

Generatiivinen musiikin tuottaminen

Lauri Gröhn
Gallery Music from Pictures
www.synesthesia.com

Tiivistelmä

Esiteltävä Synesthesia-ohjelma (Java-sovelle) voi olla hyödyllinen väline kun halutaan tuottaa eri parametrien suhteen toisiaan lähellä olevaa testidataa (midi-tiedostoja) kehitettäessä automatisoitua musiikin analyysia ja hakuja. Esiteltävä menetelmä midi-tiedostojen tuottamiseksi mistä tahansa kuvasta saattaa antaa muitakin virikkeitä ja uusia ideoita. Kehitetyn musiikin generoinnin menetelmän metaforana voidaan pitää kuvanveistoa, jossa lähdetään kivenkappaleesta tai puunpötkelöstä, joka inspiroi kuvanveistäjää tiettyyn ratkaisuun. Esiteltävä menetelmä perustuu kuvan pikseleiden poissuodattamiseen ja toisaalta useiden musiikillisten valintaparametrien käyttöön. Kunkin sävellyksen pohjana oleva kuva ja parametriluettelo ovat yhdessä sävellyksen metapartituuri.

1 Johdanto

Tässä esityksessä käsitellään vain Synesthesia-ohjelmiston avulla tapahtuvaa musiikin automaattista generointia. Lukuisiin muihin vaihtoehtoisin voi tutustua Internetin välityksellä tai esityksen lopussa mainittujen kirjojen perusteella.

Synesthesia on säveltämisen kehitysjärjestelmä, jossa ei ole omaa käyttöliittymää ja jota ajatetaan JBuilder Java-kehitysympäristön tai muun vastaavan kautta. Tyypillinen konfiguraatio on sisältää JBuilderin käyttöliittymän, kuvakansion ja miditiedostokansion (liite 1). Kun parametrit on valittu, generoi ohjelma käynnistyksen jälkeen miditiedoston muutamassa sekunnissa. (liite 2).

Syntyvä midi-tiedosto voidaan soittaa millä tahansa mediasoittimilla. Musiikkiesimerkkejä midi- ja mp3-muodossa on runsaasti kuvineen nettisivulla www.synesthesia.com.

Toistaiseksi en ole suorittanut miditiedostojen jälkikäsitelyä. Raakapartituurista näkee, että jos musiikki halutaan saada esitettävään muotoon, on miditiedostosta saatavaa partituuria (liite 3) muokattava käyttökelpoisempaan muotoon (liite 4).

Generoinnissa syntyy kokonaisia sävellyksiä, jotka ovat parametreistä riippuen minimalistisia, new age –tyylisiä tai vahvasti klusteroituja, genrenä lähinnä ”klassinen nykymusiikki”. Kestot ovat olleet minuutista 10 tuntiin, Ohjelmassa ei ole lainkaan käytetty satunnaisuutta.

Tässä esityksessä käydään läpi joukko Synesthesia-ohjelman ideoita erittäin karkealla tasolla.

