, due to variations in intraclass and interclass variances, the MD values can exhibit significant range disparities. To address this, the authors normalized the MD for each channel using min-max normalization, yielding the normalized MD (MD_N):

$$MD_N = \frac{MD - MD_{\min}}{MD_{\max} - MD_{\min}}. \tag{2}$$

The normalized MD falls within the range [0, 1]. A lower MD indicates that the point is closer to the distribution center of the corresponding class, signifying a stronger presence of the timbre associated with that sound category, and vice versa. Using this approach, the authors derived a compact vector that encodes timbre information, which can be intuitively controlled via a fader or knob [48]. When applied to a generative model, this enables the creation of timbral variations.

2.2 Design of the Transient Model

The transient modeling approach is founded on a simple yet effective principle: the inverse discrete cosine transform of a sinusoidal signal corresponds to an exponentially decaying impulse, as introduced in [39]. This technique has been successfully applied to DDSP in combination with harmonic-plus-noise synthesis to model various sounds, such as footsteps [32], vehicle engines [38], and piano sounds [49]. The present approach builds upon DDSP-SFX [32], where the decoder outputs frequency and amplitude values to synthesize sinusoids, which are then converted to impulses in the time domain using the inverse discrete cosine transform. Since multiple transient signals may be required within a single frame of sound, the authors synthesized $f/2$ sinusoids per frame, where f denotes the frame length and acts as a subsampling rate. The synthesized sinusoids were then modulated by an amplitude vector, A,

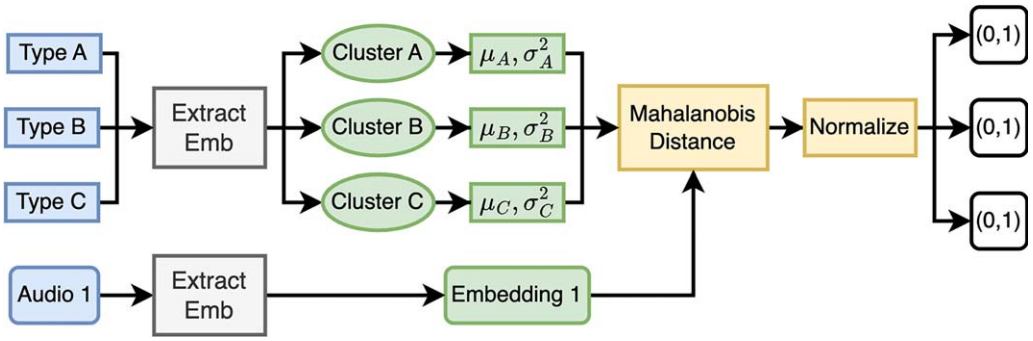


Fig. 2. The process of similarity score extraction. CLAP is used to extract embeddings of each sound. The embedding group of audio class can be treated as a cluster of sounds with certain characteristics. For any audio, the MD can be measured relative to each class. The returned score is normalized across each channel to range [0,1] for easy control.

which is learned from the decoder. As the decoder operates at a lower sampling rate, A was upsampled to the target sampling rate Fs to match the length of the entire signal. This upsampling is achieved using a combination of transposed 1D convolution and rectified linear unit activation. The complete transient model can be formally expressed as

$$x[n] = \sum_{n=0}^{N-1} \sum_{k=1}^{K} A_{n,k}\,\mathrm{IDCT} \int \sin\left(2\pi F_{n,k} \frac{t}{f}\right) dt, \qquad (3)$$

where

- K is the number of sinusoids synthesized per frame and $K = f/2$ in this case.
- $A_{n,k}$ is the amplitude of the kth sinusoid in the nth frame.
- $F_{n,k}$ is the frequency of the kth sinusoid in the nth frame.
- t is the time index.
- f is the frame length.

2.3 Loss Function for the Transient Model

As the FIR-filtered noise synthesizer operates at a lower sampling rate, it may inadequately capture the details of transient energies. NoiseBandNet [33] addresses the trade-off between frequency and time resolution by designing a large-scale filterbank, albeit at the expense of increased computational cost. However, the equidistant placement of transients along the time axis, as proposed in [39, 38], does not fully resolve the issue of missing transient energies.

To better guide transient synthesis, the authors propose incorporating a dedicated loss function. First, the percussive components of the training audio were separated using harmonic-percussive source separation (HPSS) [50], with a kernel size of 31 and a margin of (1, 3). Next, local peak estimation is performed to identify spectral peaks along the time axis, which are then converted into one-hot vectors, with nonpeak regions set to 0. To ensure that synthesized impulses were placed only at these peak locations, the authors created a sparse waveform X_s by multiplying the one-hot peak vectors with the original waveform. Finally, an L_2 loss was computed directly between the transient signal X_t and the sparse peak waveform, providing a more targeted approach for transient synthesis:

$$\mathcal{L}_2 = \|X_\mathrm{t} - X_s\|_2^2 = \sum_n |X_\mathrm{t}[n] - X_s[n]|^2\,, \qquad (4)$$

where n refers to the time-sample index.

2.4 Architecture

Fig. 3 presents the model architecture, which builds upon DDSP. Similar to NoiseBandNet [33], spectral centroid and loudness envelopes were extracted from the input audio, sampled at $F_s = 44{,}100$ Hz, using a window size of $W = 256$. The Loudness envelope provides a general dynamic guidance for the generative model. The spectral centroid, similarly, is another temporal vector that provides an overall "brightness" of the spectrum across time. These features are concatenated and passed into the decoder to generate parameters for the noise and transient synthesizers.

Additionally, a similarity vector was computed by calculating the MD relative to each class, as illustrated in Fig. 2. The similarity vector is an n-channel vector where n represents the total number of classes. (The similarity vector, on the other hand, serves as a global attribute that encodes the actual spectral distributions for sounds with different perceptual characteristics.) This vector was first smoothed using a multilayer perceptron (MLP) layer. Subsequently, feature-wise linear modulation (FILM) [51] was employed to project the similarity vector onto the decoder with a sequence length matching that of the input features. The authors utilized the same decoder architecture as DDSP [17], which is responsible for generating the magnitude frequency response for the noise synthesizer, as well as the amplitude and frequency parameters for the transient synthesizer. The outputs of the noise and transient synthesizers were combined and subsequently processed using the same reverb algorithm as DDSP [17].

3 EXPERIMENTS

3.1 Dataset

This research focused primarily on the generation of impact sounds. Most existing sound effect datasets categorize

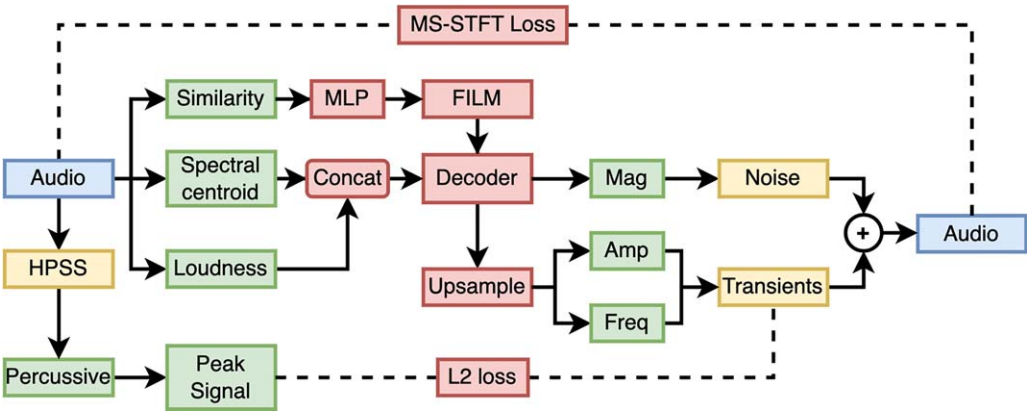


Fig. 3. Overview of the proposed model architecture. Spectral centroid and loudness are extracted explicitly from input audio during the preprocessing stage. Similarity scores are also obtained by measuring the MD between each audio embedding and group audio embedding of each class. A decoder is conditioned on these three features to generate parameters required for noise and transient synthesizers. The synthesized audio is used to calculate the multiscale STFT loss for audio reconstruction. Furthermore, to help the transient synthesizer precisely place the transient energies across the time axis, an L2 loss is computed relative to extracted peaks signal from input audio by performing harmonic-percussive source separation.

