

Vesa Hagström
vesa.hagstrom@helsinki.fi
Op.Nro: 013865575
Tiralabra harjoitustyö
16.6.2013

Simrec toteutusdokumentti

1 Ohjelman yleisrakenne

Simrec koostuu kahdesta luokasta ja kahdesta nimiavaruudesta, mitkä kaikki sisältyvät *simrec* nimiavaruuteen. Ohjelman algoritmit sijaitsevat nimiavaruudessa *simrec::algorithms* ja muuten hyödylliset funktiot ja vakiot – kuten π – *simrec::utils*:sta.

Data säilytetään yksinkertaisissa tietorakenteissa *Image* ja *ComplexArray*. *Image* tietorakenne on toteutettu *ComplexArray*:lla, eli *Image* on käytännössä vain kaksiulotteinen *ComplexArray*.

Ohjelman rakenne on kuvattu UML luokkakaaviossa kuvassa (??).

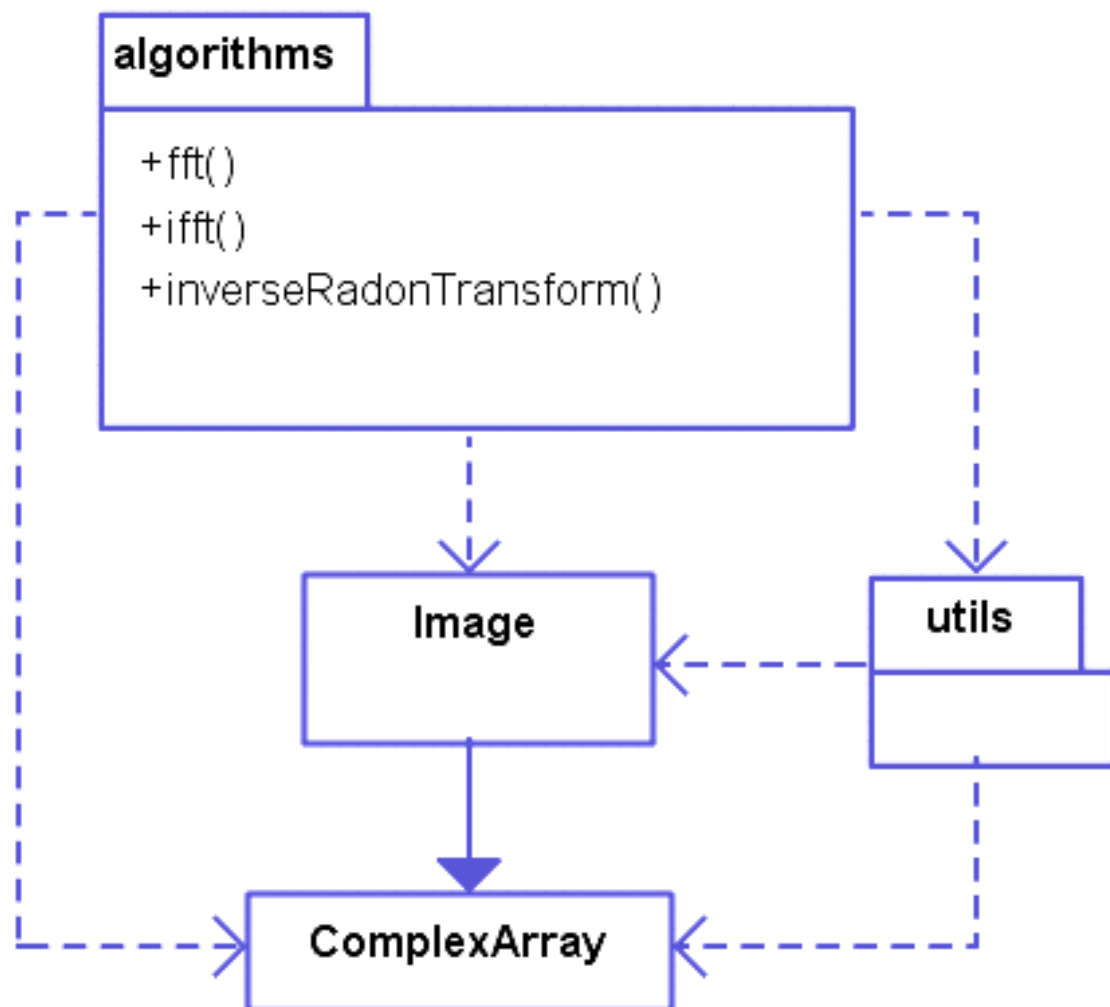
Ohjelmaan siis luetaan kuvadata, mikä tallennetaan kompleksilukuina *ComplexArray*-olioon. Tämän olion ja kuvan dimensioiden perusteella luodaan *Image*-olio. *Image*-olion avulla data skaalataan FFT-algoritmillemme sopivaan kokoon, eli sellaiseksi, että sen sivut ovat pituudeltaan pienin kahden potenssi, mikä on suurempi kuin kuvan pidemmän sivun pituus. Tämän jälkeen *Image*-olion data Fourier-muunnetaan, filtteriöidään ja Fourier-käänteismuunnetaan *simrec::algorithms*:ssa sijaitsevilla algoritmeilla. Tämä filtteriöity data sitten Radon-käänteismuunnetaan *simrec::algorithms*:n sisältämällä algoritmilla.

