

# **VÄRISPEKTRIKUVIEN TEHOKAS SIIRTO TIETOVERKOISSA**

Juha Lehtonen

20.3.2002

Joensuun yliopisto  
Tietojenkäsittelytiede  
Kandidaatintutkielma

## ESIPUHE

Olen kirjoittanut tämän kandidaatintutkielman Joensuun yliopistossa Tietojenkäsittelytieteen laitoksella lukuvuonna 2001–2002. Tutkielmani kokeellisen osan olen tehnyt työskennellessäni Joensuun yliopiston väritutkimusryhmässä kesällä 2000 ja keväällä 2001.

Haluan kiittää professori Jussi Parkkista kannustuksesta ja hyvistä neuvoista. Erityisesti haluan antaa kiitokset työni ohjaajalle yliassistentti Markku Hauta-Kasarille kärsivällisestä ja erinomaisesta opastuksesta, työni kriittisestä arvioinnista sekä arvokkaista kommentteista.

Lisäksi haluan kiittää vanhempiani, sukulaisiani ja ystäviäni, jotka ovat rohkaisseet minua tämän työn teossa. Lopuksi haluan antaa erityiskiitokset kihlatulleni Kaisalle hänen kannustuksestaan ja kärsivällisyydestään.

Joensuussa 20.3.2002

Juha Lehtonen

## TIIVISTELMÄ

Värikuvia käsitellään yleisesti kolme-arvoesityksinä, esim. RGB-kuvina. Värien esittäminen kolmella arvolla ei kuitenkaan aina riitä, sillä on olemassa tilanteita, missä ihminen ei erota kahden kohteen väriä toisistaan tietyssä valaistuksessa. Valaistusta kuitenkin muutettaessa väriero voidaan havaita. Lisäksi kolme-arvoesitykset ovat laiteriippuvaisia ja niitä katseltaessa digitaalisella näyttölaitteella ympäröivän valaistuksen ja näytön ominaisuuksien huomioon ottaminen on vaikeaa. Väristä voidaan kuitenkin fysikaalisesti mitata aallonpituuskäyrä eli spektri, minkä avulla em. ongelmia voidaan välttää. Värispektrikuvissa jokainen pikseli esitetään spektrillä, joka mittaustarkkuudesta riippuen tallennetaan kymmenien tai jopa satojen kanava-arvojen avulla. Tällaiset kuvat vievät hyvin paljon levytilaa, eikä niitä sellaisenaan voi siirtää nopeasti nykyisissä tietoverkoissa. Tässä tutkielmassa, joka koostuu teoriaosasta ja kokeellisesta osasta, esitellään arkkitehtuuri värispektrikuvien selailuun sekä erilaisia pakkauskeinoja värispektrikuville. Pakkaus toteutetaan spektri- ja spatiaalitasossa. Spektritasossa pakkaus toteutetaan pääkomponenttianalyysillä (PCA). Spatiaalinen pakkaus tehdään myös tavallisilla menetelmillä, sillä samaan tapaan pakataankin esim. JPEG-kuvat. Lisäksi esittelen spatiaalisessa pakkauksessa ns. keskiarvo- ja mediaanimenetelmän, mitkä poikkeavat hieman tavallisista menetelmistä. Tutkielmassa vertaillaan eri pakkausmenetelmiä sekä tutkitaan pakattujen kuvien virheellisyyttä eri virhemittausmenetelmillä, tiedostokokoa, sekä siirrettävyyttä tietoverkoissa.

Avainsanat: kuvakompressio, selailu, tiedonsiirto, värispektri, värispektrikuva.

## SISÄLLYSLUETTELO

<b>1. JOHDANTO</b> .....	<b>1</b>
<b>2. VÄRISPEKTRIKUVAT JA NIIDEN SELAILUN ARKKITEHTUURI</b> .....	<b>3</b>
2.1 Värispektrikuvien hyötyjä	3
2.2 Selailun arkkitehtuuri	5
<b>3. VÄRISPEKTRIKUVIEN PAKKAUS JA PURKU</b> .....	<b>8</b>
3.1 Spektrien suodatus	8
3.2 Spektritason pakkaus	9
3.2.1 Pääkomponenttianalyysi PCA	9
3.2.2 Sisätulokuvien muodostus	10
3.3 Spatiaalitason pakkaus	10
3.4 Pakatun kuvan purku	12
3.5 Keskiarvomenetelmän aiheuttama virhe	13
3.6 Mediaanimenetelmän aiheuttama virhe	14
<b>4. VIRHEMITTAUSMENETELMÄT</b> .....	<b>16</b>
4.1 Virhekaavat	16
4.2 Värivirheet ja S-CIELAB	18
<b>5. TESTIAINEISTOT</b> .....	<b>19</b>
5.1 Bristol-kuvasarja	19
5.2 Forest- ja Coral-kuvasarjat	19
<b>6. VIRHELASKENNAT</b> .....	<b>21</b>
6.1 Spektrien suodatus	21
6.2 Spektritason pakkaus	22
6.3 Spatiaalitason pakkaus	23
6.4 Keskiarvomenetelmä	27
6.5 Mediaanimenetelmä	29
6.6 Keskiarvomenetelmä vs. mediaanimenetelmä	30
6.7 Bristol-kuvasarjan virheitä	32
<b>7. PAKKAUSSUHDE JA SIIRTO TIETOVERKOISSA</b> .....	<b>33</b>
<b>8. POHDINTA</b> .....	<b>36</b>
8.1 Työn kulku	36
8.2 Työn analysointi	36

