Määrittelydokunetti Tietorakenteiden ja algoritmien harjoitustyö Pihla Toivanen

Ongelma

Kuvien pikseleiden arvot voivat olla suuria, ja yhdessä kuvassa vaikkapa jokainen arvo voi olla eri. Tällöin kuva vie paljon muistitilaa. On kuitenkin olemassa sekä häviöllisiä että häviöttömiä pakkausmetodeja tilantarpeen pienentämiseksi. Häviöllisissä pakkausalgoritmeissa kuvien pikseleihin sovelletaan matemaattisia muunnoksia, jotka myös pyöristävät arvoja. joten dataa häviää. Yleensä kuvat on kuitenkin tarkoitettu vain ihmissilmän katsottaviksi.

Algoritmi ja tietorakenteet

Toteutan työssäni JPEG -kuvanpakkausalgoritmin. Siihen kuuluu neljä vaihetta, joista jokaisen voi toteuttaa myös käänteisesti, joskin ei aina symmetrisesti.

1. **Esikäsittely**

   Esikäsittely ottaa parametrinaan kuvan, joka sisältää vain pikseleiden arvoja 8-bittisinä YCrCb -arvoina. Mikäli kuva ei ole tässä muodossa, sitä täytyy muokata. Kuva jaetaan 8*8 pikselin blokkeihin, ja vähennetään jokaisesta pikselistä luku 127.

2. **Transformaatio**

   Tavoitteenani on tehdä DCT – eli diskreetti kosinimuunnos. Siinä jokainen esikäsitelty blokki kerrotaan tietynlaisella kosinimatriisilla tämän muunnoksen kaavan mukaisesti.

3. **Arvojen muokkaus**

   Tässä vaiheessa arvot ovat liukulukuja, joten täytyy tehdä päätöksiä niiden pyöristämisestä ja muusta koodausta edeltävästä muokkauksesta. JPEGin nimi häviöllisenä pakkausalgoritmina tulee tästä vaiheesta, sillä arvojen pyöristyksen jälkeen niitä ei voida käänteisessä vaiheessa enää muuttaa täysin samoiksi.

4. **Koodaus**

   Arvot koodataan Huffman -koodauksella. Tunnistetaan pikselit, jotka esiintyvät usein, ja tehdään sanakirjapuu, jossa pikselit saavat frekvenssiensä mukaan uudet arvot.

Aika -ja tilavaativuudet

Transformaatiovaihe on O(n), missä n on pikselien määrä. Huffman-koodaus on myös O(n). En ole vielä perillä muiden vaiheiden aikavaativuuksista, mutta luultavasti nekin ovat polynomiaikaisia.

Lähteet:

http://stackoverflow.com/questions/6189765/big-o-complexities-of-algorithms-lzw-and-huffman
http://www.whydomath.org/node/wavlets/basicjpg.html
http://www.ams.org/samplings/feature-column/fcarc-image-compression
http://www.whydomath.org/node/wavlets/dct.html