

Fraktální audio vizualizér

Autorka: Radka Hošková

Vedoucí práce: Ing. Radek Richtr, Ph.D.

Fakulta informačních technologií ČVUT

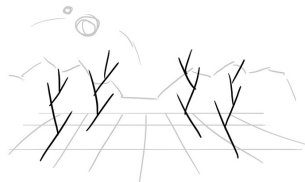
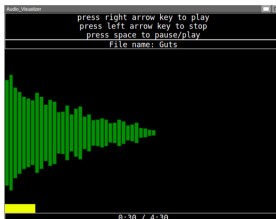


1 Úvod do tématu

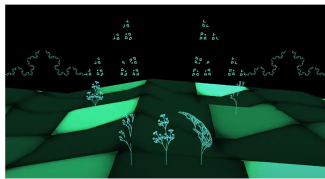
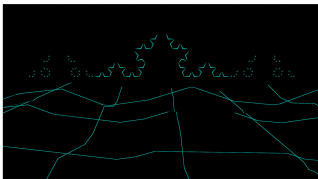
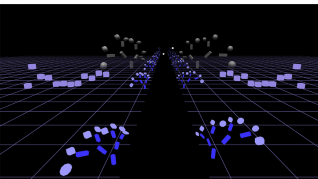
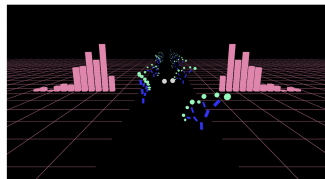
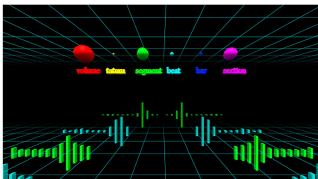
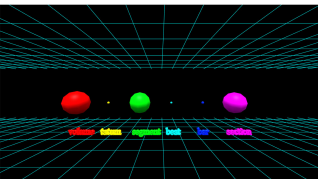
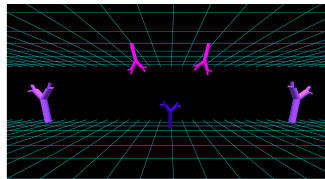
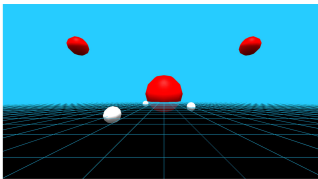
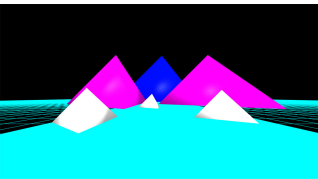
2 Zvolené řešení

3 Ukázka prototypu

Výběr tématu

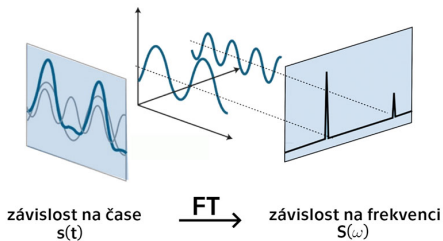


Předchůdci nápadu vizualizace hudby pomocí fraktálů



Jak získat data z hudby?

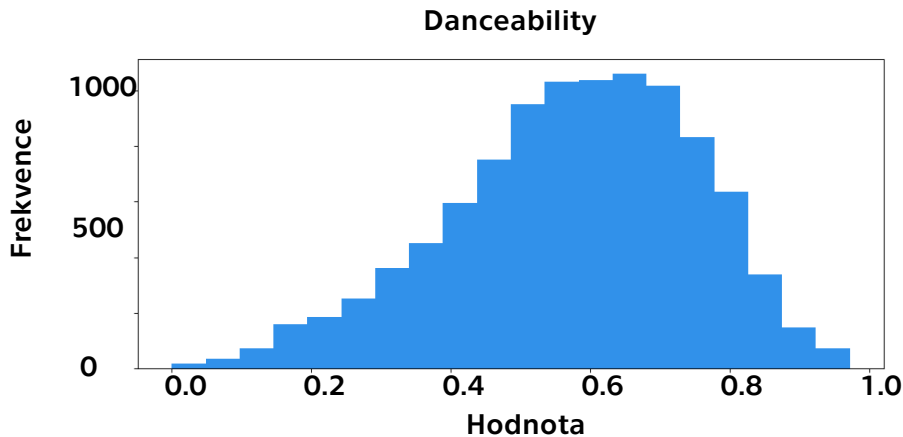
- MIDI noty
- rozložení kanálů
- existující API
- kombinace postupů



