I recently ran into a problem with how one of my video editing programs imports a specialized audio track, specifically engineered to deal with YouTube's compression algorithm when you upload videos to the platform. It should be known that YouTube compresses both video and audio from its original sampling rate once a video is processed through the upload system. The program I was working with was Pinnacle Studio 24 from Corel. Some projects go smoothly. Others send you on a wild goose chase, digging through software settings, running test after test, and questioning your life choices. This is the story of the latter—a deep dive into the frustration of handling audio files and the long road to finding a way around limitations of how some software's handle importing tracks. If you've ever imported a crisp, professional-grade audio track into a video editor only to have the final export sound dull and lifeless, you're not alone.

This issue wasn't immediately obvious. In fact, it took an outside ear to even catch it. An audio engineer—someone with a much sharper sense of sound than I have—noticed something was wrong. The higher frequencies, the "top end" of the audio, were being cut off in the final export. The mix sounded fine on my editing timeline, but something was happening during the rendering process. What followed was a multi-layered, 3-day, troubleshooting process, leading to some surprising *(and infuriating)* discoveries about how Pinnacle Studio handles audio tracks.

Step 1: Confirming the Problem:

The first clue that something was off came before I even uploaded the video to YouTube. The song, originally a rich and balanced .wav file, suddenly sounded like it had been run through a low-pass filter. The shimmer of cymbals, the brightness of the vocals, and the overall clarity were gone.

At first, I assumed it was YouTube's infamous compression. It's well known that YouTube processes uploaded videos with its own audio compression algorithms, sometimes reducing quality. But even playing back the exported file before uploading revealed the same issue—YouTube wasn't the only problem. Pinnacle Studio itself was altering the audio. Both on the import into the program and the rendered export of the finished product.

Step 2: Checking the Import Settings:

One of the most overlooked steps in video editing is how your software imports media. Most people assume that dragging and dropping a file into the timeline means it stays in its original format. Unfortunately, that's not always the case.

Pinnacle Studio doesn't have the most transparent way of handling audio files. While it can work with .wav files, the way it processes them internally is another story. There was no obvious warning that my high-quality, uncompressed audio was being altered upon import, but I had my suspicions.
To test this, I tried playing the original .wav file outside of Pinnacle in a separate audio player. It sounded perfect. Then, I played the same file within Pinnacle's timeline. It still sounded fine. But after exporting? That's where the trouble began. The import settings are defaulted and cannot be changed directly for just an audio tract. One usually sets up the project parameters and Pinnacle will treat the files based on their presets on how one sets up the parameters. While no direct compression was detected. I went ahead and went into all the audio effects and made sure if anything was added automatically, that they were removed and the audio track was locked down after that. I do not know for sure if this does anything or not but I was able to delete some settings added to the track when I clicked on corrections. However, I believe nothing was added that made a direct difference. Only a placeholder for a change was in the settings and I just clicked on the garbage can to remove it in the case any changes were made. When I played the track before and then after I did not see or notice any changes. This made me think it had to be the output settings.

Step 3: Tweaking Pinnacle's Export Settings:

If importing wasn't the issue, maybe it was the export process. Pinnacle Studio 24 has multiple export presets, and many of them apply compression by default. By default, it exports in AAC audio format, which, while decent, applies lossy compression. Lossy compression is fine for casual videos, but not when you need to preserve the full depth of a professionally mixed music track.

I started tweaking settings.

- First, I tried exporting in .mp4 with the highest available audio bitrate. No change.
- Then, I tried different sample rates and bit depths. Still no change.
- Finally, I tried exporting in .mkv, just to see if that made a difference. It didn't. it actually crashed Pinnacle every time I tried to run the export as .mkv.

At this point, I was convinced that Pinnacle was doing something to the audio that wasn't obvious in its settings. Pinnacle in general is a massive black hole on system resources on a windows computer. Memory (RAM), Graphics Card (GPU) are working so hard my whole system is almost crippled by the usage. I have an old computer and it's very possible my installation of Pinnacle on this machine is too much a strain for the system to work as it's supposed to work. I have to shut down most of my computer programs, and web browsers in order to run Pinnacle smoothly.

Step 4: The Hidden Culprit – Pinnacle's Audio Presets:

Digging deeper into Pinnacle's settings, I discovered something frustrating: Pinnacle Studio 24 applies audio compression and EQ settings by default. As I discussed before this was nice to find and know it was adding something to my audio track, but not enough to make that the main reason the problem existed in the first place. These aren't listed as part of the export settings, but rather buried in the "Corrections" tab under audio adjustments.

