An Adaptive Musical Vibrotactile System (MuViT) for Modern Smartphones

Tianqiang Yan[1](✉) and Yuwei Liu[2]

[1] The Chinese University of Hong Kong, Shenzhen, Shenzhen, China
222010015@link.cuhk.edu.cn
[2] Nanjing University of Science and Technology, Nanjing, China
Liuyw@njust.edu.cn

Abstract. Multimedia is a must-have for almost every person in daily life nowadays, but the truth is that most of the existing ways of enjoying multimedia sources are limited to listening to music or watching videos. In pursuit of the next-level experience, multimodal or multisensory entertainment has arisen in recent years, yet most of which involves special equipment and are designed for extremely limited scenarios that makes them hard to reach the populace. In this article, we propose a software-based, device-independent way to interact with audio contents on the smartphones involving another human sense besides hearing and vision, which is tactile sensation. We provide a system prototype, called MuViT, an adaptive musical vibrotactile system that allows users to feel the vibrations only by holding their phones while playing multimedia contents. The system is designed as a mobile application so that users can easily obtain and use it, and it's designed to adapt most of the popular multimedia applications found in apps market, such as YouTube and Apple Music. Furthermore, the system is fully open source on GitHub, so that interested individuals are welcomed to further tap its potential.

Keywords: Linear motor · Vibrotactile · Haptics · Onset detection · Onset event · Subjective evaluation

1 Introduction

As smartphones are updated and popularized at tremendous speed under the promotion of manufacturers and operators, the advancement of software ecosystem has been a heavy demand, and multimedia technologies and contents are a matter of great concern. In addition to ensuring visual and auditory needs, research and development on multimodal interactivity is blooming these years.

Haptics is an important dimension giving information as a position and force. For example, when people learn how to play musical instruments, not only visual and auditory feedback, but also tactile feedback from the instruments is momentous. Tactile information integrated into existing audio-visual scenarios can bring more immersive experience to human, such as movie experience [1], Virtual Reality [2, 3], music learning [3, 4] and accessibility for impaired listeners [4, 5]. Generally speaking, haptic

interaction can make the audience feel the power of music and increase the reality for the auditory experience. Accordingly, we believe that multimodal perception including haptics has great potential to improve multimedia interaction experience.

In recent years, a growing corpus of studies explores how to use haptic stimulation to enhance the experience related to audio. Along the same lines, a broad range of musical haptics systems have been proposed in the literatures. Some of these systems are embedded on fixed equipment while listening to music or watching videos with soundtracks. For example, L. Hayes in [6] created an audio haptic work named Skin Music. By lying on a piece of bespoke furniture, the listener perceives the music both through the usual auditory channels, as well as by different types of haptic sensation, through their body. However, this is not a portable project. In addition, the project was optimized for just one special music, which is of no practical significance. Other proposed techniques rely on wearable accessories. Mazzoni et al. in [1] developed a wearable prototype system named Mood Glove to annotate the emotions of movie soundtracks through haptic sensation, thus enriching the viewing experience. Hwang et al. in [7] developed a haptic music player for mobile devices. They used two types of actuators attached to the handheld model (LRA and DMA), among which the DMA can generate vibrations composed of two main frequencies, which can lead to greater diversity in vibrotactile perception. The truth is that, the above-mentioned systems are all divided structures, which makes them hard to be integrated into daily life. To achieve a high-quality overall experience, we believe higher integration, or an all-in-one software platform, is more capable of creating seamless and on-the-go experience. Smartphones are the most widely used mobile devices, and their potential for haptic usability has been explored continuously.

