Interactive Mobile Music

A Practical Guide

yvan vander sanden
Mute, Leuven, Belgium

Abstract

It's not always clear what is and what is not possible when working on interactive music for mobile media. In this text we will give a shortlist of possible interactive combinations. The most current version of this list is available on our website (http://mutecode.com)

contact: yvan@mutecode.com

Contents

Introduction

When creating music for mobile media it is important to realize a crucial difference between Android and iOS. While Apple only produces expensive, high-end devices with about every type of sensor available, everyone can make an Android phone. This means that the hardware on an Android phone varies a lot more. One can buy an Android phone of €50 today, but while a €500 Android phone is as good as a €800 iPhone, a €50 Android phone is a lot more limited. It is because of this wide range of possibilities that we choose to support only mid-range and high-range Android devices.

The text below lists interactions which are possible on iOS and most Android phones priced €250 and up. We also limit ourselves to devices at most 3 years old. Older devices might work (especially the more expensive ones) but there is a fair chance they won't.

1. Playing Audio

Android and iOS both use *OpenSL ES* for generating interactive sound[1]. This has the following implications:

- **32 sounds can be played simultaneously.** This is plenty for most applications. Should you want to generate music note-by-note, 32 sounds is not that much. In this scenario you'd want to limit yourself to 8 voices to allow for overlap between notes.

[1] We have not included the features of our upcoming YSE sound engine just yet.

- **Volume.** The volume of every sound can be changed while playing. Gradual changes are possible.

- **Pitch & speed.** The pitch & speed of every sound can be changed while playing. Just like in most software or analog devices, pitch and speed are always changed together. Higher is also faster, lower means slower. Playing a sound faster is limited to two times the original speed *(one octave up)*. Slower playing has a limit of 0.5 times the original speed *(one octave down)*.

- **Panning.** Sounds can be panned in a virtual space. This correlates to left-right panning and volume changes with headphones. When using the devices' speaker, only a volume change will occur.

- **Encoding.** Sound are encoded as ogg *(comparable to mp3)* and take about 1MB of disk space for every minute of stereo sound. Short audio samples which should be played at different speeds can be included as wave files for better results, but this takes a lot more disk space (about 40MB a minute). A typical application will be about 30 MB in size without any audio. Preferably, the total size of the application should be below 100 MB.

2. Interactivity

The following sensors are available on most devices and supported on iOS and Android.

2.1 GPS

GPS is available on all phones. Resolution can be as good as $1m^2$, but it is better to assume $5m^2$ to allow for jitter. A lot of factors can have an impact on GPS reception: clouds, rain, trees, high buildings *(mirror glass buildings in particular)*, large water masses, ... And not all devices are equally good with handling GPS either.

In our experience, trying to compensate for these factors is no use at all and only results in less optimal performance. It is better to assume conditions are optimal and explain to the audience results can vary.

GPS returns the following information:

- Latitude

- Longitude

- Altitude *(meters above sea level)*

- Speed *(the approximate moving speed of the user)*

This means you are not limited to assigning sound to a fixed location. Other indicators can be used as well.

Most tablets do not have GPS, but in urban areas they can use WIFI for location tracking. WIFI is less accurate though.

2.2 Compass

Most devices can track which way is north. Of course a device cannot know how the user is holding it. You should assume that the user is holding his device in front of him, horizontally.

2.3 Gyroscope and accelerometer

These sensors can be used to track the motion of a device. We have not done any tests with these, but some crude motion detection should be possible.

2.4 Multi touch

Every device can track at least 3 simultaneous touches on the screen. Most devices support at least 5. Movement of these touches can also be tracked independently.

Of course sensors are not the only means to get information from a device. The time of day and the current date can also be used. If an internet connection is available it is also possible to retrieve other information, such as the local weather.

3. Conclusion

In short, every combination of two or more parameters is possible. This combination does not have to be exact: a relative approach is also possible. Below is a short list of possible combinations.

With 'the music can change' we mean a sound can start, stop, fade in or out, change in volume, pitch & speed or left-right panning. And while it is not possible to apply filters on sound, you can create multiple versions of an audio fragment, play them at the same time and do a cross-fade between them.

- The most obvious: music changes when you move to another location.
- A bit more surprising: music changes when you move to another location, but the change is dependent on where the user has been before. *(So we track where the user has been and decide what to do depending on where he comes from.)*
- It is possible to mimic sudden device movement in the audio by modifying the sound or playing different ones.
- The location of a sound can also change over time, or by user interaction. *(A user touching the screen could mean dragging the sound along until the screen is released.)*
- A different time of day or the local weather could result in other sounds playing.
- In a mix of several looping sounds, a certain sound can grow louder if a user walks uphill *(using the GPS altitude value)*.

Bear in mind that a change in the sound texture is only meaningful to the user if it is clear what causes it. The relation between the interaction and the audible result can be emphasized in the audio itself, or by onscreen visualization.