Audio editing software for android mobile

In recent years, mobile has become the go-to platform for most people's media consumption. From audio playback to movie streaming, there's a growing amount of content available in your pocket and tablet, and the market is still expanding. Today we see a step towards high-end 3D gaming environments, live music aids, and even home studio audio software suites designed to work on mobile devices and tablets. But Android unfortunately has not been at the forefront of this growing market, that position is firmly held by Apple.Especially in a creative capacity, tablets quickly replace laptops for music creation and live performance uses. Not to mention that there is an entire market for digital effects, which can be purchased at much lower costs than traditional analogue equipment. Line 6's latest digital effects amplifier is designed entirely around a mobile interface, but Android support is nowhere in sight. However, migration to more digital content requires higher levels of processing power, on a platform limited by smaller batteries and thermal constraints. Android owners pride themselves on having some of the best hardware on the market, so why is it that Android seems so far behind its rival when it comes to audio applications? A little about audio processing Our mobile phones are more than powerful enough for simple playback tasks. But as processing power has increased, we've also started demanding more signal processing from our mobile devices, much more of it in real time as well. We can take it for granted, but even when playing a game, each audio file takes time to be pulled from memory, converted from binary information to numerical values, before being pushed to a DAC, taking up valuable clock cycles. Additional post-processing, such as sending the file through your optimized EQ settings or supplementing the audio with extra reverb, takes up even more time, and modern applications are becoming more complex. Although modern mobile processors have long surpassed the multiple GHz brand and can match high-end PC equipment in core counting, these simple numbers aren't all that matter when it comes to digital signal processing. Different processors design completes different tasks in a different number of clock cycles, making some CPUs faster than others on the same tasks. That's why direct GHz and core count comparisons don't always apply over design. Android may not be able to compete with expensive studio grade hardware, but much can be accomplished on a tight processing budget, if you know where to optimize. With real-time audio, it is essential to be able to process floating point data, digital numbers with decimal points, and SIMD instructions (simple instruction, multiple data) instructions quickly, preferably all within the short time between samples, which is usually 44100 or 48000 kHz for most audio applications. Floating point units, a mathematical coprocessor usually located in the CPU core mathematical operations on the digital audio signal with high levels of accuracy are used. Multiple cores are not that important for sound – instead brute speed is key. Multiple cores are not that important for audio, because most DSP algorithms are not optimized for multiple threads, instead brute speed is key. The limitations of mobile processors, in this regard, can be found in the smaller memory bus bandwidth and less CPU cache's, compared to beefier desktop grade PROCESSORS. This may mean that your mobile CPU may actually end up spending more time waiting for data as it does processing it. An example of one of the more – if not the most – demanding sound processing tasks is time stretching, where the tempo/speed of an audio sample changes without trade-offs in pitch/frequency change that can come from changing a sample wavelength. In this technique, sound converted to digital, a Fast Fourier Transform algorithm then extracts the frequency information from the sound, which is used to correct/recover frequency information as the sample is stretched or shrunk in the time domain. Fixed Fourier Transform is the process of extracting specific frequency information from a more complex waveform, and is very CPU intensive. Sounds pretty complicated doesn't it? This type of process puts a huge load on the CPU, which can result in unacceptable latency. There are actually fewer than five FFT algorithms in the world that can run this type of process on mobile devices effectively. The maximum latency in all real-time systems should ideally not exceed 20ms, which is about the perceptual limit of delay in humans. Longer and our brains will notice the difference between sounds coming in and out of a system, or between a button press and something happening on the screen. Unfortunately, typical Android latency is located in the region of 100 to 250 milliseconds. In an effort to boost performance and work around some of these flaws, mobile SoC developers, like Qualcomm, have begun to include their own dedicated DSP hardware along with their key processors. ARM & DSPARM has long included floating point devices in almost all of its core designs, excluding Cortex-M3 and below, and supports additional digital signal processing and SIMD extensions in its mobile processors. ARM's SIMD extension and NEON engine are particularly important for these types of scenariosThis DSP processing capacity aims to keep power consumption down, while maintaining the maximum performance available, up to 75 percent higher than the one that can be achieved without the extensions. ARM's tools are used for a range of common mobile applications, from monitoring sensors, to voice recognition, VOIP and audio code/decode. ARM's SIMD extension and NEON engine, which is available in the standard ARMv7 architecture, are especially important for the types of scenarios that we're talking about. ARM