Several attempts have been introduced based on similar concepts. QQ Music carries out a sound effect called Super-Hyped DJ which extracts upbeats and downbeats through artificial intelligence. Then it generates templates suitable for tracks according to the beat features. Super-Hyped DJ combines with the sound-track to produce rhythmic vibration and flash effects, enhancing the atmosphere for the users especially when someone is holding a party and playing some music. Although it is capable of fully-automatic vibrotactile generation based on audio features, the actual experience is not so desirable, and it only adapts to a fraction of songs. SONY also presents a dedicated Digital Vibration System (DVS) for its mobile phones, the Xperia series. The software and hardware operate in coordination, which can adaptively detect beat features and match vibration signals with great precision and elaboration. Nevertheless, it is a closed source system optimized solely for SONY's flagship phones and cannot be applied to others.

Driven by the same idea, we introduce an Adaptive Musical Vibrotactile System (MuViT), an upper-level software-based musical haptics extension compatible to any smartphones. Our scheme has the following advantages. First of all, we design an adaptive algorithm to extract the low-frequency features of music, which does not require manual settings for specific contents. The system can realize the real-time output of vibrotactile effect, producing vibration signals directly from audio sources without prior knowledge of the entire soundtrack, also protecting the user privacy. Furthermore, our vibration generation algorithms satisfy the functional requirements using digital signal processing techniques with very low computational complexity, while the vibration of

the motor is generated jointly by two different control functions, which can still lead to delicate vibrotactile perception. More importantly, the system extends its usability to even more scenes, mobile gaming in particular. Meanwhile, we designed an application for MuViT, which is an open-source program for Android smartphones, and users can freely adjust the parameters to achieve better haptic experience according to their hardware environment.

Our research exploits the usability of haptics on arbitrary smartphone models. In the second section, the MuViT architecture is discussed in detail and by part. To further verify the functionality and performance, we proposed a subjective evaluation for the system and received reasonable feedback in our offline surveys, the results and analysis of which is delivered in Sect. 3. Section 4 is a conclusion of the paper.

2 System Description

The MuViT is designed fully based on subjective evaluations from real-world smartphone users. Therefore, some of the given parameters and algorithms are based on our experience gained during testing and may be inexplicable. However, the entire system is not designed to be a black box. Low-complexity signal processing and analysis methods are utilized to make the system mechanism as simple and interpretable as possible, and most of the values and expressions are customizable in some degree, the influence of which on our system as well as the user's somatosensory experience can be explained. As most of the existing technical means are still not capable enough to achieve full adaptation, we believe that appropriate system transparency and customizability should be left for users and further optimization. Generally speaking, the concept of this system is mainly delivered throughout the paper, and all technical details in this section are for annotating our implementation of this concept.

The following content of this section will be divided into two parts. The overall architecture will be introduced in the first part, while the second part decomposes the system process, and key technicalities are discussed.

2.1 Introduction to System Architecture

The following image (see Fig. 1) shows the overall architecture of MuViT.

MuViT does not require any peripheral equipment and is fully and well functioned with just a single smartphone. Compared with traditional smartphones equipped with rotator motors, the popularity of linear motors has driven the texture of haptic feedback on modern smartphones to a brand-new level, enough to set users free from sundry gadgets or specially-designed environments [1, 5–7] that generate vibration for audio signals. Besides that, we've seen more and more manufacturers paying attention to the vibration experience of smartphones, and integrating more advanced haptics tuning algorithms and stronger third-party application adaptability into the newly-launched models or the system upgrading of relatively old models. The mobile linear motors are generally divided into Z-axis (vibrate vertically) and X/Y axis (vibrate horizontally). The Z-axis linear motor was put into use earlier, possessing the advantages of occupying less interior space and lower cost. However, compared with the horizontal linear motor,