Table 1. Number of sounds in footstep and impact categories.

Footsteps		Impacts	
Category	Amount	Category	Amount
All	2,127	All	1,666
Board	83	Smash	153
Concrete	278	Gunshots	280
Gravel	292	Knocks	321
Leaves	378	Explosion	169
Snow	253	Footsteps	528
Squeaky	65	Bounce	155
Stairs	207	Punch	60
Tile	201		
Wood	370		

sounds based on broad classifications (e.g., footsteps, rain, birds), with limited detailed separation within categories (e.g., footsteps recorded on different materials). To address this, 2,127 footstep sounds were sourced from Freesound [52] and categorized into nine groups based on the material of the contacting surface, as detailed in Table 1. The timbral differences across these categories provide a suitable framework for testing the effectiveness of the proposed conditioning variable in controlling the timbre of synthesized sounds.

In addition to the Footstep-set, another 1,666 impact sounds commonly used in games from Freesound were collected. This broader classification allows for evaluation of the effectiveness of the conditioning approach under greater variance across class embeddings. All sounds were sampled at 44.1 kHz with 16-bit depth and a duration of 4 s. The authors acknowledge that audio from Freesound [52] varies greatly in audio quality and label descriptions. Therefore, each audio sample in this dataset was manually curated to ensure it contained only the desired sound, free from interruptions by unwanted sounds or background noise. To verify that each sound matched its designated category, the authors carefully listened to all samples and removed any mismatches. Additionally, silent parts and quiet components (amplitude lower than –60 db for the entire duration) were trimmed. The datasets can be found at https://zenodo.org/records/14286414.

3.2 Training

To set up the training, spectral centroid and loudness features were first extracted from each audio sample, using an fast Fourier transform (FFT) size of 256 and 50% overlap. This process resulted in feature vectors with a sequence length of 690. For the decoder, a hidden size of 512 was used across all experiments. The noise synthesizer was configured with 100 frequency bands. As detailed in SEC. 2.2, 128 frequencies were used for sinusoidal modeling, which corresponds to half the frame size. To prevent aliasing, the frequencies were clipped to the range (0, 128). For the multiscale short-time Fourier transform (STFT) loss [53], spectrograms were computed using FFT sizes of 2,048; 1,024; etc. down to 16, dividing by a factor of two. The model is trained for 5,000 epochs on an NVIDIA RTX 4070 Ti GPU, with a batch size of 16. The ADAM optimizer [54] was used with an initial learning rate of of 1×10^{-4}, which decays to 1×10^{-5} after 80% of the training epochs. Training takes approximately 17 h for the Footstep-set and 13 h for the Impact-set.

3.3 Fine-Tuning Based on Similarity

Since the extracted similarity score is normalized to the range [0, 1], its distribution is not constrained to follow any specific form, unlike in implicit conditioning generative adversarial network (ICGAN) [36]. To better understand its probability density, kernel density estimation was performed as follows:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right), \tag{5}$$

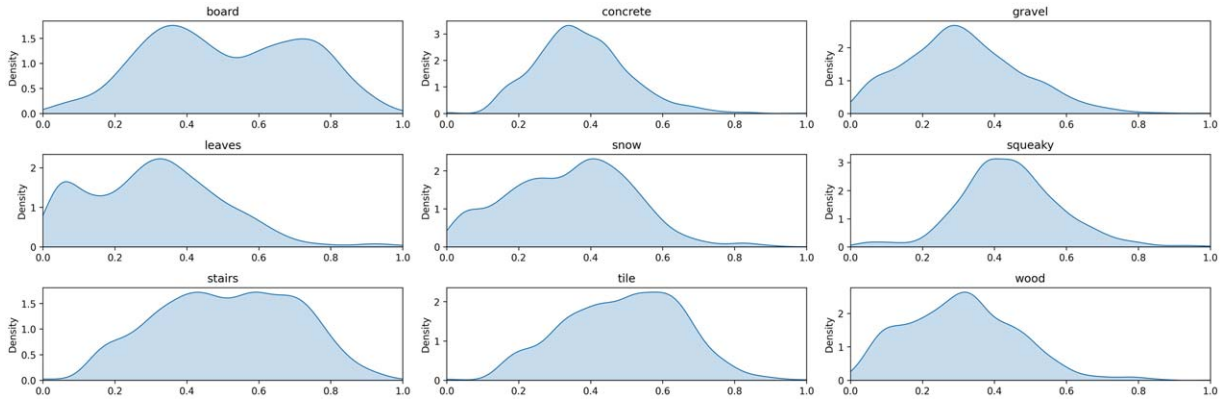


Fig. 4. Kernel density estimation of the distribution of similarity scores of Footstep-set. The horizontal axis represents the similarity scores, ranged from [0,1]. The vertical axis shows the estimated density values. Different distributions of similarity scores were noticed across different categories.

where $\hat{f}(x)$ is the estimated density function at point x, n is the number of data points, x_i are the data points, $K(\cdot)$ is a Gaussian kernel function, $K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}$, and h is the bandwidth, a positive scale factor that affects the smoothness of the density estimation. In Fig. 4, the kernel density estimation results are presented for each category in the footsteps dataset. The plots reveal significant variability in the probability density of similarity scores across classes. For instance, certain ranges of similarity scores (e.g., 0.8–1.0 for concrete) are underutilized during training compared to others (e.g., 0.3–0.5 for concrete). This imbalance can hinder the model's ability to generate diverse timbre variations.

To address this, after the model achieves satisfactory synthesis performance, the similarity conditioning layers were fine-tuned to account for the underutilized ranges. Instead of using the similarity scores extracted from the dataset, pseudo-scores sampled uniformly from the range [0, 1] were input. This ensures that all score regions are captured and utilized for guiding sound synthesis. As illustrated in Fig. 5, during fine-tuning, the decoder layers were frozen, and only the similarity conditioning layers were updated. Since the objective is no longer to improve synthesis quality, the reconstruction multiscale STFT loss was omitted. Instead, fine-tuning was treated as a regression task, minimizing the error between the ground-truth similarity score (sampled from the uniform distribution) and the score measured from the generated audio. The loss function combines L_1 and L_2 losses and is defined as follows:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(s_i - \hat{s}_i)^2 + \frac{1}{n}\sum_{i=1}^{n}|s_i - \hat{s}_i|, \tag{6}$$

where s_i represents the ground-truth similarity score, $\hat{s}_i$ is the predicted score, and n is the batch size. Similar to the training process, a batch size of 16 and a learning rate of 1×10^{-4} with the ADAM optimizer [54] were employed. The fine-tuning is conducted for a total of 10,000 epochs on an NVIDIA 4070 Ti GPU. This process takes approximately 10 h for the Footstep-set and 8 h for the Impact-set (as the Footstep-set contains more samples than the Impact-set).