**LIITTEET**

1. Kuvien pakkaus ja purku
2. Spatiaalisen pakkauksen aiheuttamaan virheeseen liittyvät todistukset
3. Kuvaaja:  $2 \times 2$ -/ $1 \times 4$ -keskiarvomenetelmä vs.  $4:2:0$ -/ $4:1:1$ -menetelmä
- 4a. Bristol-kuvasarja
- 4b. Forest-kuvasarja
- 4c. Coral-kuvasarja
5. Pohdintaesimerkkejä

## 1. JOHDANTO

Värikuvia käsitellään yleisesti ihmissilmän värinäköön perustuvilla kolmearvoesityksillä, missä jokainen pikseli esitetään kolmella arvolla (esim. RGB-kuvat). Kolmearvoesityksiin liittyy kuitenkin useita ongelmia, esimerkiksi se voi joissakin tilanteissa aiheuttaa *metameerisyyden* [27], missä ihminen ei erota kahden eri kohteen väriä tietyssä valaistuksessa. Valaistusta kuitenkin muutettaessa kohteiden väillä oleva väriero voidaan havaita.

Värispektrikuvissa jokainen pikseli esitetään spektrillä esimerkiksi näkyvän valon alueella (400...700 nm) 5 nm välein, jolloin yksi pikseli esitetään 61 arvolla. Tällaiset kuvat poistavat kolmearvoesityksiin liittyviä ongelmia, esimerkiksi metameerisyyden. Nykyään spektrikuva-tietokantojen määrä onkin kasvamassa, kuvaustekniikat kehittyvät ja ns. moniprimäärinäytöt, missä väri voidaan esittää jopa kuudella päävärillä, ovat tulossa. Spektrin tallennustarkkuudesta riippuen värispektrikuvat saattavat viedä hyvinkin paljon levytilaa, jolloin niitä ei voida sellaisenaan tehokkaasti siirtää tietoverkoissa.

Nykyään on kehitetty useita erilaisia pakkausmenetelmiä värispektrikuville. Tähän ei vielä kuitenkaan ole olemassa yhtenäistä standardia. Kondou et al. [12] käsittelevät pääkomponenttialyysiin PCA sekä JPEG-pakkausmuodon diskreettiin kosinimuunnokseen perustuvaa menetelmää. Kaarna [11] käsittelee Wavelet-muunnokseen perustuvaa menetelmää, johon mm. JPEG 2000 -järjestelmä perustuu. König et al. [15,16] esittelevät tehokasta ja tavanomaisiin kuvaformaatteihin yhteensopivaa TIFF-tallennusmuotoon perustuvaa pakkausmenetelmää. Lisäksi Derbyn yliopistossa MacDonald et al. [19] sekä Liu [17] ovat tutkineet pääkomponenttialyysiin perustuvaa spektritason pakkausta sekä myös JPEG-kuvapakkaukseen perustuvaa spatiaalitason pakkausta. He toteavat viitteessä [18], että värispektrikuva on mahdollista pakata spektritason PCA-menetelmällä erittäin pieneen tilaan siten, että kuvasta saadaan vielä purettaessa kelvollinen. Tämän lisäksi kuvaa voidaan vielä pakata spatiaalitasossa JPEG-pakkausmuotoon perustuvien menetelmin. PCA-menetelmään perustuvaa pakkausta on tehty myös spektrivideolle [13].

Tässä tutkielmassa värispektrikuvan pakkaus tehdään ensin spektritason, sen jälkeen spatiaalitasossa. Spektritason pakkaus toteutetaan pääkomponenttialyysillä. Tämän lisäksi esitän useita erilaisia spatiaalisia pakkausmenetelmiä, jotka perustuvat JPEG- ja MPEG-muotoon. Tutkielmani tuloksia on jo aikaisemmin julkaistu alan kansainvälisissä konferens-

seissa [7,22]. Lisäksi esitän näihin pakkausmenetelmiin perustuen uuden arkkitehtuurin värispektrikuvien *selailuun*, mikä mahdollistaa nopean standardin tavan hakea värispektrikuvia internetissä.