which has two symmetrical spring coils, the Z-axis motor integrates only one lower-side spring, which enables the horizontal type to have a more delicate touch feedback experience. Due to the fact that the above two types of linear motors can be found in the market today, the parameters of MuViT are optimized for both of them. The respective settings will be given later.
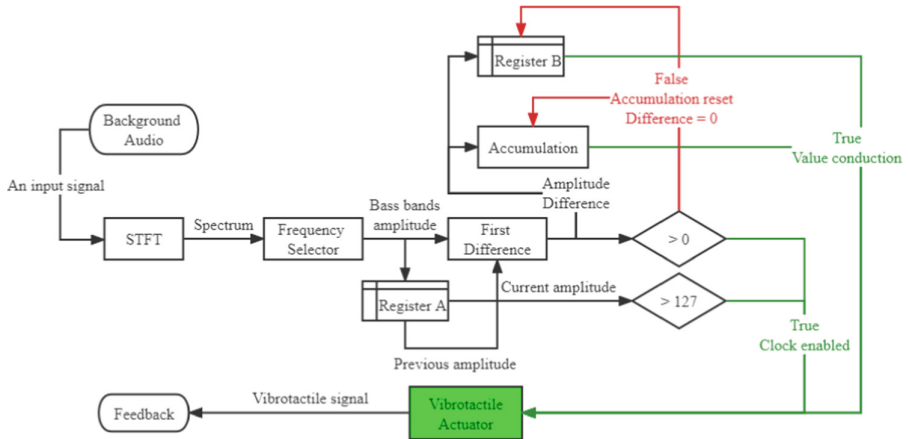

Fig. 1. The schematic flow of MuViT

The complete mechanism involves as few complicated calculations and delays as possible to meet satisfactory real-time performance. In addition, in the following analysis, you will see that MuViT requires low energy consumption on our test devices. When the audio signal is detected and obtained, short-time Fourier transform (STFT) is performed to convert discrete-time signals (DTS) to spectrums in real time. A frequency selector is utilized to filter out high frequency signals, and the output is a value ranging from 0 to 255, proportional to the instantaneous strength of the retained bass bands. Then, the time-domain first backward difference of the current amplitude is calculated in order to find the onset point and its strength. A dual-channel threshold is connected behind to let pass the high-enough amplitude with nonzero difference. When both the amplitude and the difference value meet the conditions for passing the threshold, the difference value is accumulated to compute the initial vibration intensity, the clock is enabled, and all required values are transmitted to the vibrotactile actuator, where two major functions are embedded for producing haptic signals. Otherwise, the clock is disabled and no tactile feedback is output. Suppose a vibration behavior is ready to be evoked, the clock driver function dynamically adjusts the clock interval before the clock starts. The feedback, delivered as the timer ticks, the strength of which is manipulated by a vibrotactile control function with the aim of providing sufficiently delicate texture.

The whole process is jointly realized and tested on two mobile devices, of which the specification list is shown in Table 1. Please note that the two models are equipped with two different types of linear motors.

An Android application, namely Viv Beats, is the instance of our MuViT system. The application does not perform as an audio player. Instead, it monitors the audio played in

Table 1. The specification list of smartphones for development and testing

Model name	Processor	RAM	Battery size	Motor type
Huawei Mate 20 Pro	HiSilicon Kirin 980	8 GB	4200 mAh	Z-Axis Linear
Sony Xperia XZ2	Qualcomm SDM845	4 GB	3180 mAh	X-Axis Linear

the background and applies the MuViT model to generate vibrotactile feedback based on the retrieved audio signals in real time. Figure 2 displays a screenshot of the application running in the foreground together with a popular multimedia application playing in the background.
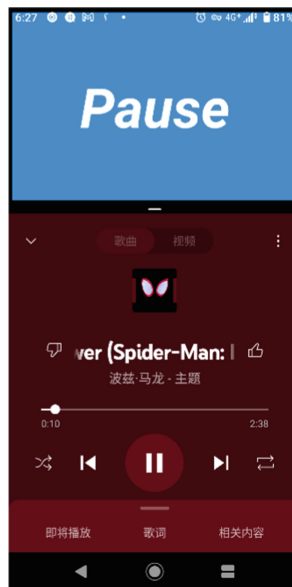

Fig. 2. As is shown, the Viv Beats application is running in the split-screen mode in the foreground, and the audio source is being fed by YouTube Music, a widely-used multimedia streaming platform on Android.