4 EVALUATION

4.1 Synthesis Performance

The synthesis performance of this method was evaluated against three neural audio synthesis models: the original *DDSP*, on which this model is based (using only subtractive noise synthesis); the *NoiseBandNet* model, which is a variant of DDSP that uses differentiable filterbanks to synthesize sound effects; and *ICGAN* [36], which is another GAN-based sound effects generative model conditioned on implicit soft labels learned from an encoder.

For the *DDSP* model, the same configuration was maintained for its noise synthesizer and training settings as in this approach. This included parameters such as the number of FFTs, multiscale STFT loss, batch size, and training epochs. The model was also guided by spectral centroid and loudness features, extracted in the same manner as in this method, to inform the decoder during sound synthesis.

For the *NoiseBandNet* model, its filterbank was implemented using 2,048 frequency bands, a hidden size of 128, and a window size of 32. The model was trained separately on the two proposed datasets, using 10,000 epochs, a batch size of 16, and a learning rate of 0.001. The *ICGAN* model, which operates in the spectral-temporal domain, requires the extraction of Mel-spectrograms from each audio sample. These spectrograms are computed with a time resolution of 690 and 64 frequency bins. The model is then trained for 10,000 epochs with a batch size of 16 and a learning rate of 1×10^{-4}.

4.1.1 Objective Metrics

To evaluate synthesis performance, two objective metrics were used: FAD [42] and log spectral distance (LSD). FAD is a statistical evaluation metric that measures the overall similarity between two distributions: the reference dataset and reconstructed dataset. In contrast, LSD provides a pairwise comparison between a reference sound and its corresponding generated sound. To generate the reconstructed dataset, the test sets of the datasets were used as input audio. For each test sample, the required audio features were ex-

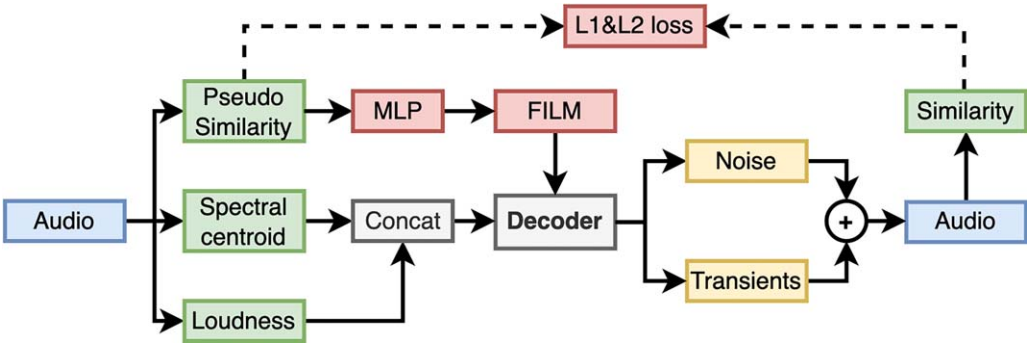


Fig. 5. Fine-tuning process. The weights of the decoder are completely frozen during the fine-tuning stage. For each batch, the authors sampled a vector from a uniform distribution in [0,1] and passed it to the decoder. The similarity score of the generated audio was then measured, and a mixture of L1&L2 loss was evaluated. In this way, the MLP + FILM layers were trained to match the input similarity scores with generated sounds with diverse timbres.

tracted to guide the models and set each model to evaluation mode to synthesize the reconstructed audio. This ensured that every reference sound in the test set was paired with a corresponding generated track from each model. FAD and LSD calculations were then performed using these reference and generated datasets, allowing to objectively assess the synthesis performance of each model.

4.1.2 Subjective Tests

Because similarity-based statistic measurements for audio evaluation are not strictly aligned with human perception, nuanced details might not be sufficiently captured by such objective metrics [55]. To complement the objective metrics, a subjective listening test was conducted. There was a total of 17 participants, six females and 11 males, with ages ranging from 25 to 38. Nine of them were audio experts, and eight of them were nonexperts in audio. Listeners were encouraged to put on headphones to listen to 16 sets of sounds generated from the present model, *DDSP*, *NoiseBandNet*, and *ICGAN*. One sound was randomly selected for each subcategory of Footstep-set and Impact-set from the test dataset (9 + 7). Each participant was asked to report a mean opinion score (MOS) over the sound quality based on a Likert scale from 1 (lowest quality) to 5 (highest quality). In this case, a high-quality sound means it is free from noise, artifacts, and glitches, and it sounds realistic rather than being generated. The stimuli were derived from the same reference sound per subcategory to ensure semantic consistency. To mitigate bias, the stimulus presentation order was randomized across participants.

4.2 Objective Measure of Controllability

This study employed regression analysis, specifically ordinary least squares (OLS) [56], to quantitatively evaluate the effectiveness of the conditioning method used in audio signal synthesis. Regression analysis is a robust statistical technique for quantifying the relationship between independent variables and a dependent variable. The OLS method, a fundamental and widely used form of regression analysis, estimates the parameters of a linear regression model by minimizing the sum of squared residuals. This approach ensures that the best-fit line closely approximates the true data points. In this analysis, the independent variable, c, represents similarity scores interpolated between 0 and 1, indicating the proximity to a specific class A (ranging from close to distant). The dependent variable, y, is the measured similarity of the synthesized sound relative to each class. Ideally, when $c = 0$, the synthesized sound should exhibit a lower distance to class A, indicating high similarity. Conversely, when $c = 1$, the synthesized sound should demonstrate a higher distance, reflecting low similarity. This setup allows for directly evaluating the model's ability to maintain meaningful conditioning during synthesis.

4.3 Regression Analysis Setup

To set up the experiment, variations of each reference sound in the test set were generated by interpolating the similarity score for one channel from 0 to 1 in 100 steps, while keeping all other channels fixed at 1. This resulted in 100 synthesized versions of each sound, all guided by the same reference. To measure the similarity of each output sound relative to each category, the authors opted to abandon the pretrained CLAP model for embedding extraction to minimize potential biases. Instead, pretrained audio neural networks (PANNs) [44], a model extensively employed in audio tagging and sound event detection, was used. A PANNs model pretrained on Audioset [57] was fine-tuned to classify the synthesized sounds into their respective categories. After fine-tuning, the model achieved a classification accuracy of 94.7%. Subsequently, the final layer of the model was removed to use the remaining network as an embedding extractor.

Using these embeddings, the MD of each synthesized sound's embedding was calculated relative to the embedding clusters of each class, following a process similar to the one described in Fig. 2. For consistency and ease of visualization, the MD values for each channel were normalized to the range [0, 1]. Finally, each synthesized sound's similarity metric was paired with its corresponding guiding vector value, creating a dataset with independent variables (interpolated guiding vectors) and dependent variables (normalized MD values). This dataset forms the basis for evaluating

the relationship between the guiding vectors and the resulting synthesized sound similarities.