2 Saavutetut aika- ja tilavaativuudet

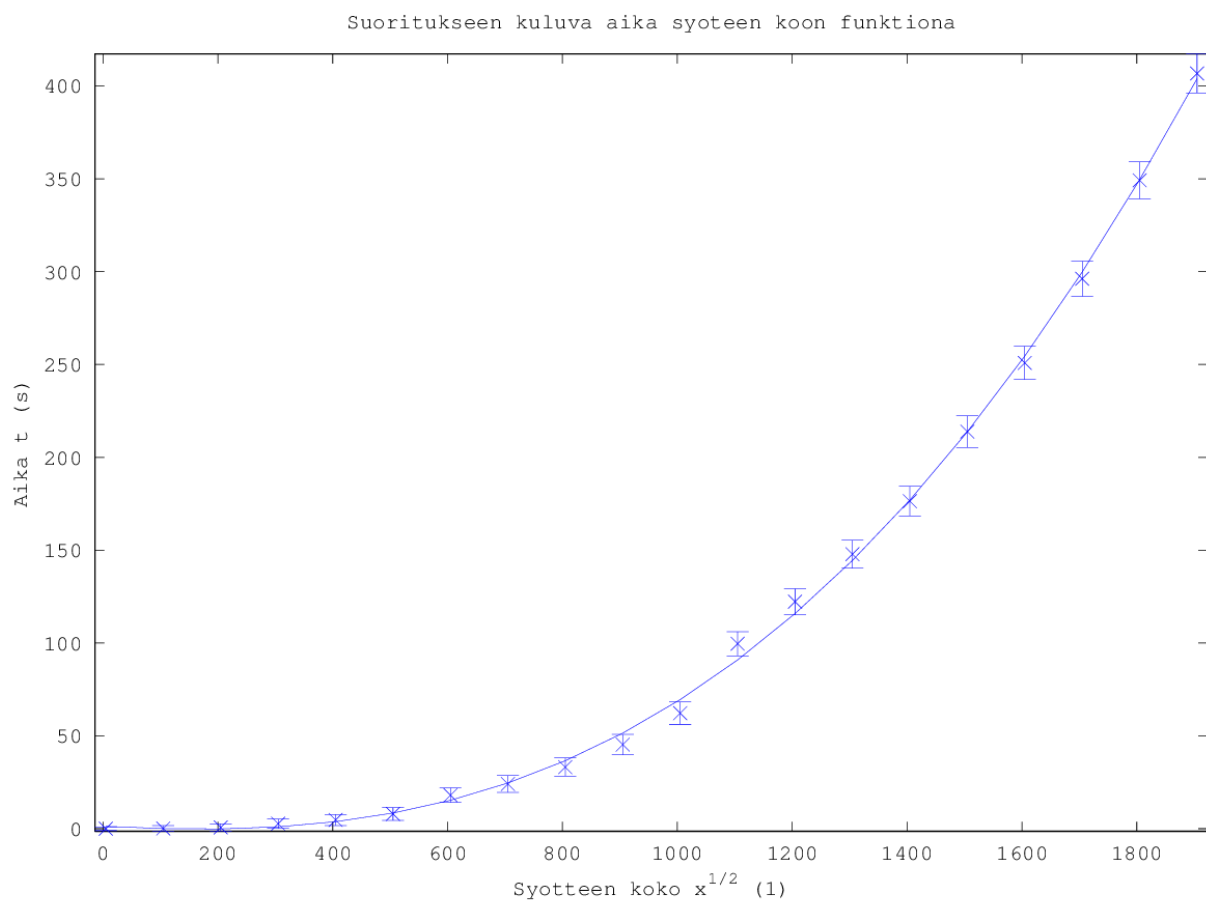
Kuvadatan filtteriöinnissä käytettävän Fourier-muunnoksen ja Fourier-käänteismuunnoksen aikavaativuudet algoritmista analysoituna ovat molemmat $O(n \log n)$, kuten FFT:n olettaisikin olevan. 2D FFT:ssä suoritetaan myös kaksi kuvamatriisin transponointioperaatiota, joiden aikavaativuudet ovat $O(\frac{n}{2}) \approx O(n)$. Kuvalle suoritettavat filtteriöinnit ovat aikavaativuudeltaan $O(n)$, sillä niissä käydään koko kuvadata läpi kertaalleen. Radon-käänteismuunnos on aikavaativuudeltaan luokkaa $O(n^3)$.