The application is developed using an online Android development platform called Kodular. All the source file as well as the APK file (installation package) can be found and downloaded on its exclusive GitHub repository [8]. Table 2 demonstrates the basic information (size, permissions, etc.) of Viv Beats.

2.2 System Disassembly

Short-Time Fourier Transform (STFT). The Fourier transform only reflects the characteristics of the signal in the frequency domain, it cannot analyze the signal in the time

Table 2. Basic information of Viv Beats

Platform	API level	APK size	Storage required	Permissions
Android	29 (Android 10)	5.52 MB	At least 18 MB	Recording

domain. Therefore, the Fourier transform is only suitable for processing stationary signals. The frequency characteristics for non-stationary signals vary with time, in order to capture this time-varying characteristics, we need to perform time-frequency analysis of the signal. Gabor proposed the short-time Fourier transform (STFT) in 1946, which is essentially a windowed Fourier transform.

The process of STFT is as follows: before the Fourier transform, the signal is multiplied by a time-limited window function. The non-stationary signal is assumed to be stationary in the short interval of the analysis window subsequently. Through the movement of window function $h(t)$ on the time axis, the signal is analyzed segment by segment to obtain a set of spectrums of the signal.

The short-time Fourier transform of a signal is defined as:

$$STFT(t, f) = \int_{-\infty}^{\infty} x(\tau)h(\tau - t)e^{-j2\pi f\tau}d\tau \tag{1}$$

The short-time Fourier transform of the signal $x(t)$ at time t is equal to the Fourier transform made by multiplying the signal with an "analysis window" $h(\tau - t)$.

Frequency Selector. The frequency selector outputs the average amplitude of low frequency segment, i.e. the bass bands, since a significant increase in the strength of low band can usually be observed when it comes to a moment or a period with strong impact in a piece of music, mostly generated by percussion instrument. Hwang et al. in [7] separate bass and treble with a 200 Hz boundary. However, Merchel et al. [9] pointed out that pseudo-onsets are likely to be detected and converted into false vibration patterns when the dividing line is set to 200 Hz, and they further suggest a 100 Hz low-pass filter for broadband impulsive events.

Due to the limited frequency-domain sampling points with approximately 43 Hz spacing, 86 Hz (corresponding to 100 Hz), 129 Hz, and 172 Hz (corresponding to 200 Hz) are selected as the upper bound of the frequency selector respectively for testing. There is no need to examine even higher upper bounds for highly-noticeable spurious feedback ruining the experience. Eventually, our subjective assessment result indicates that the 129 Hz frequency limit can help the system achieve relatively first-rate experience in overall situations.

After the frequency selector finished its job, a value showing the present amplitude of bass bands, ranging from 0 to 255, is stored each round for later use.

Time-Domain First Difference. When the instantaneous amplitude of bass bands is acquired, the time-domain first difference plays the role of a preparation for the upcoming

onset detection. The previous deposited amplitude value is regained to complete the calculation, which is given by (2).

$$\Delta \hat{y}[k, t_n] = \hat{y}[k, t_n] - \hat{y}[k, t_{n-1}] \tag{2}$$

Note that this is a backward difference. Here, $\hat{y}$ stands for the amplitude of bass bands (from 0 Hz to 129 Hz. k equals 129) sampled at the t_n-th time. We have also evaluated the effect of replacing this step with second difference, as second difference are widely used for extracting delicate edge features in the spatial domain of an image. A one-dimensional edge sharpening operator given by (3) was designed, in an attempt to depict subtle onset features.

$$\Delta\left(\Delta \hat{y}[k, t_n]\right) = \hat{y}[k, t_n] - 2\hat{y}[k, t_{n-1}] + \hat{y}[k, t_{n-2}] \tag{3}$$