Using OLS regression, the relationship between the guiding vector values—linearly interpolated from 1 (least similar) to 0 (most similar—and the normalized MDs, which represent the timbral similarity to each target class, was analyzed. OLS was chosen for this analysis because of its ability to provide straightforward parameter estimation by minimizing the sum of squared differences between observed and predicted values in a linear model. Through preliminary analysis, it was observed that the relationship between the guiding vector values and the normalized distances is better described by an exponential rather than a linear trend. To accommodate this, the exponential regression model was transformed into a linear form by taking the natural logarithm of both sides. This transformation allows for performing OLS regression on the modified model:

$$\ln(y) = \ln(a) + b \cdot x, \tag{7}$$

where y is the dependent variable (normalized MD), x is the independent variable, $\ln(a)$ is the transformed coefficient to be estimated as the intercept, and b remains the slope of the regression line. The objective of OLS regression is to minimize the sum of squared residuals:

$$\min_{\ln(a),b} \sum_{i=1}^{n} (\ln(y_i) - (\ln(a) + b \cdot x_i))^2, \tag{8}$$

where n is the number of interpolation steps. After estimating the parameters, the model for predictions is given by

$$y_{\text{pred}} = a \cdot e^{b \cdot x_{\text{pred}}}. \tag{9}$$

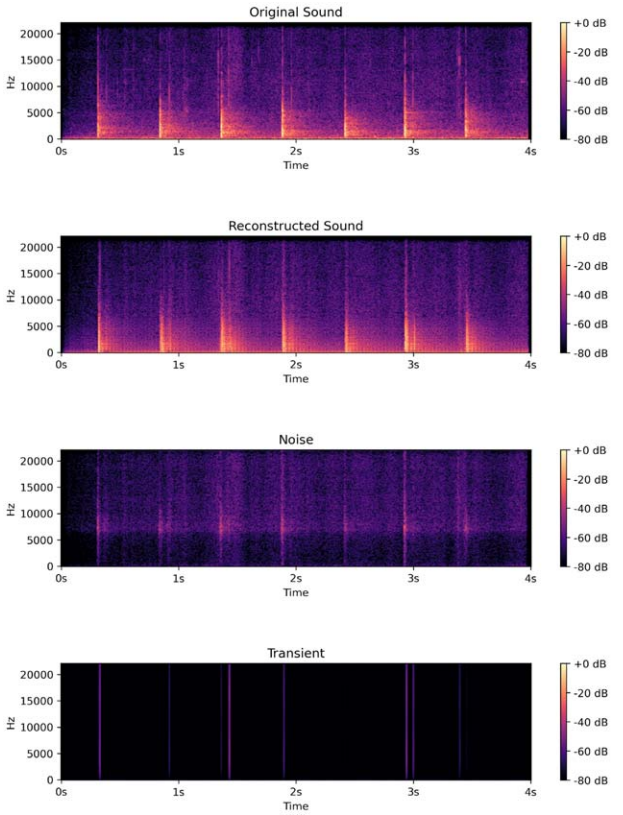


Fig. 6. Sound reconstruction based on a reference track. The spectral centroid, loudness envelope, and similarity scores are extracted from the original sound and used for audio reconstruction. The reconstructed, noise, and transient separations are shown from top to bottom.

5 RESULTS

5.1 Synthesis Performance

5.1.1 Qualitative Analysis

An impactful footstep often contains a sudden burst of energy that is challenging to model solely by filtering white noise through a linear time-varying FIR filter. Fig. 6 presents a randomly selected footstep sound, its reconstructed version using the proposed approach, the synthesized noise component, and the synthesized transient component. The reconstructed sound effectively captures the spectral content of the reference sound.

While the noise component, modeled with the linear time-varying FIR filter (using 256 FFTs), accurately reproduces the frequency spectrum of the reference audio, it falls short in representing the temporal dynamics characteristic of the original sound. This limitation is evident in the lack of temporal sharpness and impact, which are crucial for delivering the precise auditory cues associated with the source material.

To address these shortcomings, transient synthesis (illustrated as vertical impulses with evenly distributed frequencies over a very short duration) is incorporated. This addition enhances the model's capability to reproduce the rapid onset and decay times that contribute to the immediacy and impact of natural sounds. Moreover, by regularizing transient synthesis with an additional L_2 loss based on a sparse, peak-detected signal, the decoder learns to emphasize amplitudes specifically at onset locations while attenuating amplitudes elsewhere. This targeted approach improves the realism and perceptual quality of the synthesized sounds.

5.1.2 Quantitative Analysis

Since the datasets consist of multiple categories, the mean and standard deviations were computed for the entire dataset to ensure robust evaluation. Both the FAD and LSD results demonstrate that the performance of this model is comparable to that of the *NoiseBandNet* model, which employs a large filterbank noise synthesizer with prestored noise bands. Although this architecture uses the same noise synthesizer as the original *DDSP*, the addition of the transient synthesizer significantly enhances the placement of transient energies at a subframe level. This improvement is crucial for reconstructing fine-grained details of impulsive sounds.

The *ICGAN* model performs similarly to *DDSP* but falls short of delivering satisfactory quality compared to the other models. The authors hypothesize that this shortfall may be due to the implicit nature of its conditioning method for encoding class information. This approach introduces

Table 2. Model performance comparison across datasets and measurements.

Models	Footsteps	Impacts
	FAD↓	
Present	6.0 ± 1.39	2.06 ± 0.91
DDSP	10.0 ± 2.39	6.07 ± 3.53
NoiseBandNet	4.06 ± 1.08	1.89 ± 1.04
ICGAN	12.52 ± 4.81	4.92 ± 0.84
	LSD↓	
Present	0.14 ± 0.03	0.11 ± 0.05
DDSP	0.22 ± 0.05	0.30 ± 0.11
NoiseBandNet	0.13 ± 0.02	0.11 ± 0.04
ICGAN	0.20 ± 0.06	0.18 ± 0.08
	MOS↑	
Present	3.08 ± 1.42	3.15 ± 1.26
DDSP	2.76 ± 1.81	2.55 ± 1.08
NoiseBandNet	3.76 ± 1.07	3.54 ± 1.16
ICGAN	1.96 ± 0.58	1.76 ± 0.74

Table 3. The 95% CIs for each model on the Footstep-set and Impact-set.

Model	Footsteps (95% CI)	Impacts (95% CI)
Present	±0.23	±0.23
DDSP	±0.29	±0.19
NoiseBandNet	±0.17	±0.21
ICGAN	±0.09	±0.13

stochasticity into the conditioning process, potentially degrading audio fidelity.

In terms of perceptual evaluation, the subjective scores measured by MOS reveal that *NoiseBandNet* achieves the highest listener ratings across both the Footstep-set (3.76 ± 1.07) and Impact-set (3.54 ± 1.16) datasets. The proposed model also demonstrates strong perceptual quality, with MOS values of 3.08 ± 1.42 and 2.55 ± 1.08, respectively. These results suggest that the transient module in this model contributes meaningfully to perceived realism, outperforming both *DDSP* and *ICGAN*, which received lower scores overall. Notably, *ICGAN* achieved the lowest MOS scores (1.96 ± 0.58, 1.76 ± 0.74), aligning with the hypothesis regarding the limitations of implicit conditioning for audio fidelity.