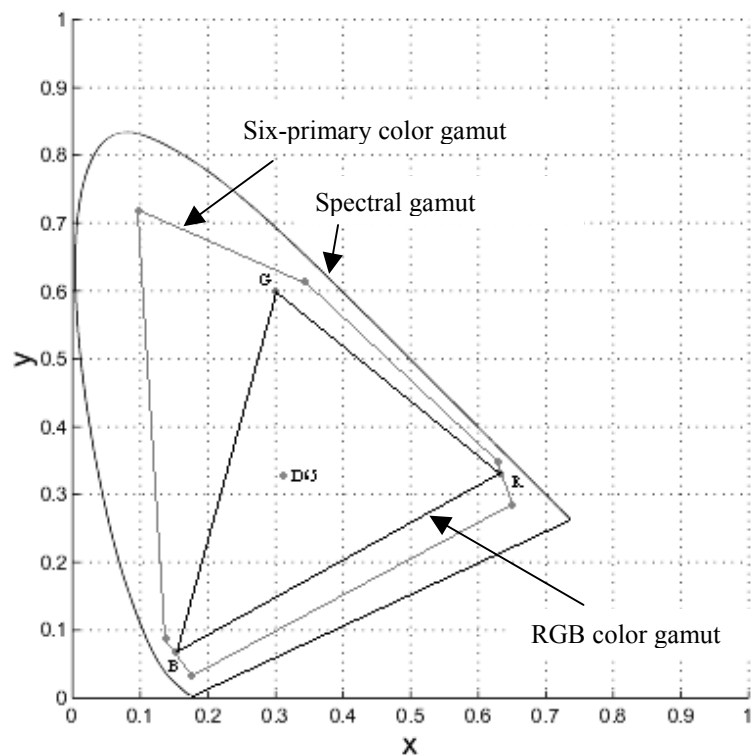
Käsittelen *luvussa 2* värispektrikuvien hyötyjä sekä uuden arkkitehtuurin värispektrikuvien selailuun. Varsinaiset pakkausmenetelmät suoritusjärjestyksessä esitän *luvussa 3*. *Luvussa 4* on selitetty kokeellisessa osassa käyttämäni eri virhemittausmenetelmät. Tutkielmani kokeellinen osa alkaa varsinaisesti *luvusta 5*, missä esittelen testiaineiston. *Luvussa 6* on pyritty laajasti selvittämään eri pakkausmenetelmien aiheuttamia virheitä testiaineiston avulla. Eri pakkausmenetelmien pakkaussuhteet ja tiedonsiirrossa kuluvat ajat on esitelty *luvussa 7*. Lopuksi pohdin tuloksia *luvussa 8*.

## 2. VÄRISPEKTRIKUVAT JA NIIDEN SELAILUN ARKKITEHTUURI

Tässä luvussa selvitan lyhyesti värispektrikuvien hyötyjä käytännössä. Värispektrikuvia ei kuitenkaan voi tehokkaasti siirtää nykyisissä tietoverkoissa niiden tilavaativuuden takia, joten esitän myös uuden arkkitehtuurin värispektrikuvien selailuun.

### 2.1 Värispektrikuvien hyötyjä

Kolmeväriesitysten suurimpana ongelmana on metameerisyys. Tämän lisäksi kolmeväriesitykset ovat laiteriippuvaisia ja niitä katseltaessa digitaalisella näyttölaitteella ympäröivän valaistuksen ja näytön ominaisuuksien huomioon ottaminen on vaikeaa [3]. Nykyisten kolmeväriesitykseen perustuvien näyttölaitteiden esityskyky on myös rajoittunut Kuvassa 2.1 havainnollistetaan CIE 1931  $xy$  –värikoordinaatistossa RGB-väritoistoalue, eli värigamut. Spektraalinen värigamut on koko spektrialueen esityskyky välillä 380...780 nm.

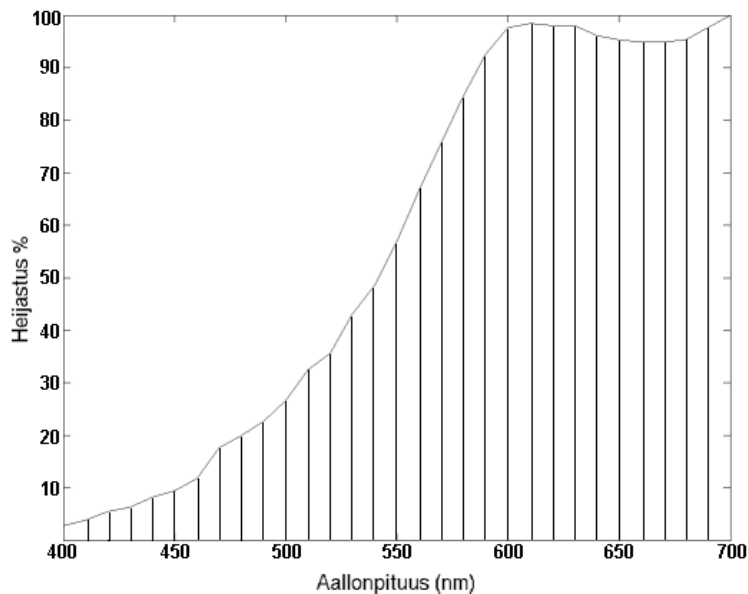


Kuva 2.1: RGB-, moniprimäärinäytön ja spektraalinen väritoistoalue CIE 1931  $xy$  –värikoordinaatistossa.

Kolmeväriesitysten ongelmia voidaan välttää esittämällä väri tarkasti. Tämä voidaan tehdä fyysisesti mittaamalla värin aallonpituus. Värispektrikuvissa jokainen pikseli esitetään tällaisella värispektrillä, minkä arvot voidaan tallentaa tietokoneelle esim. 10 nm välein näkyvän valon alueelta 400...700 nm. Tällöin spektri esitettäisiin 31 kanava-arvolla (kuva 2.2).



Värispektrikuville kehitetään parhaillaan uusia näyttölaitteita. Viime aikoina onkin julkaistu ns. moniprimäärinäyttö, missä kuuden päävärin avulla saadaan laajennettua näytön väriesityskykyä huomattavasti suuremmaksi (kuva 2.1, Six-primary color gamut). Moniprimäärinäyttöjä kehitetään koko ajan eteenpäin, joten tulevaisuudessa tämä mahdollistaa värikuvien näyttämisen entistä tarkemmin [1].