Unfortunately, it turns out to be disappointing. As is shown in Fig. 3, the operator catches fallback signals while still returning positive values sometimes which are useless for producing vibration but are hard to filter within a few steps, and the exact opposite situation can happen either. In addition, compared with the first difference, the operation of second difference needs to wait for two additional input signals after the system starts to calculate the first value and output a vibration signal. That is to say, enabling second difference will waste about more than twice the initial time as the first difference, losing more onset information if an onset event happens to be located at the beginning.
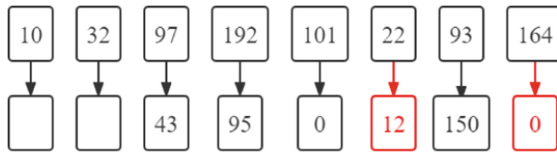

Fig. 3. As it's seen, some of the results of second backward difference are faulty, which are marked red in the picture. The '22-to-12' is a typical pseudo-onset, while the '164-to-0' is a wrongly-ignored onset point. (Color figure online)

Every output difference value in each process is summated, by which the envelope of the time-variant bass intensity level can be characterized. Yet not all features of such an envelope are demanded for perceiving and portraying remarkable onset events. More specifically, a noteworthy onset event contains a prominent signal take-off and, in some cases, a slope (usually rough) behind, reaching the local peak. Thus, it is essential to sort out an approach to determine whether the difference value should be reserved, and whether the accumulation is allowed. For this purpose, we introduced a dual-channel threshold to solve the problem.

Dual-Channel Threshold. The dual-channel threshold identifies the onset initiation point to the nearest onset peak as an onset event. It has two inputs, including the current bass amplitude and its first difference, each of which corresponds to a judging condition, namely a threshold. Figure 4 demonstrates the internal mechanism.

For the bass amplitude, values that are excessively low can be ignored, especially when the difference result of which is positive. A low-amplitude signal with positive onset strength is, in most cases as we've tested, a tactile disturbance when it comes to vibration formation. The threshold for bass amplitude is set to 127 (max. 255), an empiric value which facilitates relatively better haptic experience comparing to other alternatives examined.

For the difference value, the result of which is negative when a decline in bass intensity is detected. Such negative results make no more use for further steps, so they are set to zero after passing through the threshold. Likewise, declines are needless for the summation of differences, as summations are operated only when an onset event is being portrayed, the definition of which is aforementioned. Under such circumstances, the accumulation process should also be reset and remain inactive until the next onset point approaches.
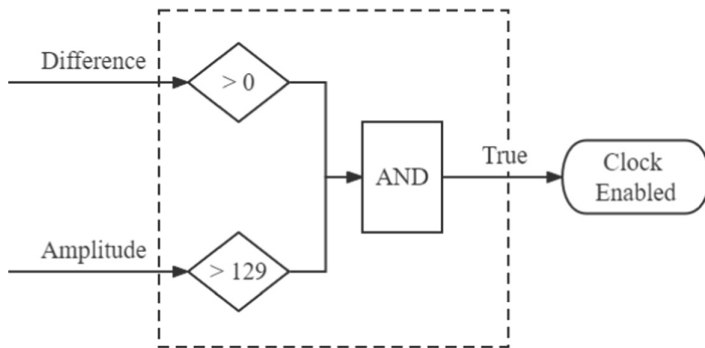

Fig. 4. The dual-channel threshold mainly acts as a conditional switch of the clock. Here in this image, the value transfer operations as well as the false condition are not drawn for involving too many external components.

The clock is used to actuate motor at appropriate intervals. By default, the clock is off, and remains off until inputs in both channels are greater than their corresponding thresholds. When the conditions are satisfied, the inputs are retained and are ready to be delivered into two key functions that drive the vibrotactile feedback on the basis of audio signals. The two functions are described right after.

Clock Driver Function and Vibrotactile Control Function. Delivering vivid and accurate vibration feedback is no easy job. Depending on these two functions, we try to provide tactile experience as excellent as possible. The combination of them matches the proper vibration frequency and intensity for each onset event in no time. Since whether the feedback texture is pleasant or not is decided by users, the two functions are designed mainly based on subjective experience, yet the parameters presented in the functions are mostly tunable, and the logic behind is explicated hereinafter.

