

VGA: Vector Graphics Adapter

Vektorgrafik auf dem Oszilloskop dürfte so alt sein wie der X/Y-Modus. Neben [Lissajous-Figuren](#) erfreut sich die Soundkarte als Quelle für Oszilloskop-Vektorgrafik einiger [beliebtheit](#). Dabei werden linker und rechter Kanal mit der X bzw. Y Auslenkung des Oszilloskops verbunden und bewegen so den Elektronenstrahl im analogen Oszilloskop.

Eine Soundkarte ist im Prinzip ein Digital-Analog Wandler, hat jedoch Tiefpassfilter welche zwar die Audio-Qualität erhöhen, aber die Geschwindigkeit mit welcher der Elektronenstrahl bewegt werden kann eng limitiert. Außerdem sind in der Regel Koppelkondensatoren verbaut, welche den für Lautsprecher schädlichen Gleichstromanteil entfernen. Dadurch verliert die Grafik jedoch den Nullpunkt und wabert um den Mittelpunkt falls sie nicht gut auf diese Gegebenheiten angestimmt wird.

Ziel des Projektes ist eine stabile Ausgabe beliebiger (Vektor-)Grafik zu erreichen.

0. Vektorisierung mit patrace

Zu allererst muss Material zur Anzeige her. Die Wahl viel auf das Video [Bad Apple](#), welches sich als reines schwarz-weiss-video Video leicht vektorisieren lässt.

Dazu wird zuerst das Video zunächst in seine Frames zerlegt:

```
ffmpeg -i <Quellvideo> -vf scale=480:-1 -f image2 <Videoframes>-%4d.bmp
```

Und dann vektorisiert:

```
for F in <Videoframes>-*.*.bmp; do
  potrace -s --flat -t25 -a0 "$F.bmp" -o "$F.svg";
done
```

Die Optionen erzeugen ein SVG mit nur einem Pfad welches sich weiterverarbeiten lässt. Ein Ruby Skript erzeugt schließlich daraus PCM Output:

[Vector Convert PCM](#)

1. Verbesserte Kantenerkennung mit imagemagick

Für nicht schwarz-weiße Videos ist die Schwellwörtererkennung welche patrace durchführt, nicht zufriedenstellend. Daher kommt als Zwischenschritt wird daher der [Canny](#)-Kantenerkennungsalgorithmus eingeführt. Welcher auf den Video-Frames operiert und ein Schwarzweißbild erzeugt auf dem nur die Kanten des Originalbildes enthalten sind.

```
for F in <Videoframes>-*.*.bmp; do
  convert "$F" -canny 0x1+15%+20% -negate "$F.sw.bmp"
  potrace -s --flat -t25 -a0 "$F.sw.bmp" -o "$F.svg";
done
```

2. Atmel Xplain als externen DAC

3. VGA Port als DAC

4. Echtzeit Vektorisierung mit opencv

5. Videowiedergabe mit mpv

From:

<https://wiki.warpzone.ms/> - warpzone

Permanent link:

https://wiki.warpzone.ms/projekte:vector_graphics_adapter?rev=1447620465

Last update: **01.03.2017**