Kuva 2.2: Esimerkki värispektristä ja sen tallennuksesta.

Koska värispektrikuvan jokaisessa pikselissä on spektri, vie se levytilaa jopa useita megatavuja, joten niitä ei pysty sellaisenaan siirtämään tehokkaasti nykyisissä tietoverkoissa. Taulukossa 2.3 on esimerkki eri kuvatyyppien tilavaativuoksista, kun jokainen pikseliarvo vie yhden tavun. Nämä kuvat voidaan kuitenkin häviöllisesti pakata hyvinkin pieneen tilaan siten, että kuvien laatu säilyy vielä kelvollisena.

Taulukko 2.3: Eri kuvatyyppien tilavaativuus.

Kuvan koko	256×256 pikseliä	512×512 pikseliä
Harmaasävykuva	64 Kt	256 Kt
RGB-kuva	192 Kt	768 Kt
Värispektrikuva, 20 nm resoluutio	1 Mt	4 Mt
Värispektrikuva, 5 nm resoluutio	4 Mt	15 Mt
Spektrivideo, 20nm resoluutio, 30 kuvaa/s, 10 s.	300 Mt	1 Gt

Käytännön sovelluksia värispektrikuville ja niiden tiedonsiirtoon ovat mm.

**Etälääketiede**, missä potilas ja lääkäri ovat etäällä toisistaan. Lääkäri voi tehdä tarkkojen värikuvien avulla havaintoja potilaasta internet-verkon yli.

**Internet-kaupat**, missä henkilö voi tilata esimerkiksi kotiinsa sopivan tapetin. Tapetti näyttää seinällä täsmälleen samanlaiselta kuin tietokoneruudulla.

**Nettitaidekaupat ja digitaaliset museot**, missä asiakas voi katsella teoksia selaimella.

**Osittain haalistuneet maalaukset** voidaan restauroida (lähes) alkuperäisiksi vertailemalla teoksen haalistuneita ja haalistumattomia kohtia. Maalausten aitoutta voidaan myös tutkia värispektrikuvien avulla.

**Tarkka laadunvalvonta**, missä kone tutkii ottamastaan värispektrikuvasta tuotteen laadun, kuten esim. paperiteollisuudessa, missä valmistetun paperin värin tulee olla tasainen ja oikea.