There, I found an automatic preset subtly altering the sound. Deleting these presets improved the clarity, but the issue still wasn't entirely gone. Pinnacle was still rendering the audio in a way that affected the high frequencies.

Step 5: A Workaround – Enter VSDC:

If Pinnacle wasn't going to let me export a clean audio track, I needed another solution. That's where VSDC came in. VSDC is a free, non-linear video editor that offers more control over export settings, especially for audio. Instead of fighting Pinnacle, I decided to take a different approach. I had been using VSDC a lot more than Pinnacle in the past few months. VSDC can do a lot of things Pinnacle cannot do. However, Pinnalce's interface is a little easier to work with for this type of work. I have yet to really learn VSDC to a point where this is what I always use. I have been a Pinnacle Systems-guy since the year 2000. I have used many other PC-based editors but I always stuck with Pinnacle. Its only been recently that I have been using VSDC. I began using it for its audio spectrum abilities but then I learned VSDC have many options for exporting videos using different presets and being able to customize those presets in a much more refined and detailed way:

1. **Export the video from Pinnacle Studio 24** with the best possible visual settings but accept the slightly degraded audio.

2. **Import the exported video into VSDC** and replace the audio track with the original .wav file.

3. **Export the final version in VSDC** using lossless PCM S16LE audio encoding.

This method allowed me to bypass Pinnacle's internal audio processing while still using it for its video editing capabilities. The results? A massive improvement. The final export retained the original high frequencies and overall clarity of the song.

Step 6: Uploading to YouTube – The Final Test:

Even after fixing the problem on my end, there was still the matter of YouTube's own compression. Some degradation is unavoidable when uploading to YouTube, but I wanted to minimize it as much as possible.

Through further research and testing, I found that YouTube applies less aggressive compression to videos exported in **.mkv format with PCM audio**. While .mp4 is the most common format for YouTube uploads, it often triggers heavier compression, especially on the audio side. By exporting in **.mkv with PCM S16LE** audio compression, I managed to retain far more of the original sound quality after uploading. The difference wasn't night and day, but it was enough to be noticeable, especially to those with trained ears.

There were several triggers I needed to understand with how VSDC allows the user to edit its export settings. First, I had to choose this is an export for a computer file. Not necessarily, for YouTube or social media but an export for my computer. Second, I had to choose .mkv as the export settings. Third, I then could tweak the audio compression export settings. Doing it this way tells VSDC to unlock more encoders to choose from. When choosing say YouTube as an .mp4 it only allows AAC to choose from. What we wanted here was PCM. I cannot access PCM from the dropdown unless I choose the output as for my computer, as a .mkv, then I am able to choose PCM from choices of audio compression encoders. I could probably find a way to edit this so that I can use them all in any format-presets, but I am not nearly as crafty about this as I'd like to be. Plus, it's of no real consequence here. The point is I needed to be able to export the video in full HD with this audio encoding feature in order to trick YouTube to not compress as much on this file over other uploads.

Final Thoughts – Lessons Learned:

This entire experience was a reminder of how complex audio handling can be in video production along with all the issues that can and most of the time will arise within a given project and the resource demands on one's system. Even with high export settings, Pinnacle Studio applied hidden audio processing that degraded quality in ways I didn't expect. Along with its limited control over specific output settings. It took three days of research, watching videos on the subject, having dialog with AI, trial and error, overnight renders, and format tests to pinpoint the problem—and the solution.

VSDC proved to be a critical tool for final audio processing, allowing me to replace the degraded track while keeping the video quality intact. YouTube also handled audio better with **.mkv using PCM S16LE**, which made a noticeable difference in the final upload. While I originally used VSDC as a workaround in this case, it's clear this program is more modernized on how video and audio are handled inside the program, how the tools work with your system and efficient in handling certain tasks that Pinnacle struggles with.

This reinforced a simple truth: **—no software is perfect.** Pinnacle has served me well for decades, but its hidden processing, resource-heavy nature, and age of the software made this project more complicated than it needed to be. In contrast, VSDC runs smoother, is far less resource intensive, and allows me to keep multiple programs open without taxing my system. While my working knowledge of it isn't strong enough to fully replace Pinnacle yet, I'll been integrating it more into my workflow with smaller projects and will be using VSDC more moving forward. At the end of the day, **video editing is as much about problem-solving as it is about creativity**. Never assume your software is handling things exactly the way you want—test, experiment, and be ready to pivot. For now, the music video is finished, the audio is intact, and the lesson is learned.

Sine Damno Educendi
Latin for: (Exporting Without Loss)
By David-Angelo Mineo
3/16/2025
1,965 Words