has done special optimization for faster sleep, 4-8x DSP algorithm improvements, dedicated tools for Fast Fourier Transform applications and a host of other optimization to perform complex and processor heavy mathematical calculations on a strict power budget. ARM's NEON Data Engine and Floating Point Units, which are available in all Cortex-A patterns, are important for efficient DSP processing. The transition to ARM's 64-bit ARMv8 architecture may also have some useful benefits for audio software developers and consumers, as audio applications can be highly memory dependent, and 64-bit devices with larger frame pools. However, there is only so much that ARM can do on its own, and ARM's library just really serves as an example of how developers can go about creating their own lower code. Without a full-fledged library, different developers would have to go over the same processes over and over again just to build the basic tools they need. A further obstacle for smaller development teams is the high cost of ARM's proprietary compiler. AudioDevelopment and AndroidWhile we have mobile hardware that is clearly capable of providing a high quality audio app experience, there seems to be a lack of application support for developers on Google's side of things. For the app developer, the first port of call is usually the Android SDK. But Google's Media API's for Android are pretty limited, to say the least. You will not find many useful tools, blocking very basic MediaRecorder and playback from file functions. Delving a little deeper into the various Android packages will reveal some tools for an equalizer, reverb pre-set and noise suppression. However, there are no acceptable tools for low latency real-time audio processing, and the various operating system fragments found out in nature often mean that these tools can suffer and miss depending on the user's hardware. Android has an acceptable selection of audio-focused apps already, but the platform doesn't play so nice with the wider world of audio. Comparing this situation with Apple's iOS platform, it couldn't be in greater contrast. Apple has long included its Core Audio digital audio infrastructure in its operating systems, offering developers a dedicated software framework for a variety of applications, such as the ones we've already discussed. There seems to be a lack of application support for developers on Google's side of thingsThe Core Audio library includes tools for mixing and converting signals and files, easily implementing signal chains, as well as essential built-in effects, while maintaining high performance. Apple also includes easy access to its hardware abstraction layer, allowing audio applications to easily interface and communicate with other parts of hardware, such as microphones or output devices that accept incoming audio signals. Most of this functionality is completely missing from the Android platform. As painful as it is to admit, Apple's Core Audio platform is much more developer friendly than Android ecosystem. Instead, more complicated programs may find that they need to do much more of the low-level coding themselves, adding development times and costs. This is the main reason why Android is so far behind Apple, when it comes to advanced audio applications. That is, if you can not find a third party SDK. Introducing - SuperpoweredSuperpowered is one of the few feature rich audio SDKs available for mobile, which has just recently been made available for Android. It offers up a range of tools for Android and iOS developers to easily implement some more complex audio applications and effects. The SDK offers up a library of prebuilt features for sound filters, reverbs, echo effect, time domain stretching, and FFT, all designed with high quality studio grade audio in mind. Superpowered has been built from the ground up to maximize DSP performance, while siding problems with Android audio issuesVarious other audio engines, Superpowered is not a cover around Core Audio or Android's pre-build library. Instead, it has been built from the ground up to maximize DSP performance, while bypassing problems with Android fragmentation, its lacklustre feature set, and latency issues. Superpowered claims that, as a result, it can even surpass Apple's industry's famous Core Audio platform, which is no mean feat. Superpowered is designed for ARM devices that use the NEON architecture extension, which basically means that 99% of smartphones and tablets are covered. It can be used to accelerate the development of almost everything audio related, from DJ apps and instrument effects, to audiobook readers, podcast apps and games. The video below shows off Superpowered's co-founder demoing a portable powered DJ interface powered by the platform. Important for developers, Superpowered is a cross-platform SDK, allowing apps to seamlessly work on both Android and iOS without differences in audio quality. While iOS may be the leading platform at the moment, this opens the door for a wider number of developers to consider Android as well. Superpowered is not stop with sound though, the company will also be releasing dsp SDKs for image and video processing in the near future. Which could open up Android to a new generation of media editing apps and content. If you are a developer interesting Superpowered's SDK, the good news is that it is free to download and implement in your app. When the app reaches 50,000 installs, Superpowered helps you set up a contract with them, which includes extra support for your app. With hindsight, Android's lack of out-of-the-box support for high-end audio apps and features seems to have been a missed opportunity. Fortunately, third-party developers have stepped up to provide solutions to the problem. In the future, hopefully Android will prove to be a worthy platform for power media developers as well. For.