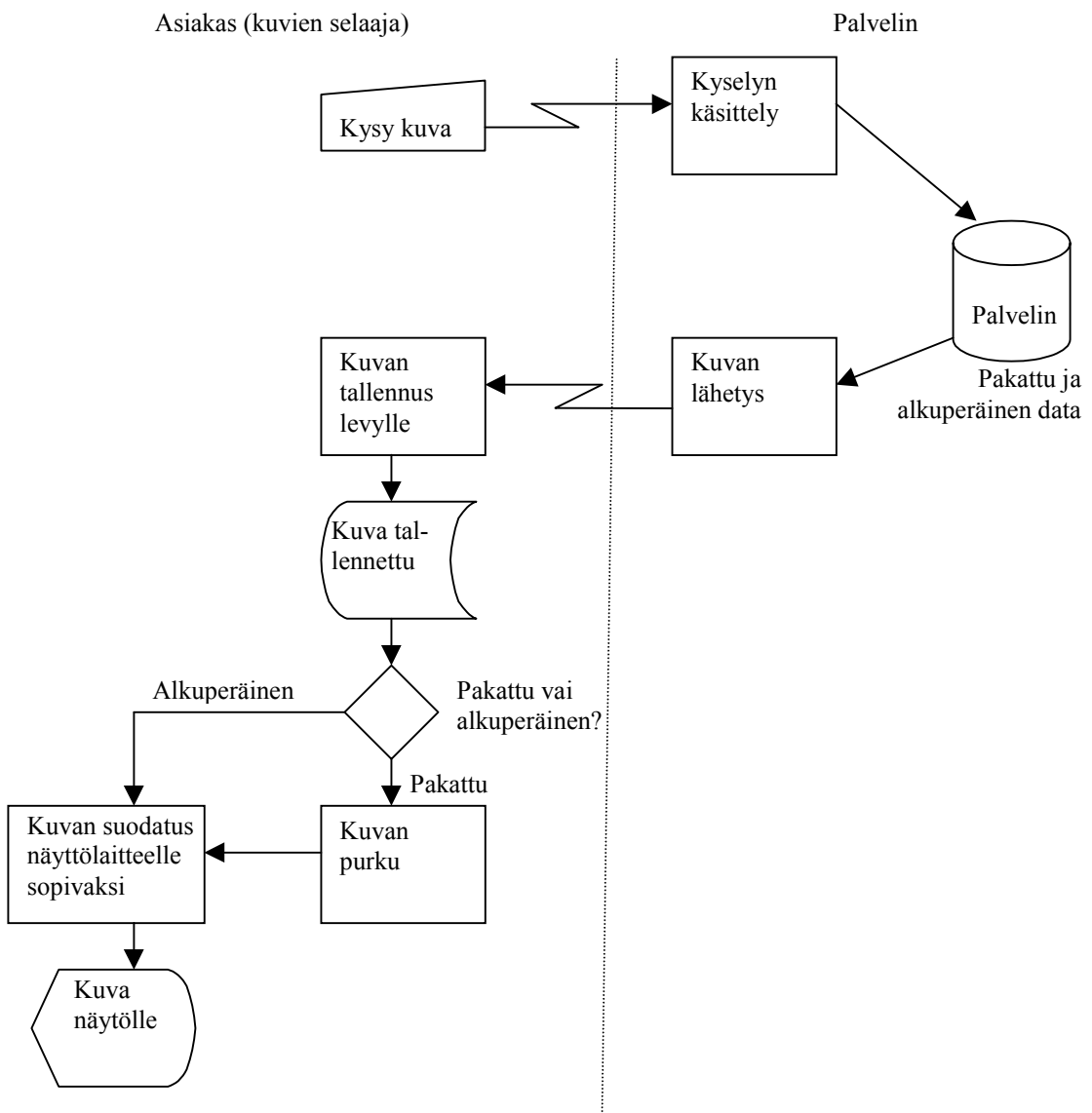
## 2.2 Selailun arkkitehtuuri

Spektrikuvien määrä kasvaa nopeasti. Vielä ei kuitenkaan ole sovittu standardia esitysmuotoa spektrikuville, sillä niitä on pakattu mm. TIFF-muotoon perustuvien pakkausmenetelmin [15,16] ja JPEG-muotoon perustuvien menetelmin [12]. Värispektrikuvat vievät pakkaamattomana niin paljon levytilaa, että nykyiset järjestelmät eivät sovellu näiden kuvien tiedonsiirtoon. Seuraavaksi esitellään uusi arkkitehtuuri värispektrikuvien selailuun.

Värispektrikuvat pakataan eli *kompressoidaan* palvelimelle siten, että niiden siirtämiseen nykyisten menetelmillä kuluu huomattavasti vähemmän aikaa. Kompressio tapahtuu siten, että värispektrikuvasta muodostetaan *kantavektorit* ( $k$  kpl) pääkomponenttianalyysillä PCA, minkä jälkeen kantavektorien avulla muodostetaan alkuperäisestä kuvasta sisätulokuvia  $k$  kpl. Ensimmäinen sisätulokuva jätetään ennalleen, mutta muut sisätulokuvat kompressoidaan vielä spatiaalisesti siten, että kuva jaetaan pieniin lohkoihin, missä jokaisessa lohkoissa yksi arvo korvaa koko lohkon. Pakatut sisätulokuvat (ensimmäinen pakkaamatonkin mukaan lukien) ja kantavektorit muodostavat kompressoitun värispektrikuvan. Tästä prosessista kerrotaan tarkemmin luvussa 3. Jo muutamalla sisätulokuvalla (esim. 3 kpl) voidaan moniin tarkoituksiin saada jo kelvollinen tulos. Tilavaativuus saattaa sopivilla sisätulokuvien määrällä ja spatiaalisen kompression lohkokokoilla olla vain muutama prosentti verrattuna pakkaamattomaan värispektrikuvaan (ks. luku 7).

Käytännössä selailu voidaan toteuttaa siten, että sekä kompressoitut että alkuperäiset värispektrikuvat sijaitsevat palvelimella. Kuvien selaaja hakee tietokoneellensa kompressoitun

kuvan, mikä puretaan (l. *rekonstruoidaan*) hänen omalla tietokoneellansa. Lisäksi kuvat suodatetaan hänen omalle näyttölaitteelleen sopivaksi. Suodatus mahdollistaa kuvien vastaanottamisen myös esim. kannettavalla tietokoneella ja matkapuhelimella, jolloin kuvat näyttävät kaikilla laitteilla samalta. Mikäli kuva ei ole haluttu, kuva poistetaan, ja selaaja voi siirtyä palvelimella seuraavaan kuvaan. Lopuksi selaaja voi imuroida haluamansa kuvan pakkaamattomana, joka myös suodatetaan hänen näytölleen sopivaksi (kuva 2.3). Pakkaamatonta kuvaa voi käyttää esim. laskentaan, missä vaaditaan alkuperäisen kuvan tarkkuus. Näytön sijaan laitteena voi olla myös tulostin.



Kuva 2.3: Värispektrikuvien selailun arkkitehtuuri.

Värispektrikuvien hakua tietokannasta voidaan kehittää edelleen siten, että selaaja käyttää hyväkseen kantavektoreita, mikä kertoo palvelimelle, minkälaisia värejä kuvissa on. Kanta-

vektorien tilavaativuus on vain promilleluokkaa pakkaamattomaan värispektrikuvaan nähden. Tämä vähentää selailun eri vaihtoehtoja, jotka selaaja voi pakattuina nopeasti käydä läpi. Tämän jälkeen selaaja voi imuroida halutun kuvan pakkaamattomana.

### 3. VÄRISPEKTRIKUVIEN PAKKAUS JA PURKU

Värispektrikuvien kompressointi ja rekonstruointi koostuvat useasta eri osasta. Seuraavassa on käsitelty kaikki osat suoritusjärjestyksessä: *suodatus*, *spektritason pakkaus*, *spatiaalinen pakkaus ja pakatun kuvan purku*. Koko pakkausprosessi on tiivistetysti kuvattu liitteessä 1. Liu [17] sekä MacDonald et al. [18, 19] käsittelevät samankaltaisia menetelmiä keskittyen pääkomponenttianalyysiin perustuviin spektritason pakkausmenetelmiin, mutta he esittelevät myös spatiaalitason pakkausta, minkä eri vaihtoehtoja pohdin tässä tutkielmassa enemmän.