Clock Driver Function. The clock driver function (4) allocates the length of intervals for the clock in real time, which determines the frequencies of vibration outputs. It

is a composite function consisting of a constant coefficient and an exponential decay function.

$$t_{interval} = t_c \cdot a^{-s} \tag{4}$$

In the expression, t_{interval} represents the clock interval in milliseconds. t_c is a constant, which stands for the maximum clock interval value. a is an attenuation factor greater than 1, deciding the attenuation rate of the curve. s is the dependent variable in this function, equal to the current difference which is always a positive value. As difference s becomes larger, t_{interval} decays along the curve. The experience values of t_c and a are 280 and 1.06 respectively, which fit both two types of linear motors. Higher t_c requires more powerful onset to drive the clock within a single sampling period, allowing the system to better resist forged onsets, while resulting in greater attenuation rate and possibility of omitting effective strikes, so it should be adjusted with caution. To mitigate the excessive attenuation of the interval with the increase of the onset strength, appropriately lowering the value of a is recommended. Moreover, we suggest that after tuning the above parameters, the interval can be less than or equal to the minimum interval acceptable when s exactly reaches a high-enough value. It can be utilized to check if the attenuation rate is suitable or not. This high-enough value in our test platform is set between 70 to 80, and the maximum value 255, based on subjective sensation, is generally not suitable.

Vibrotactile Control Function. The basic idea behind the vibrotactile control function is to offer vibration feedback whose strength is proportional to the current impact of bass bands while providing a sense of damped oscillation during the sampling gap. The expression of the function is presented in (5):

$$A = A_{init} \cdot e^{-\left(\frac{\alpha}{255} \sum \Delta\right)^2 t} \tag{5}$$

In (5), A_{init} is the initial amplitude of vibrotactile signal transferred to the palm. It is a compound variable, the expansion of which is shown in (6) where A_{max} is the maximum vibration amplitude. α is the vibration attenuation factor that helps compensate or weaken the damping ratio. $\sum \Delta$ is the difference cumulative sum representing the current cumulative impact. t is the major argument of A. It changes with the clock.

$$A_{\text{init}} = \frac{A_{\text{max}}}{255} \sum \Delta \tag{6}$$

As it's shown in (5) and (6), the value of $\sum \Delta$ determines the initial amplitude of the vibration feedback, and as t grows larger, amplitude A attenuates. In the next paragraph, an example is given and described in order to provide a more intuitive impression of a complete vibrotactile event based on a typical onset event.

An onset event is made up of at least one onset point, the duration of which is a multiple of the signal sampling interval. Consider encountering an onset event formed by an initial onset point and a series of gradually rising signals before the upcoming peak which is also captured and contained, while the initial point has the largest difference compared with its followers, a period of high-frequency vibration feedback are delivered at the very beginning.
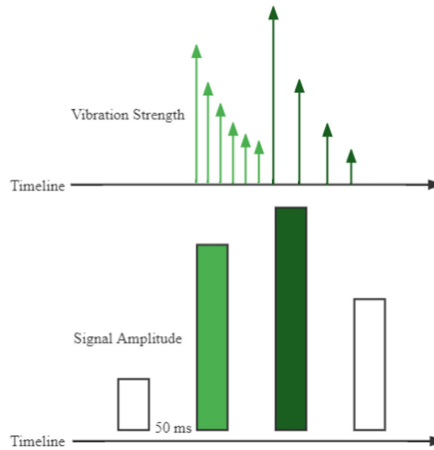
Fig. 5. This image demonstrates a vibration event synchronized with an onset event including two onset signals. The light green arrows constitute the feedback flow of the light green onset, while the dark green arrows form the feedback flow of the dark green onset. (Color figure online)

As the clock ticks within a sampling interval, the vibration amplitude decays along curve *A*, which is the same with the tactile significance. For the next signal, the clock is reset as soon as it arrives. According to what we've defined and functions (4)–(6), this signal has a relatively smaller difference but greater amplitude compared with the last one, which delivers more powerful initial vibration strength and lower feedback frequency. In addition, according to (5), the damping of vibration power should be comparatively faster. The above process lasts until the peak signal finally passes. The difference of the next signal is supposed to be negative, and the clock returns to standby mode, no vibrotactile event is to be triggered before the next onset event arrives. Figure 5 illustrates a vibrotactile event from its beginning to the end.