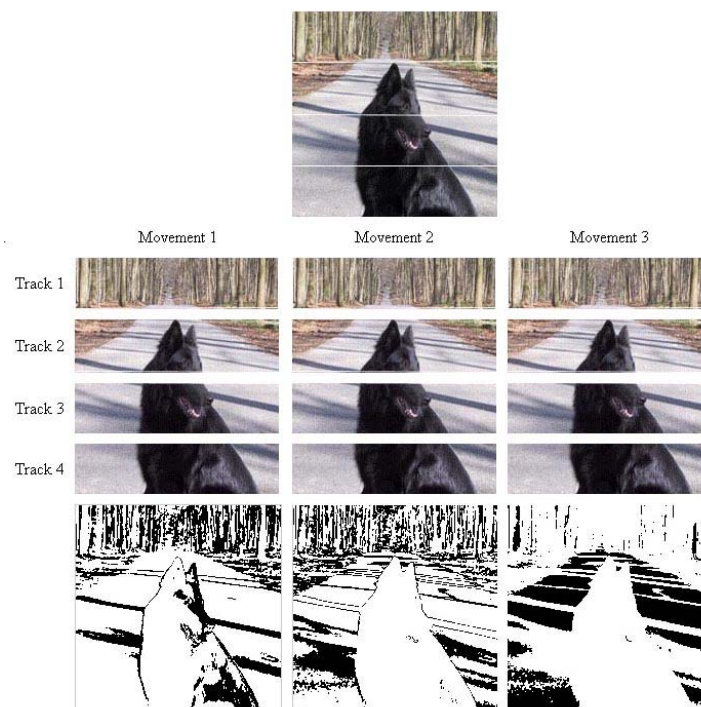
2 Kuvan suodatus

Ohjelmassa valitun kuvan kunkin pikselin r/g/b -arvot lasketaan yhteen (moodi 0) tai kerrotaan keskenään (moodi 1) ja jokaisella suodatuskerralla (parametri) ne pikselit jotka ovat johtaneet yleisimmin esiintyvään arvoon ”veistetään pois”. Kuvassa 1 ylinnä alkuperäinen, seuraavana moodissa 0 suodatettu 10 ja 400 kertaa ja alinna moodissa 1 suodatettu 100 ja 400 kertaa.



3 Kuvan pilkkominen

Kanavien (parametri) määrästä riippuen kuva jaetaan pystysuunnassa yhtä suuriin lohkoihin



ja osien (parametri) määrästä riippuen kunkin osan pikseleinä käytetään käytössä olevien pikselien määrää jaettuna osien määrällä. Osassa 1 käytetään tumminpia pikseleitä ja viimeisessä vaaleimpia.

4 Säveltasojen määräytyminen

Pikselistöä käydään läpi ajallisesti vasemmalta oikealle niin monta kertaa kuin sävellyksessä on osia. Aika-akseli siis kulkee vasemmalta oikealle. Sävelkorkeus määräytyy koko sävellyksen korkeimmasta nuotista (parametri), kunkin trakin suhteellisesta säveltasosta ylempään verrattuna (parametrijoukko) ja valitusta asteikosta, joita voidaan muodostaa mielivaltaisesti lisää:

```

47 // Scales =====
48 int[] vscale [] = {
49 {0,1,2,3,4,5,6,7,8,9,10,11}, //#0 chromatic
50 {0,0,2,2,4,5,5,7,7,9, 9,11}, //#1 major
51 {0,0,0,2,2,5,5,5,7,7, 9, 9}, //#2 pentatonic
52 {0,0,2,3,3,5,6,6,8,9, 9,11}, //#3 octatonic
53 {0,0,2,2,4,4,6,6,8,8,10,10}, //#4 whole-tone
54 {0,0,0,3,3,5,5,5,6,6,10,10}, //#5 sakura
55 {0,0,0,0,0,0,0,0,0,0, 0, 0}, //#6 Scelsi
56 {0,1,1,3,4,4,6,7,7,9, 9,11}, //#7 Arabic
57 {0,1,2,2,5,5,6,7,7,8,11,11}, //#8 Messiaen mode 4
58 {0,0,2,2,3,5,5,6,6,9, 9,11}, //#9 harmonic minor
59 {0,0,0,0,0,0,1,1,1,1, 1, 2}, //#10 eyes wide shut
60 {0,0,0,0,0,0,0,0,0,0, 0,11}); //#11

```

Pikseleitä luetaan kussakin trakissa ylhäältä alas. Sävelkorkeudet, volyyymi ja ne saavat arvoikseen midi käytettäviä arvoja. On mahdollista, että jokaisella trakilla on oma sävelasteikkonsa.