To further demonstrate the statistical significance among different synthesis models, in Table 3, the authors follow the exact procedure of the MelGAN [58] evaluation to compute the 95% confidence interval (CI) for the MOS as Eq. (10)

$$CI_i = \hat{\mu}_i \pm 1.96 \frac{\hat{\sigma}_i}{\sqrt{N_i}}, \tag{10}$$

where i stands for each model, m_i is the MOS result, N_i is the total number of utterances evaluated for that model, and $\hat{\sigma}_i$ is the sample standard deviation of the per-utterance MOS values for model i. The present model shows slightly wider CIs (±0.23) but still achieves MOS values that lie entirely above those of DDSP and ICGAN, whose intervals do not overlap with this one. In particular, the nonoverlap between the CIs of this model and DDSP/ICGAN confirms that the observed MOS gains are statistically robust. Finally, ICGAN's very tight CIs (±0.09, ±0.13), despite its low MOS, indicate that listeners uniformly judged its fidelity to be poor.

In general, the MOS results are generally consistent with the trends observed in the objective evaluations. Interestingly, it was observed that FAD exhibits more pronounced variation across the two datasets, particularly with a larger gap between models on the Footsteps set. In contrast, both LSD and MOS scores remain relatively stable across datasets. The authors hypothesize that this divergence may be due to FAD's sensitivity to subtle timbral and temporal differences, which are more prevalent in the Footsteps dataset due to its higher acoustic variability and transient complexity. Conversely, LSD primarily captures spectral distortion, which may be less affected by context-dependent nuances. Similarly, MOS may be influenced more by overall perceptual coherence rather than fine-grained fluctuations, resulting in its relative stability across both datasets. This highlights the importance of using complementary evaluation strategies when assessing the quality of synthesized sound effects [55].

5.2 Conditioning Performance

Fig. 8 presents the plots of measured normalized MDs as a function of interpolated similarity scores ranging from 0 to 1. The relationship between the normalized MD of each synthesized sound and its conditioning similarity score closely follows an exponential trend. OLS regression yields a mean R^2 value of 0.4774 for the Footstep-set model and 0.6041 for the Impact-set model, indicating a strong correlation between the independent variable c and the dependent variable y.

The plots reveal that almost all sound categories exhibit clear separations between the presence of a particular feature (at $c = 0$) and its absence (at $c = 1$). This demonstrates that the model successfully learns to output distinct timbres based on the proposed similarity scores. Additionally, the regression lines indicate a positive correlation between the normalized MD and the input similarity scores. This suggests that the conditioning method effectively encodes timbral information unique to different sounds, independent of other acoustic features such as loudness and spectral centroid, which were held constant during this test.

Notably, a clear separation is observed in the similarity score range of 0.8–1.0 across most categories. This implies that values below 0.8 have limited influence on encoding timbre variations, potentially indicating a model bias toward generating outputs with specific modes. To address this limitation, future research could explore strategies to expand the effective range of control. This may involve employing preprocessing techniques to flatten the similarity score distribution or mapping the scores to a more meaningful scale to better utilize the entire range.

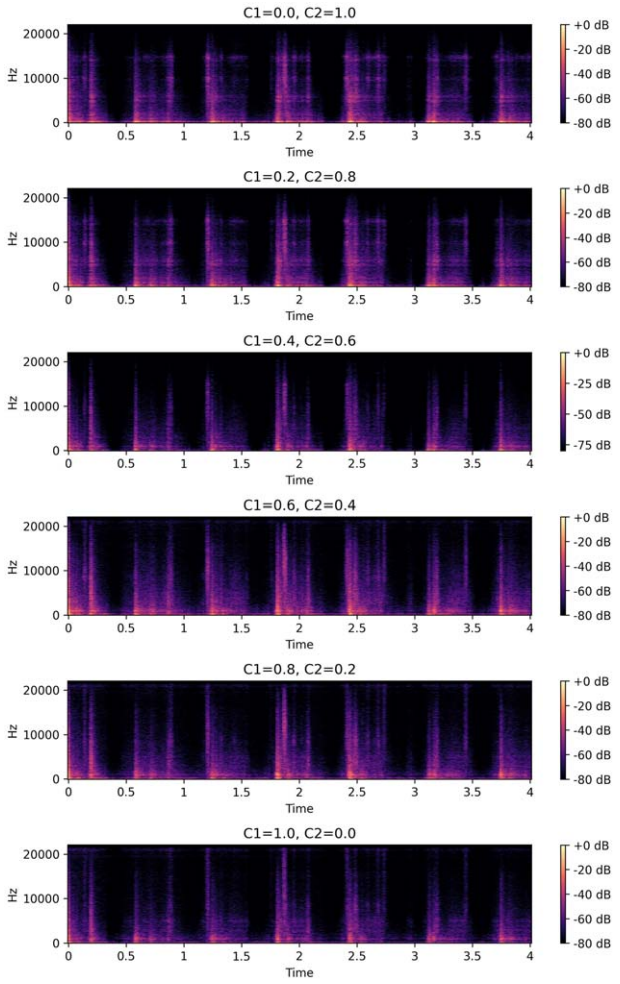


Fig. 7. An example of interpolating between two conditioning similarity scores. All other channels were fixed as 1 for the whole time. The first channel C_1 (Footstep: Board) was interpolated from 0 to 1, and the second channel C_2 (Footstep: Gravel) was interpolated from 1 to 0. A lower conditioning score of C_1 or C_2 indicates a higher similarity with class one or two, respectively.

6 CREATIVE USAGE OF TIMBRE CONTROL

Fig. 7 qualitatively demonstrates how interpolating between two conditioning similarity scores produces distinct timbres. For this example, a sound from the test dataset was randomly selected, and its loudness and spectral centroid were extracted as input. All channels of the similarity score were fixed at 1, except for the first channel (C_1) and the second channel (C_2), which correspond to footsteps on metallic boards and footsteps on gravel, respectively. These channels were interpolated from 0 to 1 (C_1) and 1 to 0 (C_2).

The resulting spectrograms reveal a clear progression in timbre. Initially, the signals exhibit prominent harmonics in the higher frequencies, characteristic of footsteps on metallic boards. As the interpolation progresses, these harmonics gradually transition into noisier signals lacking harmonic structure, which are indicative of footsteps on gravel. For a more detailed visualization and auditory experience, please visit the accompanying website.[1]

7 CONCLUSION

This research proposed a similarity-based conditioning method designed to control timbre variations in sound effects. This method was implemented within a lightweight neural audio synthesis model and evaluated using the authors' custom sound effect dataset, demonstrating its creative potential for sound design. A big motivation of this research is based on the limitation of existing SFX datasets, where descriptive labels are few. Different from conventional generative models that rely on specific architecture design on the latent space [23] or timbre remapping [22] to meaningfully interpolate between two targets, the authors obtained controllable "soft labels" [35] that could be easily interpreted and controlled by novice users by leveraging pretrained audio representation models.

It was shown that this conditioning approach enables the interpolation of subtle sound characteristics unique to different sound categories, providing nuanced control over the synthesis process. The synthesis quality achieved by this method is comparable to state-of-the-art data-driven sound effect synthesis algorithms. Furthermore, regression analysis was conducted to evaluate the relationship between the synthesized sounds and the conditioning vectors. The results indicate that this method effectively separates sound characteristics along the extremes of the conditioning vectors, validating its ability to generate distinct timbral variations.

However, it was observed that the effective range of control is densely concentrated around similarity scores of 0.8–1.0, limiting the method's overall controllability. Future research could focus on improving the distribution of the conditioning similarity scores, ensuring a flatter, more uniform range to enhance the granularity and effectiveness of timbre control.

8 REFERENCES

[1] K. Siedenburg, C. Saitis, S. McAdams, A. N. Popper, and R. R. Fay (Eds.), *Timbre: Acoustics, Perception, and Cognition*, Springer Handbook of Auditory Research (Springer, Cham, Switzerland, 2019).