3 Evaluation Framework, Results and Analysis

3.1 Framework

We designed a HAPTIC framework with six subjective metrics (H, A, P, T, I, C) to evaluate the MuViT system based on offline survey results. First four of the metrics are listed as follows: Happiness (H)—The degree of the extra pleasure provided by the sensory extension; Adoption (A)—User's adaptability to such a novel function; Precision (P)—The onset detection accuracy reflected by actual multisensory impression, and Touch Sensation (T)—The overall comfort level of vibrotactile feedback during one's evaluation process regardless of Precision P. These four dimensions of metrics are graded in A (90–100), B (75–89), C (60–74), and D (0–59). Of the remaining metrics, Interaction (I) appraises the significance of the system, i.e. the willingness to use it in daily life, which is a binary option of 'yes' or 'no'. Compatibility (C) evaluates the compatibility ratio of MuViT to the most popular multimedia interactive platforms on Android, and the score is expressed in terms of a percentage. It is the only objective metric and is

removed from our real-world assessment (HAPTI), the score of which is later obtained by ourselves.

Five CD-quality music clips of five music genres are chosen as the testing soundtracks, the details of which are shown in Table 3. Sixteen volunteers are selected randomly to participate in the HAPTI survey. Two versions of Viv Beats with minor configuration differences are installed separately on our two testing smartphones (as shown in Table 1), and the volunteers are divided into two groups, with eight testers in each, to partake in the same hands-on session and evaluation process on the two devices respectively. At the end of the hands-on session, participants are required to grade the MuViT system according to the HAPTI metrics and their subjective feeling. A questionnaire is designed for the grading procedure.

Table 3. Genre and music clips used for evaluation

Music title	Genre	Music clips
Harder, Better, Faster, Stronger	Electronics	1:36–2:36
Firework	Hip-pop	0:00–0:50
Nope!!!!!	ACG	0:00–1:00
Sunflower	R&B	0:00–0:27
H.O.L.Y.	Country	0:00–0:26

3.2   Results and Analysis

Figure 6 displays the gained HAPTI results. Since I (Interaction) is a binary option, it is not drawn in the chart. We take the average value of each grade (i.e. B denotes 82) and
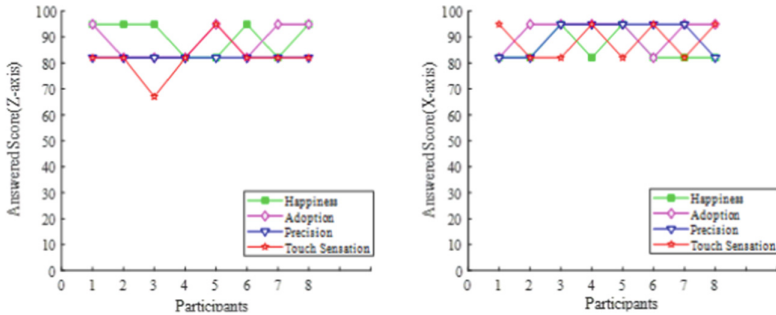

Fig. 6. The numbers of participants constitute the horizontal axis, while the gradings for four metrics (HAPT) constitutes the vertical axis. The two charts respectively show the evaluation results collected by the two test devices respectively (left: Huawei Mate 20 Pro's, right: Sony Xperia XZ2's).

draw the two graphs. On the whole, almost all the scores fall to or above Grade B (75–89), and Table 4 shows the average evaluation results of HAPTI collected respectively from the two testing smartphones.