Näistä laskemalla $O(2 \cdot n \log n + n + n + n^3) \approx O(n^3)$. Algoritmin ajamiseen kuluva aika eri syöteteo'oilla näkyy mitattuna kuvassa (??).

Ohjelma tallentaa muistiin tiedostosta luetun kuvan, minkä tilavaativuus on $O(n)$. FFT ja iFFT pitävät ajonsa aikana muistissa palasia dataa, ja tilavaativuus näissä on sama kuin FFT:n/iFFT:n aikavaativuus, eli $O(n \log n)$. Radon-käänteismuunnos luo syötedatan koosta riippuvan uuden kuvan, minkä tilavaativuus on siis $O(n)$. Lopullinen tilavaativuus on siis laskettuna $O(n \log n + n + n) \approx O(n \log n)$.



Kuva 1: Simrec:n UML luokkakaavio



Kuva 2: Mitatut ajat eri kokoisilla syötteillä. $x=1000$ tarkoittaa, että algoritmi ajettiin kuvalla, minkä koko on $1000 \cdot 1000$.

3 Puutteet ja parannusehdotukset

Ohjelman algoritmeja voisi optimoida huomattavasti. Esimerkiksi Fourier-muunnoksien $e^{\pm 2\pi i A}$ -muotoiset kertoimet voisi laskea kerran etukäteen, minkä jälkeen niitä ei tarvitsisi laskea jokaisella FFT:n kierroksella erikseen. Tällä hetkellä algoritmeille myös kopioidaan kussakin kohdassa tarvittavan data (esim. 2D FFT:ssä kuvan rivit) kokonaisdatasta ja suoritetaan algoritmi tälle osalle, minkä jälkeen se syötetään takaisin alkuperäiseen paikkaansa. Tällä hetkellä algoritmit on toteutettu näin, koska se selkeyttää ohjelman rakennetta hieman.

Viitteet

- [1] Peter Toft, *"The Radon Transform - Theory and Implementation"*,
<http://petertoft.dk/PhD/>
- [2] J.W. Cooley, J.W. Tukey *"An algorithm for the machine calculation of complex Fourier series"*,
<http://dx.doi.org/10.1090/S0025-5718-1965-0178586-1>