#### 3.1. Spektrien suodatus

Suodatus tasoittaa spektrejä, joten mahdolliset kuvausvirheet (esim. äkilliset muutokset) vähenyvät huomattavasti. Äkillinen muutos, kuten piikki, ei tosin aina ole virhe, mutta suodatuksessa jää piikistä informaatiota spektriin. Tällä tavalla päästään lähemmäksi todellista kuvaa ja kuvauksesta tai kuvausolosuhteista aiheutuneet virheet jäävät vähemmälle huomiolle.

Kuvan jokainen spektri suodatetaan *keskiarvostuksella*, missä lasketaan kanava-arvo,  $x$  edellistä arvoa ja  $x$  seuraavaa arvoa yhteen, ja jaetaan se näiden arvojen lukumäärällä (eli lasketaan keskiarvo), minkä tulos sijoitetaan uudeksi kanava-arvoksi. Tämä suoritetaan spektrin kaikille arvoille  $x$  ensimmäistä ja  $x$  viimeistä arvoa lukuun ottamatta.

Olkoon  $i$ :s spektri

$$\bar{s}_i = (s_{1,i}, s_{2,i}, \dots, s_{c,i}), \quad (3.1)$$

missä  $c$  on spektrin kanavien lukumäärä. Olkoon  $x$  lukumäärä siitä, kuinka monta kanava-arvon edellistä ja seuraavaa arvoa lasketaan mukaan keskiarvoon ( $x = \{1, 2, 3, \dots, \frac{c}{2} - 1\}$ ).

Spektrin kanava-arvot lasketaan uudelleen kaavalla

$$s_{j,i} = \frac{\sum_{y=j-x}^{j+x} s_{y,i}}{2 \cdot x + 1}, \text{ missä } j = x+1, x+2, \dots, c-x. \quad (3.2)$$

Spektrin  $x$  ensimmäistä ja  $x$  viimeistä kanava-arvoa jätetään ennalleen.

## 3.2 Spektritason pakkaus

Ensimmäisessä kompressiovaiheessa kuvat pakataan spektritasossa *pääkomponenttianalyysin* (PCA = *Principal Component Analysis*) [20] avulla [17]. Pakkausta on tehty myös *riippumattoman komponenttianalyysin* (ICA = *Independent Component Analysis*) [9,25] avulla [7], minkä tulokset ovat olleet samantapaisia kuin PCA:n.

### 3.2.1 Pääkomponenttianalyysi PCA

Olkoon värispektrikuvan leveys  $n$ , korkeus  $m$  ja jokaisessa spektrissä kanavia  $c$  kpl. Esitetään tämä värispektrikuva 2-ulotteisena siten, että kaikki pikselit (eli spektrit) ovat yhtenä listana peräkkäin. Olkoon  $S = (\bar{s}_1 \ \bar{s}_2 \ \dots \ \bar{s}_{n \cdot m})$  keskiarvostuksella suodatettu värispektrikuva ja tämän  $i$ :s spektri  $\bar{s}_i = (s_{1,i} \ s_{2,i} \ \dots \ s_{c,i})^T$ , missä  $s_{j,i}$  on spektrin  $j$ :s kanava-arvo. Olkoon myös  $R$  kuvan  $S$  korrelaatiomatriisi [10]

$$R = \frac{S \cdot S^T}{n \cdot m} = \frac{\sum_{i=1}^{n \cdot m} s_i s_i^T}{n \cdot m}. \quad (3.3)$$

Tällöin  $R$  on  $c \times c$ -kokoinen neliömatriisi, minkä karakteristinen polynomi on muuttujan  $e$  polynomi

$$\det(R - e \cdot I) = \begin{vmatrix} r_{1,1} - e & r_{1,2} & \dots & r_{1,c} \\ r_{2,1} & r_{2,2} - e & \dots & r_{2,c} \\ \vdots & \vdots & \ddots & \vdots \\ r_{c,1} & r_{c,2} & \dots & r_{c,c} - e \end{vmatrix}, \quad (3.4)$$

missä  $I$  on yksikkömatriisi. Yhtälö

$$\det(R - e \cdot I) = 0 \quad (3.5)$$

on  $c$ . asteen polynomiyhtälö, minkä ominaisarvot  $e_1 \dots e_c$  ovat sen nollakohdat. Vastaavasti voidaan määrittellä  $c$  kpl ominaisvektoreita  $\bar{e}_1 \dots \bar{e}_c$  yhtälöryhmästä

$$R \cdot \bar{e} = e \cdot \bar{e} \Leftrightarrow (R - e \cdot I) \cdot \bar{e} = 0. \quad (3.6)$$

Suurimpia ominaisarvoja  $e_1 \dots e_k$  (missä  $k \in [1..c]$  ja  $e_1 \geq e_2 \geq \dots \geq e_k$ ) vastaavat ominaisvektorit ovat spektrikuvan  $S$  kantavektoreita ja siten *kanta*

$$B = (\bar{e}_1 \ \bar{e}_2 \ \dots \ \bar{e}_k), \quad (3.7)$$

jonka lineaarisella kombinaatiolla alkuperäinen data voidaan rekonstruoida [8,21,23].