The concept of bringing haptics to multimedia interaction is solid, as 100% of our volunteers recognize the prospect of MuViT as a product (see Interaction in Table 4). As for the other four markers, Xperia XZ2 wins a relatively better set of results, which doesn't surprise us, since the X-axis linear motor carried by Xperia XZ2 is superior to the Z-axis linear motor in Mate 20 Pro on the transient responsivity as well as the dynamic range. Additionally, Xperia XZ2 achieves more than 90 points in P (Precision), which embodies the outstanding adaptability of our algorithms. As it turns out, the idea behind MuViT is well-accepted, and it has shown fairly good performance as a tangible product prototype.

Table 4. HAPTI evaluation results in mean values

Model	H (Happiness)	A (Adoption)	P (Precision)	T (Touch Sensation)	I (Interaction)
Sony Xperia XZ2	85.3	91.8	90.1	88.5	100
Huawei Mate 20 Pro	90.1	88.5	82.0	81.8	100

Finally, the Compatibility (C) of HAPTIC is evaluated on ten mainstream multimedia platforms, including YouTube, YouTube Music, Apple Music, Spotify, TikTok, Netflix, Tidal Music, SoundCloud, QQ Music, and Bilibili. MuViT (Viv Beats) has achieved 90% compatibility ratio on these ten software platforms, among which only Tidal Music is incompatible with our system. This is to say that, unlike most of the other solutions, MuViT is a musical haptics extensible component on your smartphone in the true sense.

4   Conclusion

In this paper, we propose a musical haptics system called MuViT which is built for modern smartphones. The system takes an Android application (Viv Beats) as a carrier, and can provide vibrotactile feedback following the audio playing in the background in real time without any prior knowledge of the soundtrack data and regardless of the audio sources. We also design a HAPTIC evaluation method and conduct a real-world survey to let real participants help us assess the concept as well as the performance of MuViT, and the evaluation results indicate the usefulness and the prospectiveness of the proposed system.

Multimodal, or multisensory multimedia interaction, is one of the emerging areas, and we believe such kind of techniques will guide a prevailing and even essential way of multimedia interaction in the future, as long as both the technology providers and content providers work together to make this concept more and more mature.

References

1. Mazzoni, A., Bryan-Kinns, N.: Mood glove: a haptic wearable prototype system to enhance mood music in film. Entertainment Comput. **17**, 9–17 (2016)
2. Lin, X., Mahmud, S.: Virtual reality-based musical therapy for mental health management. In: Chakrabarti, S., Paul, R. (eds.) Computing and Communication Workshop and Conference (CCWC) 2020, pp. 948–952. IEEE, Las Vegas (2020)
3. Elvezio, C., Amelot, P.: Hybrid UIs for music exploration in AR and VR. In: IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct) 2020, pp. 411–412. IEEE, Munich (2018)
4. Kobayashi, N., Matsumoto, M.: Music learning support system using blocks. In: Matsuo, T., Takamatsu, K. (eds.) 12th International Congress on Advanced Applied Informatics (IIAI-AAI) 2022, pp. 164–169. IEEE, Kanazawa (2020)
5. Nanayakkara, S., Taylor, E.: An enhanced musical experience for the deaf: design and evaluation of a music display and a haptic chair. In: Proceedings of the Conference on Human Factors in Computing Systems, pp. 337–346. ACM (2009)
6. Hayes, L.: Skin music (2012): an audio-haptic composition for ears and body. In: Proceedings of the Conference on Creativity and Cognition, pp. 359–360. ACM, Glasgow (2015)
7. Hwang, I., Lee, H., Choi, S.: Real-time dual-band haptic music player for mobile devices. IEEE Trans. Haptics **6**(3), 340–351 (2013)
8. Project's GitHub Repository. https://github.com/henryyantq/MuViT. Accessed 6 Oct 2022
9. Merchel, S., Altinsoy, M.E.: Auditory-tactile experience of music. In: Papetti, S., Saitis, C. (eds.) Musical Haptics. SSTHS, pp. 123–148. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-58316-7_7