5 Musiikilliset parametrit

Tempo, instrumentit jne. voidaan valita seuraavilla parametreilla tai parametrijoukoilla (taulukosta jokaiselle trakille erikseen. Harmoniat syntyvät trakin vertikaalinuottien määrästä, minimi-intervallista ja sävelalan laajuudesta.

```

7 // Main parameters =====//default =====
8 int nmov      = 3;          //3: structures (~ movements, piece length)
9 int ntracks   = 8;          //9: number of instruments (tracks) used
10 int mpitch    = 106;       //110: highest pitch of the first track (defines also key
11 int pdi       = 6;          //6: pitch difference to the upper track (harmony)
12 int [] pdiff  = {0,pdi,pdi,pdi,pdi,pdi,pdi,pdi,pdi,pdi,pdi,pdi,pdi,pdi,pdi};
13 int thm      = 50;         //36: pitch range of tracks
14 int [] trackhmin = {thm,thm,thm,thm,thm,thm,thm,thm,thm,thm,thm,thm,thm,thm,thm};
15 int mnt      = 1;          //1: maximum number of notes in chords in one track
16 int [] maxnotes = {mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt,mnt};
17 int itv      = 2;          //1: minimum interval in chords
18 int [] minitv  = {itv,itv,itv,itv,itv,itv,itv,itv,itv,itv,itv,itv,itv,itv,itv};
19 int ins      = 0;          //12: instrument (patch)
20 int [] instrum = {ins,ins,ins,ins,ins,ins,ins,ins,ins,ins,ins,ins,ins,ins,ins};
21 boolean instrumsel = true; //true:instruments set by SW

```

Kunkin nuotin volyyymi määräytyy seuraavalla tavalla:

```

30  int vom          = 77;          //77: maximum volume
31  int [] volmax    = {vom,vom,vom,vom,vom,vom,vom,vom,vom,vom,vom,vom,vom,vom,vom};
32  int vol          = 1;          //77: minimum volume
33  int [] voltracks = {vol,vol,vol,vol,vol,vol,vol,vol,vol,vol,vol,vol,vol,vol,vol};
34  int vlx         = 1;          //2: crescendo slope (cyclic)
35  int [] volx     = {vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx,vlx};

193      volumeset = voltracks[tr] + (pp[x][y]+volx[tr]*x)*(127-voltracks[tr]);
194      if (volumeset >volmax[tr])
195          volumeset = volmax[tr];
196      varLen(tr,dt[tr]*minnote);          //time
197      sabyte(tr,0x90+tr);                 //track
198      sabyte(tr,note(tr,y));              //pitch
199      sabyte(tr,volumeset);              //volume

```

6 Muutamia erikoisuuksia

Koska musiikkia voidaan generoida mistä tahansa kuvasta, ovat muutamit yksityiskohdat tuottaneet melkoisesti lisätyötä. Kuvasta riippuen on mahdollista, että musiikkiin tulee liian pitkiä taukoja. Ne on eliminoitu parametrillä, joka määrittelee sallitun maksimitauon. Liian pitkän tauon uhatessa ohjelma supistaa tauon parametrin ilmaisemaan pituuteen. Sävellyksen alun ja lopun pikselit on suodatettava ylimääräisellä tavalla, jottei sävellys ehkä alkaisi voimakkaalla purskeella tai lopu ikään kuin seinään. Lisää yksityiskohtia löytyy koodista, johon (vanha versio) pääsee käsiksi nettisivulta www.synestesia.com.

7 Lopuksi

Synestesia-ohjelmassa on Java-koodia nykyisellään runsaat 500 riviä ja koska käyttöliittymää ei ole, ei olio-ohjelmointia ole toteutettu taiteen sääntöjen mukaisesti. Kehitys-ympäristönä ohjelma on toiminut mainiosti, teknisiin rakenteisiin ei ole tarvinnut puuttua. Suoritin uudelleenohjelmoinnin kaksi vuotta sitten. Viitasaaren sävellyskursseilla tulleen muutosehdotukset olen pystynyt toteuttamaan muutaman rivin lisäyksillä tai muutoksilla.

Vaikeimmaksi haasteeksi on osoittautunut midin rytmisetin hyödyntäminen. Useiden yritystenkään jälkeen ei sopivia algoritmeja ole löytynyt. Sellaisten löytyminen voisi tuoda mahdollisuuksia siirtyä populaarimpaan suuntaan.