[2] J. Schouten, "The Perception of Timbre," in *Proceedings of the 6th International Congress on Acoustics*, pp. 35–44 (Tokyo, Japan) (1968 Aug.).

[3] D. Moffat, R. Selfridge, and J. D. Reiss, "Sound Effect Synthesis," in M. Filimowicz (Ed.), *Foundations in Sound Design for Interactive Media*, pp. 274–299 (Routledge, New York, NY, 2019).

[4] L. Turchet, "Footstep Sounds Synthesis: Design, Implementation, and Evaluation of Foot–Floor Interactions, Surface Materials, Shoe Types, and Walkers' Features," *Appl. Acoust.*, vol. 107, pp. 46–68 (2016 Jun.). https://doi.org/10.1016/j.apacoust.2015.05.013.

[1] https://reinliu.github.io/Simi-SFX/.

[5] L. Turchet, S. Serafin, S. Dimitrov, and R. Nordahl, "Physically Based Sound Synthesis and Control of Footsteps Sounds," in *Proceedings of the 13th International Conference on Digital Audio Effects (DAFx)*, paper 50 (Graz, Austria) (2010 Sep.).

[6] A. V. Puche and S. Lee, "Caesynth: Real-Time Timbre Interpolation and Pitch Control With Conditional Autoencoders," in *Proceedings of the IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6 (Gold Coast, Australia) (2021 Oct.). https://doi.org/10.1109/MLSP52302.2021.9596414.

[7] A. Farnell, *Designing Sound* (MIT Press, Cambridge, MA, 2010).

[8] J. Zuiderveld, M. Federici, and E. J. Bekkers, "Towards Lightweight Controllable Audio Synthesis With Conditional Implicit Neural Representations," presented at the *Workshop on Deep Generative Models and Downstream Applications at NeurIPS* (Vancouver, Canada) (2021 Dec.).

[9] P. Esling, N. Masuda, A. Bardet, R. Despres, and A. Chemla-Romeu-Santos, "Flow Synthesizer: Universal Audio Synthesizer Control With Normalizing Flows," *Appl. Sci.*, vol. 10, no. 1, paper 302 (2019 Jan.). https://api.semanticscholar.org/CorpusID:210905230.

[10] N. Jonason, *The Control-Synthesis Approach for Making Expressive and Controllable Neural Music Synthesizers*, Master's thesis, KTH Royal Institute of Technology, Stockholm, Sweden (2020 Oct.).

[11] K. Su, K. Qian, E. Shlizerman, A. Torralba, and C. Gan, "Physics-Driven Diffusion Models for Impact Sound Synthesis From Videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9749–9759 (Vancouver, Canada) (2023 Jun.). https://doi.org/10.1109/CVPR52729.2023.00940.

[12] Y. Liu and C. Jin, "Conditional Sound Effects Generation With Regularized WGAN," in *Proceedings of the 20th Sound and Music Computing Conference*, pp 34–40 (Stockholm, Sweden) (2023 Jun.).

[13] A. Barahona-Ríos and S. Pauletto, "Synthesising Knocking Sound Effects Using Conditional WaveGAN," in *Proceedings of the 17th Sound and Music Computing Conference*, pp 450–456 (Torino, Italy) (2020 Jun.).

[14] H. Liu, Z. Chen, Y. Yuan, et al., "AudioLDM: Text-to-Audio Generation With Latent Diffusion Models," in *Proceedings of the 40th International Conference on Machine Learning*, vol. 202, pp. 21450–21474 (Honolulu, HI) (2023 Jul.).

[15] S. Luo, C. Yan, C. Hu, and H. Zhao, "Diff-Foley: Synchronized Video-to-Audio Synthesis With Latent Diffusion Models," in *Proceedings of the 37th International Conference on Neural Information Processing Systems (NeurIPS)*, paper 2121 (New Orleans, LA) (2023 Dec.). https://api.semanticscholar.org/CorpusID:259309037.

[16] S. Ghose and J. J. Prevost, "FoleyGAN: Visually Guided Generative Adversarial Network-Based Synchronous Sound Generation in Silent Videos," *IEEE Trans. Multimed.*, vol. 25, pp. 4508–4519 (2023 Oct.). https://doi.org/10.1109/TMM.2022.3177894.

[17] J. Engel, L. Hantrakul, C. Gu, and A. Roberts, "DDSP: Differentiable Digital Signal Processing," in *Proceedings of the International Conference on Learning Representations* (Addis Ababa, Ethiopia) (2020 Apr.).

[18] B. Hayes, J. Shier, G. Fazekas, A. Mcpherson, and C. Saitis, "A Review of Differentiable Digital Signal Processing for Music & Speech Synthesis," *arXiv preprint arXiv:rXiv preprint arXiv:2308.154222308.15422* (2023 Aug.).

[19] S. Shan, L. Hantrakul, J. Chen, M. Avent, and D. Trevelyan, "Differentiable Wavetable Synthesis," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4598–4602 (Singapore) (2022 May). https://doi.org/10.1109/ICASSP43922.2022.9746940.

[20] B. Hayes, C. Saitis, and G. Fazekas, "Neural Waveshaping Synthesis," in *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, pp. 254–261 (Online) (2021 Nov.).

[21] X. Wang and J. Yamagishi, "Neural Harmonic-Plus-Noise Waveform Model With Trainable Maximum Voice Frequency for Text-to-Speech Synthesis," in *Proceedings of the 10th ISCA Workshop on Speech Synthesis (SSW 10)*, pp. 1–6 (Vienna, Austria) (2019 Sep.).

[22] J. Shier, C. Saitis, A. Robertson, and A. McPherson, "Real-Time Timbre Remapping With Differentiable DSP," *arXiv preprint arXiv:2407.04547* (2024 Jul.).

[23] J. Alonso and C. Erkut, "Latent Space Explorations of Singing Voice Synthesis Using DDSP," *arXiv preprint arXiv:2103.07197* (2021 Mar.).

[24] E. Grinstein, N. Q. K. Duong, A. Ozerov, and P. Pérez, "Audio Style Transfer," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 586–590 (Calgary, Canada) (2017 Apr.).

[25] C. Steinmetz, N. Bryan, and J. Reiss, "Style Transfer of Audio Effects With Differentiable Signal Processing," *J. Audio Eng. Soc.*, vol. 70, no. 9, pp. 708–721 (2022 Sep.). https://doi.org/10.17743/jaes.2022.0025.

[26] K. Tatar, D. Bisig, and P. Pasquier, "Latent Timbre Synthesis," *Neural Comput. Appl.*, vol. 33, pp. 67–84 (2021 Jan.). https://doi.org/10.1007/s00521-020-05424-2.

[27] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4401–4410 (Salt Lake City, UT) (2018 Jun.).

[28] A. Lavault, A. Roebel, and M. Voiry, "StyleWaveGAN: Style-Based Synthesis of Drum Sounds Using Generative Adversarial Networks for Higher Audio Quality," in *Proceedings of the 30th European Signal Processing Conference (EUSIPCO)*, pp. 234–238 (Belgrade, Serbia) (2022 Aug.). https://doi.org/10.23919/EUSIPCO55093.2022.9909880.

[29] P. Esling, A. Chemla-Romeu-Santos, and A. Bitton, "Generative Timbre Spaces With Variational Audio Synthesis," in *Proceedings of the 21st International Conference on Digital Audio Effects (DAFx)*, pp. 175–181 (Aveiro, Portugal) (2018 Sep.).