### 3.2.2 Sisätulokuvien muodostus

Kannasta  $B$  ja alkuperäisestä kuvasta  $S$  saadaan muodostettua  $k$  kpl sisätulokuvia, missä  $k$  on kantavektoreiden määrä. Tällöin 2-ulotteisesti yhtenä listana esitettynä sisätulokuvat

$$P = (\bar{p}_1 \quad \bar{p}_2 \quad \dots \quad \bar{p}_{n \cdot m}) = B^T \cdot S, \quad (3.8)$$

missä  $\bar{p}_i = (p_{1,i} \quad p_{2,i} \quad \dots \quad p_{k,i})^T$ . Spektritason pakkauksen jälkeen kompressoidun värispektrikuvan muodostavat yhdessä sisätulokuvat  $P$  ja kanta  $B$ .

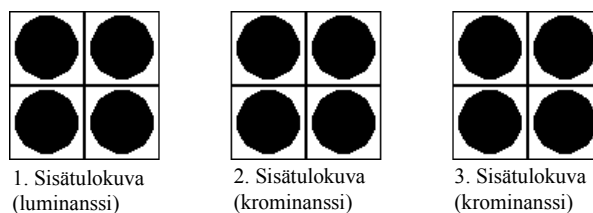
### 3.3 Spatiaalitason pakkaus

Luvussa 3.2 selitetyssä komponenttianalyysissä saadaan sisätulokuvia  $k$  kpl. Näiden kuvien spatiaalinen pakkaus käy hyvin samaan tapaan kuin JPEG- ja MPEG-kuvien pakkaus [26]. JPEG-muodossa RGB-kuvat muunnetaan  $YCbCr$ -värikoordinaatistoon, missä  $Y$  vastaa kuvan harmaasävyjä,  $Cb$  kuvan sinisiä värisävyjä ja  $Cr$  kuvan punaisia värisävyjä. Kromaattisia kanavia  $Cb$  ja  $Cr$  voidaan pakata spatiaalisesti ihmissilmän värinäköön pohjautuen menettämättä ratkaisevasti kuvan informaation sisältöä. Ensimmäinen sisätulokuva on lähellä JPEG-kuvan harmaasävyä (*luminanssikanava*) ja muut sisätulokuvat vastaavat JPEG-kuvan värisävyjä (*krominanssikanavat*).

Spatiaalisessa pakkauksessa ensimmäinen sisätulokuva säilytetään ennallaan, mutta muut sisätulokuvat (2.. $k$ ) jaetaan spatiaalisesti  $x \times y$  -kokoiisiin lohkoihin. Jokainen lohko korvataan lohkon kulma-arvolla (lohkon vasemmassa yläkulmassa oleva arvo), lohkon keskellä sijaitsevalla arvolla, lohkon arvojen keskiarvolla tai lohkon arvojen mediaanilla [7,22]. Seuraavaksi esiteltävät eri menetelmät määrittelevät lohkokoon ja lohkon korvaavan pikselin. Kuvissa lohkon mustalla värjätyt solut korvaavat lohkon.

#### 4:4:4-menetelmä

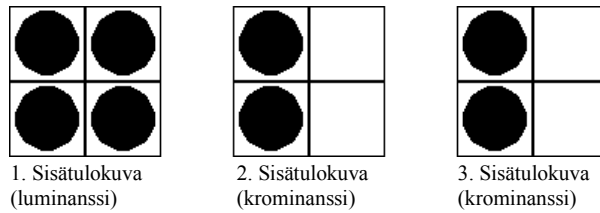
Sisätulokuvat (2.. $k$ ) jaetaan  $1 \times 1$ -kokoiisiin lohkoihin. Kunkin lohkon kulma-arvo korvaa lohkon. Käytännössä tämä ei kompressoi kuvaa spatiaalisesti, sillä lohkoissa on vain yksi pikseli, jolloin korvaava pikseli on sama kuin alkuperäinen.



Kuva 3.1: 4:4:4-menetelmä ( $k=3$ ).

#### 4:2:2-menetelmä

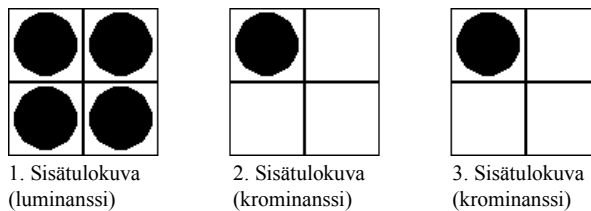
Sisätulokuvat (2..k) jaetaan  $1 \times 2$ -kokoisiin lohkoihin. Kunkin lohkon kulma-arvo (eli vasemmanpuoleinen arvo) korvaa lohkon.