Aivan uutena mahdollisena haastena on kehittää ohjelma, joka tekisi staattisen kuvan midi-tiedostosta. Näin saatuja kuvia voisi käyttää uusien midi-tiedostojen generointiin, jolloin syntyisi eräänlainen korkeamman tason musiikkia tuottava soluautomaatti

Kehitetyn Java-sovelteen avoin lähdekoodi (kevään 2002 versio) löytyy nettisivulta www.synestesia.com kuten myös lukuisa joukko midi- ja mp3-tiedostoja kuvineen.

Kirjoja generatiivisesta musiikista

David Cope: ”*The Algoritymic Composer*”. A-R Editions 2000.

Esittelee generatiivisen säveltämisen historiaa ja käsitteistöä ja romppu sisältää useita ohjelmia algoritmiseen säveltämiseen.

David Cope: ”*Virtual Music, Computer Synthesis of Musical Style*”. The MIT Press 2001.

Käsittelee Copen kehittämiä ohjelmisto ja rompun näytteillä voi testata kysyään erottaa Copen tyyliemulaatiot alkuperäisistä klassisista pianosävellyksistä.

Eduardo Reck Miranda: ”*Composing Music with Computers*”. Focal Press 2001.

Käy läpi erilaiset generatiivisen säveltämisen paradigmat ja sisältää useiden ohjelmistojen esittelyjä ja rompulla itse ohjelmia.

Iannis Xenakis: ”*Formalized Musi, Thought and Mathematics of Music*”. Pendrakon Press 1992.

Katso sivuilla 329-325 liite The new UPIC system. UPIC tuottaa audiodata graafisesta datasta.