[30] N. Devis, N. Demerlé, S. Nabi, D. Genova, and P. Esling, "Continuous Descriptor-Based Control for Deep

Audio Synthesis," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8996–9000 (Rhodes Island, Greece) (2023 Jun.).

[31] J. Engel, C. Resnick, A. Roberts, et al., "Neural Audio Synthesis of Musical Notes With WaveNet Autoencoders," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, pp. 1068–1077 (Sydney, Australia) (2017 Aug.).

[32] Y. Liu, C. Jin, and D. Gunawan, "DDSP-SFX: Acoustically-Guided Sound Effects Generation With Differentiable Digital Signal Processing," in *Proceedings of the 27th International Conference on Digital Audio Effects (DAFx)*, pp. 216–221 (Guildford, UK) (2023 Sep.).

[33] A. Barahona-Ríos and T. Collins, "NoiseBandNet: Controllable Time-Varying Neural Synthesis of Sound Effects Using Filterbanks," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 32, pp. 1573–1585 (2024 Feb.).

[34] M. Comunità, H. Phan, and J. D. Reiss, "Neural Synthesis of Footsteps Sound Effects With Generative Adversarial Networks," presented at the *152nd Convention of the Audio Engineering Society* (2022 May), paper 10583.

[35] J. Nistal, S. Lattner, and G. Richard, "DarkGAN: Exploiting Knowledge Distillation for Comprehensible Audio Synthesis With GANs," in *Proceedings of the 22nd International Society for Music Information Retrieval Conference*, pp. 484–492 (Online) (2021 Nov.).

[36] Y. Liu and C. Jin, "ICGAN: An Implicit Conditioning Method for Interpretable Feature Control of Neural Audio Synthesis," in *Proceedings of the 27th International Conference on Digital Audio Effects (DAFx)*, pp. 73–80 (Guildford, UK) (2024 Sep.).

[37] X. Serra and J. Smith, "Spectral Modeling Synthesis: A Sound Analysis/Synthesis System Based on a Deterministic Plus Stochastic Decomposition," *Comput. Music J.*, vol. 14, no. 4, pp. 12–24 (1990 Jan.). https://doi.org/10.2307/3680788.

[38] A. Lundberg, *Data-Driven Procedural Audio: Procedural Engine Sounds Using Neural Audio Synthesis*, Master's thesis, KTH Royal Institute of Technology, Stockholm, Sweden (2020 Jun.).

[39] T. Verma and T. Meng, "An Analysis/Synthesis Tool for Transient Signals That Allows a Flexible Sines+Transients+Noise Model for Audio," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 6, pp. 3573–3576 (Seattle, WA) (1998 May). https://doi.org/10.1109/ICASSP.1998.679647.

[40] D. Serrano, *A Neural Analysis–Synthesis Approach to Learning Procedural Audio Models*, Master's thesis, New Jersey Institute of Technology, Newark, NJ (2022 Dec.).

[41] R. Diaz, B. Hayes, C. Saitis, G. Fazekas, and M. Sandler, "Rigid-Body Sound Synthesis With Differentiable Modal Resonators," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 11036–11040 (Rhodes Island, Greece) (2023 Jun.). https://doi.org/10.1109/ICASSP49357.2023.10095139.

[42] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fréchet Audio Distance: A Reference-Free Metric for Evaluating Music Enhancement Algorithms," in *Proceedings of the INTERSPEECH*, pp. 2350–2354 (Graz, Austria) (2019 Sep.).

[43] S. Hershey, S. Chaudhuri, D. P. W. Ellis, et al., "CNN Architectures for Large-Scale Audio Classification," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 131–135 (New Orleans, LA) (2016 Mar.).

[44] Q. Kong, Y. Cao, T. Iqbal, et al., "PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 2880–2894 (2020 Nov.).

[45] Y. Wu, K. Chen, T. Zhang, et al., "Large-Scale Contrastive Language-Audio Pretraining With Feature Fusion and Keyword-to-Caption Augmentation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5841–5850 (Rhodes Island, Greece) (2023 Jun.).

[46] Large-scale Artificial Intelligence Open Network (LAION), "LAION-630K: A Dataset of Audio-Text Pairs," https://github.com/LAION-AI/audio-dataset (2022 Jul.).

[47] P. C. Mahalanobis, "Reprint of: Mahalanobis, P.C. (1936) 'On the Generalised Distance in Statistics,'" *Sankhya A*, vol. 80, no. 1, pp. 1–7 (2018 Dec.). https://doi.org/10.1007/s13171-019-00164-5.

[48] G. Lample, N. Zeghidour, N. Usunier, et al., "Fader Networks: Manipulating Images by Sliding Attributes," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, pp. 5969–5978 (Long Beach, CA) (2017 Dec.).

[49] R. Simionato and S. Fasciani, "Sine, Transient, Noise Neural Modeling of Piano Notes," *arXiv preprint arXiv:2409.06513* (2024 Sep.).

[50] J. Driedger, M. Müller, and S. Disch, "Extending Harmonic-Percussive Separation of Audio Signals," in *Proceedings of the 15th International Society for Music Information Retrieval Conference*, pp. 611–616 (Tapei, Taiwan) (2014 Oct.).

[51] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, "FiLM: Visual Reasoning With a General Conditioning Layer," in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence and 30th Innovative Applications of Artificial Intelligence Conference and 8th AAAI Symposium on Educational Advances in Artificial Intelligence*, paper 483 (New Orleans, LA) (2018 Feb.).

[52] F. Font, G. Roma, and X. Serra, "Freesound Technical Demo," in *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 411–412 (Barcelona, Spain) (2013 Oct.).

[53] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel WaveGAN: A Fast Waveform Generation Model Based on Generative Adversarial Networks With Multi-Resolution Spectrogram," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6199–6203 (Barcelona, Spain) (2020 May).

[54] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proceedings of the 3rd Inter-*

national Conference on Learning Representations (ICLR) (San Diego, CA) (2015 May).

[55] P. Manocha, Z. Jin, and A. Finkelstein, "Audio Similarity is Unreliable as a Proxy for Audio Quality," Proc. Interspeech 2022, 3553–3557, doi:10.21437/Interspeech.2022-405.

[56] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis* (Wiley, Hoboken, NJ, 2012), 5th ed.

[57] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, et al., "Audio Set: An Ontology and Human-Labeled Dataset for Audio Events," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 776–780 (New Orleans, LA) (2017 Mar.).

[58] K. Kumar, R. Kumar, T. de Boissière, et al., "MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis," in *Proceedings of the 33rd Neural Information Processing Systems (NeurIPS)*, paper 1335 (Vancouver, Canada) (2019 Dec.).