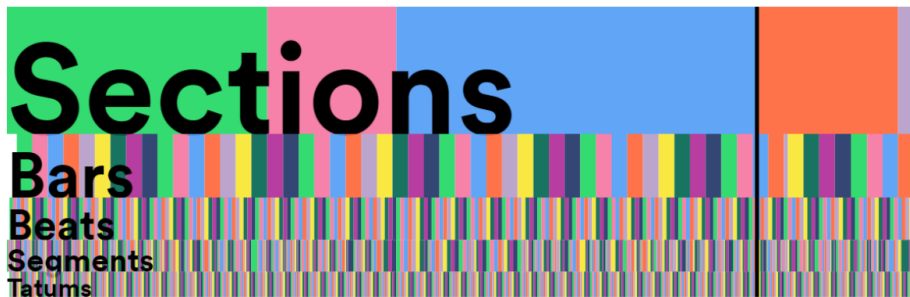
Fourierova transformace



- informace o skladbě
- hudební atributy
 - „acousticness“
 - „danceability“
 - „speechiness“
 - atd.
- analýza skladby
 - „confidence“
 - vektor „pitches“
 - vektor „timbre“
 - atd.



Distribuce hodnot atributu danceability



Spotify intervaly jedné skladby

Příklad proměnlivých parametrů



```
// pulsace beatu  
let beatPulse = beat.elapsed/beat.duration;
```

```
// stroboskop  
if (barPulse > 0.5) {  
    color *= podsviceniTextury;  
}
```

```
// rozplynutí stromu  
vec2 z = vec2(x * beatPulse * 1.5, y * beatPulse * 1.5);
```

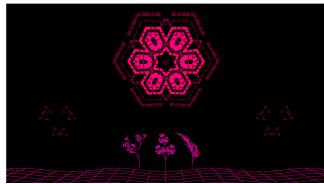
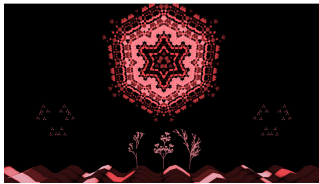
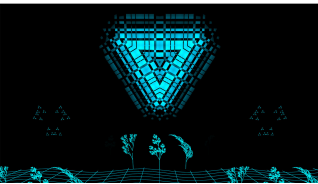
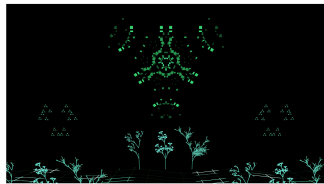
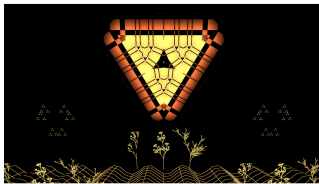
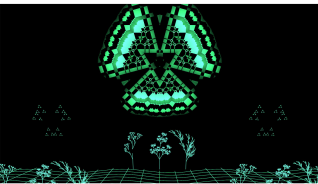
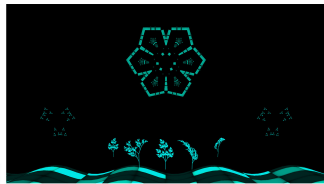
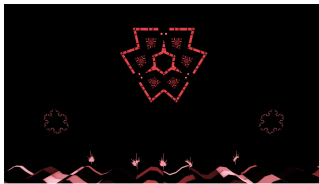
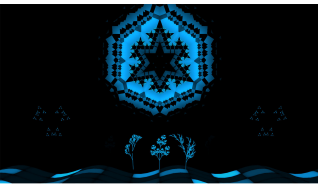


Příklad proměnlivých parametrů

```
// tanecni parket
if ( danceability > 0.65 ) {
    uniforms.option.value = 0;
}

// zive publikum
if ( liveness >= 0.8 ) {
    scene.add(audienceGroup);
}

// klidnejsi vlny
if ( energy < 0.4 ) {
    uniforms.wavelength.value = 3.26;
    uniforms.amplitude.value  = 2.7;
}
```



Otázky oponenta



- 1** Jak je řešena synchronizace hudby a vizualizace, aby to přesně odpovídalo tomu, co slyšíme?
- 2** Využívá některá vizualizace například proměnlivou hloubku iterace fraktálů v závislosti na hudbě (např. košatění stromů, detailnější Kochova křivka, atd.)?