Kuvaliitteet

Liite 1: Kehitysympäristö käyttöliittymä parametrien valintavaiheessa

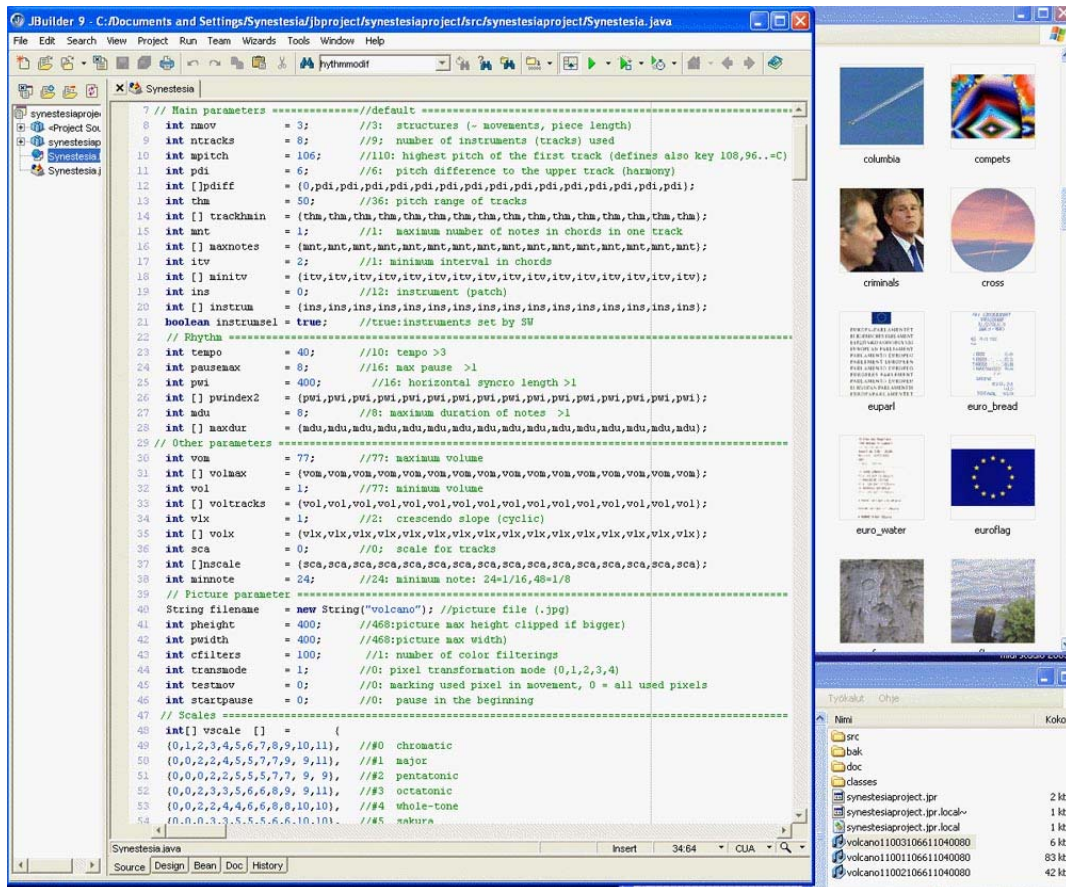
Liite 2: Näyttö generoinnin tapahduttua

Liite 3: Midi-tiedostosta saatu raakapartituuri, esimerkkinä ”Time is Up”

Liite 4: Raakapartituurista muokattu lähes muusikoiden soitettavissa oleva partituuri

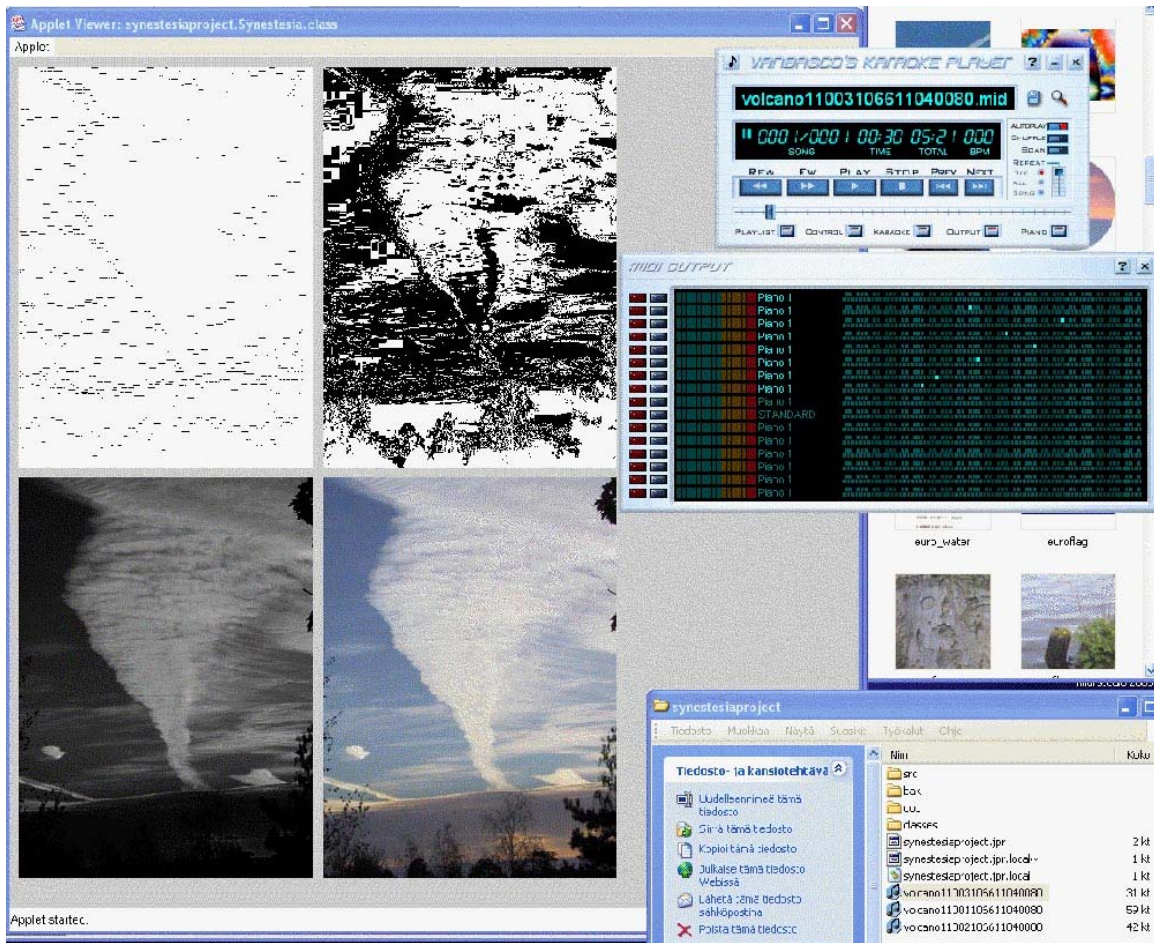
Liite 5: Midi tiedostosta saatu raakapartituuri