A.1 VISUALIZATION OF CONDITIONING PERFORMANCE

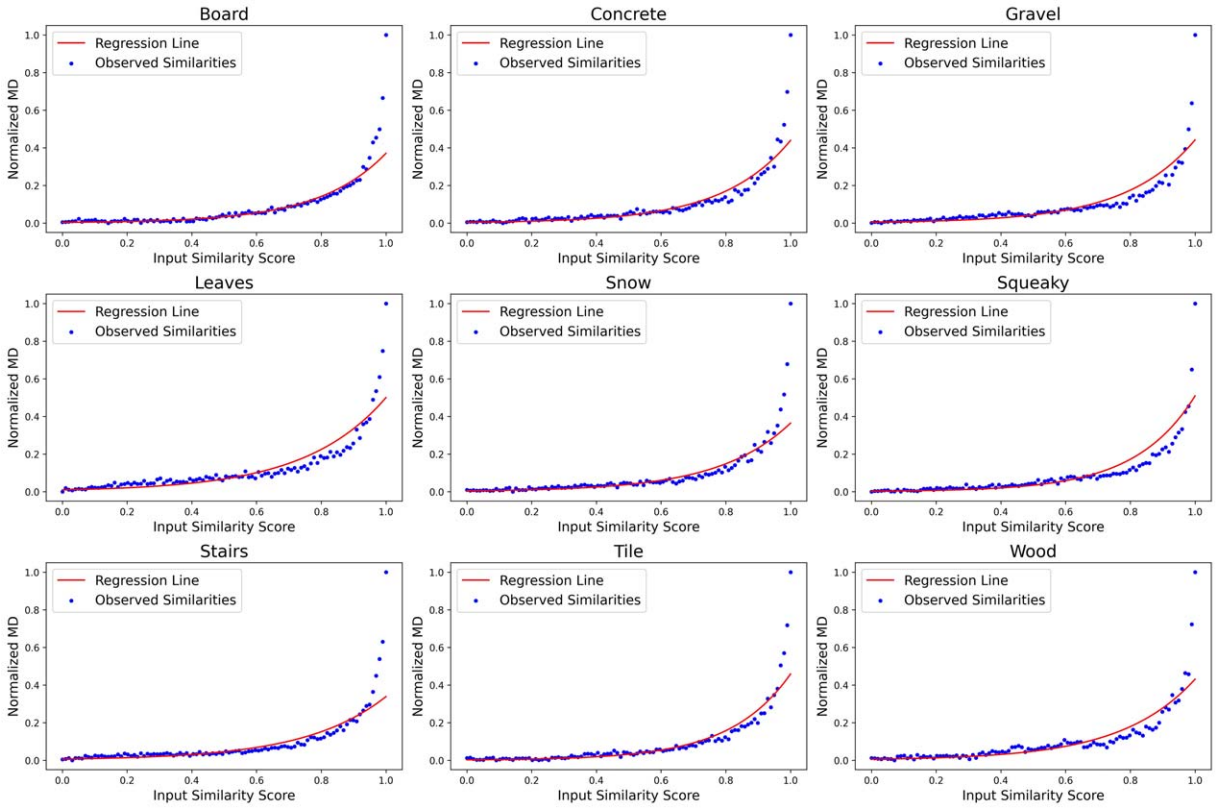


Fig. 8. Regression analysis for the model trained on Footstep-set.

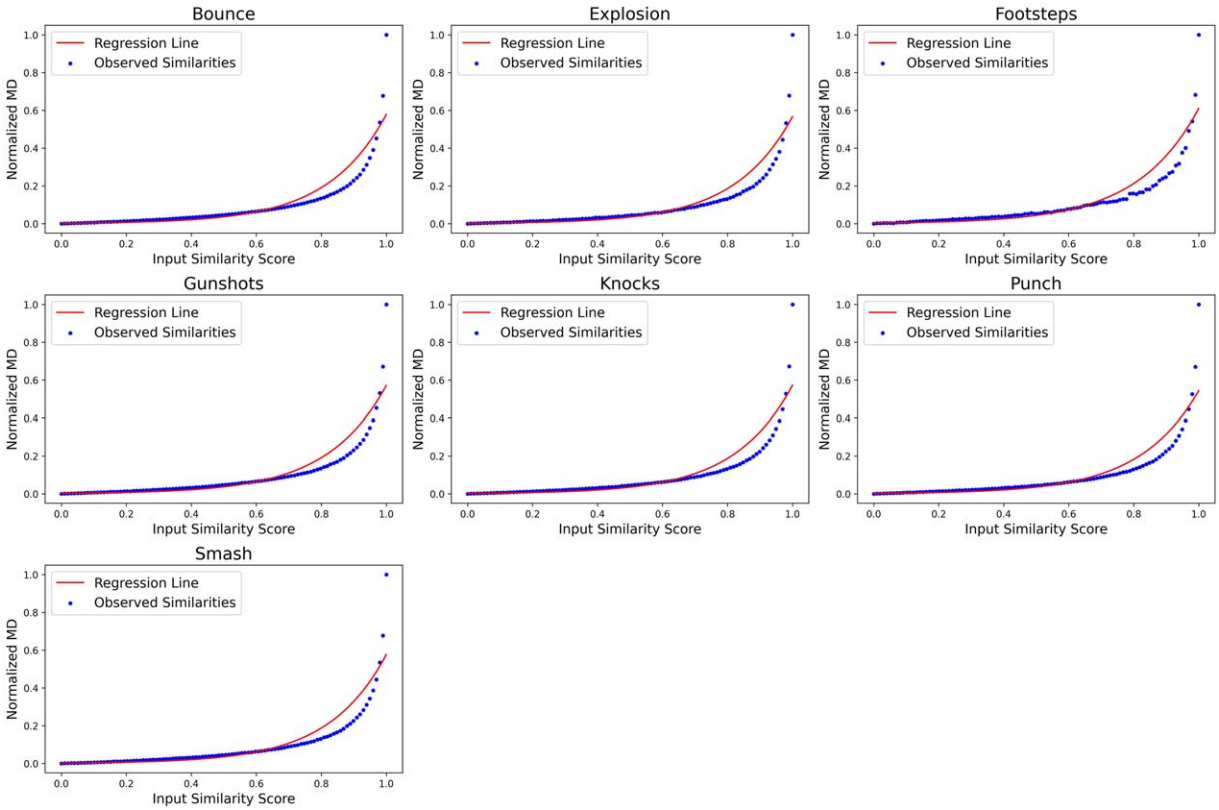
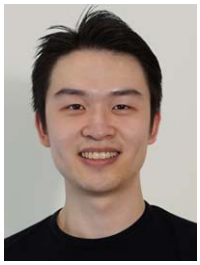


Fig. 9. Regression analysis for the model trained on Impact-set.

THE AUTHORS



Yunyi Liu

Yunyi Liu is a Ph.D. student at University of Sydney in Electrical and Information Engineering school, Sydney, Australia. He received a Bachelor of Music and Sound Design degree from the University of Technology Sydney and a Master of Interaction Design and Electronic Arts degree from the University of Sydney in 2019 and 2021, respectively. His Ph.D. topic is "interpretable and controllable sound effects generation with neural audio synthesis." He has been doing research on conditioning methods for audio generative models with a focus of unsupervised learning by extracting meaningful information from limited datasets. His research interests include neural audio synthesis, controllable and interpretable generative models, audio signal processing, machine learning, self-supervised learning, and cross-modal sound synthesis.

•



Craig Jin

Craig Jin received an M.S. degree in applied physics from the California Institute of Technology, Pasadena, California, in 1991 and a Ph.D. in electrical engineering from the University of Sydney, Sydney, Australia, in 2001. He is an Associate Professor in the School of Electrical and Computer Engineering, University of Sydney, and Director of the Computing and Audio Research Laboratory. His research interest includes experimental and theoretical aspects of acoustic and biomedical signal processing, with a recent focus on context-aware computing, assistive technologies, and auditory sensory augmentation. He has authored more than 240 papers in these research areas, holds eight patents, and founded three start-up companies. He received national recognition in Australia (April 2005, Science in Public Fresh Innovators Program) for his invention of a spatial hearing aid.