Liite 1: Kehitysympäristö käyttöliittymä parametrien valintavaiheessa



Keskellä Java-lähdekoodia JBuilder 9 ympäristössä. Generointi käynnistyy vasemman olevasta sinisellä maalatusa kohdasta ja generointi kestää muutaman sekunnin. Oikealla pala kuvahakemistoa, josta haluttu kuva (.jpg) voidaan poimia ja alhaalla oikealla kansio, johon midi-tiedosto generoituu.

Liite 2: Näyttö generoinnin tapahduttua



Vasemmassa puoliskossa alkuperäinen kuva, siitä muodostettu harmaasävykuva, josta varsinaisesti filterointii alkaa, ylempänä filteröity kuva ja ylävasemmalla kuvan ne pikselit, joita tässä generoinnissa on käytetty musiikin geneointiin. Kuvasta ei voi lukea sävelkorkeuksia kuin suuntaa-antavasti eikä kuvasta näy missä säveksen osassa kutakin pikseliä on käytetty, eikä siis ajanhetkeä, vaikka aika kulkeekin vasemmalta oikealta. Testiparametrin muutoksella saadaan erikseen näkyviin myös kussakin osassa käytetyt pikselit.

Liite 3: Midi-tiedostosta saatu raakapartituuri, esimerkkinä ”Time is Up”, sivu 1/16

[Title]

[Composer]

The image displays a MIDI score for the track "Time is Up". It consists of nine tracks, each with its own staff. Tracks 1, 2, 4, and 5 are in the treble clef, while tracks 3, 6, 7, 8, and 9 are in the bass clef. Track 1 features a complex melodic line with many beamed notes and slurs. Tracks 2, 4, and 5 contain rhythmic patterns of eighth notes. Track 3 has a more melodic line with some accidentals. Tracks 6, 7, and 8 show rhythmic patterns of eighth notes, with Track 8 having a more complex rhythmic structure. Track 9 is a simple bass line of eighth notes. The score is divided into measures by vertical bar lines, and the overall layout is clean and professional.

Liite 4: Raakapartituurista muokattu lähes muusikoiden soitettavissa oleva partituuri, sivu 1/16

Time is Up (18th March 2002)

Synesthesia Software
& Lauri Gröhn

The musical score is for the piece "Time is Up" (18th March 2002) by Synesthesia Software & Lauri Gröhn. It is in 3/8 time with a tempo of 60 BPM. The score consists of five staves: Flute, Bells, Drums, Piano, and a fifth empty staff. The Flute part starts with a dynamic of *f* and features a melodic line with some grace notes. The Bells part starts with a dynamic of *mf* and plays a rhythmic pattern of eighth notes. The Drums part starts with a dynamic of *p* and plays a simple drum pattern. The Piano part starts with a dynamic of *p* and plays a complex chordal accompaniment. A small red padlock icon is visible in the top right corner of the score.

Liite 5: Midi-tiedostosta saatu raakapartituuri, osa

The musical score shows a MIDI-derived raw score for a piano part. It consists of three staves: a treble clef staff, a middle treble clef staff, and a bass clef staff. The music is in 3/8 time and features a complex, rhythmic accompaniment with many beamed notes and chords. The score is presented as a raw MIDI export, with some unusual note groupings and accidentals.