*Kuva 3.2: 4:2:2-menetelmä ( $k=3$ ).*

#### 4:2:0-menetelmä

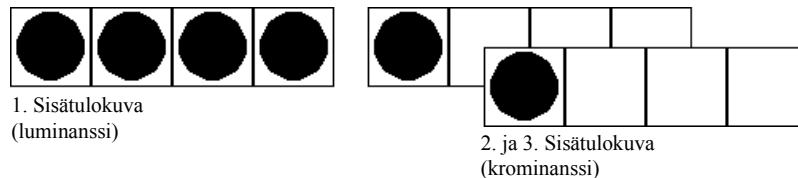
Sisätulokuvat (2..k) jaetaan  $2 \times 2$ -kokoisiin lohkoihin. Kunkin lohkon kulma-arvo (eli lohkon vasemmassa yläkulmassa oleva arvo) korvaa lohkon.



*Kuva 3.3: 4:2:0-menetelmä ( $k=3$ ).*

#### 4:1:1-menetelmä

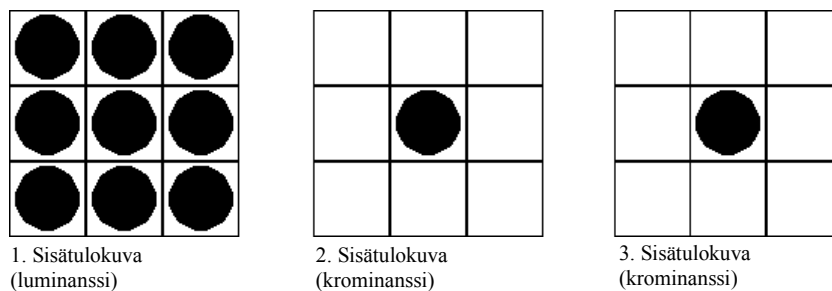
Sisätulokuvat (2..k) jaetaan  $1 \times 4$ -kokoisiin lohkoihin. Kunkin lohkon kulma-arvo (eli vasemmanpuoleisin arvo) korvaa lohkon.



*Kuva 3.4: 4:1:1-menetelmä ( $k=3$ ).*

#### 3×3-keskipistemenetelmä

Sisätulokuvat (2..k) jaetaan  $3 \times 3$ -kokoisiin lohkoihin. Kunkin lohkon keskimäinen arvo (eli lohkon keskellä oleva arvo) korvaa lohkon.

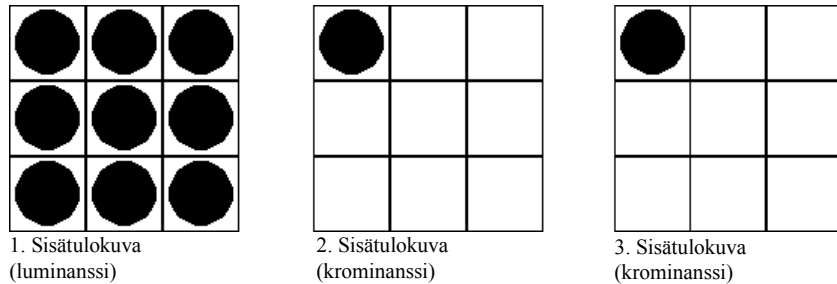


*Kuva 3.5: 3×3-keskipistemenetelmä ( $k=3$ ).*



### 3×3-kulmapistemenetelmä

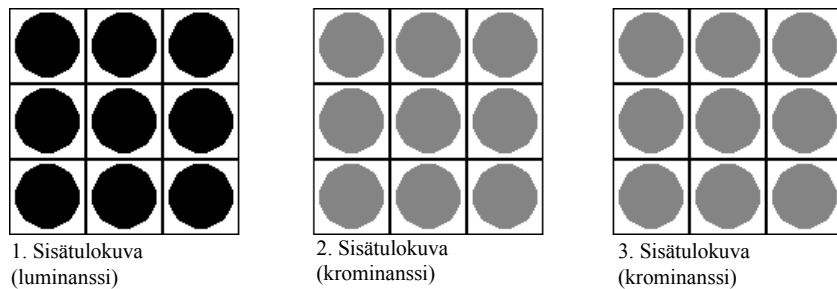
Sisätulokuvat (2..k) jaetaan 3×3-kokoisiin lohkoihin. Kunkin lohkon kulma-arvo (eli lohkon vasemmassa yläkulmassa oleva arvo) korvaa lohkon.



Kuva 3.6: 3×3-kulmapistemenetelmä ( $k=3$ ).

### $x \times y$ –keskiarvomenetelmä

Sisätulokuvat (2..k) jaetaan  $x \times y$  -kokoisiin lohkoihin. Kunkin lohkon arvojen keskiarvo korvaa lohkon. Seuraavassa kuvassa sisätulokuvat on jaettu 3×3-kokoisiin lohkoihin. Harmaalla värjättyt solut kuvaavat lohkon keskiarvoa. Ks. luku 3.5.



Kuva 3.7: 3×3-keskiarvomenetelmä ( $k=3$ ).

### $x \times y$ –mediaanimenetelmä

Kuten  $x \times y$  –keskiarvomenetelmä, mutta keskiarvon sijaan käytetään mediaania. Ks. luku 3.6.

Ensimmäinen sisätulokuva, spatiaalisesti pakatut sisätulokuvat sekä kanta muodostavat kompressoitua värispektrikuvan.

### 3.4 Pakatun kuvan purku

Luvussa 3.3 esitetyn spatiaalisen pakkauksen eri menetelmissä jokin tietty arvo korvaa lohkon sisätulokuvassa. Purettaessa kuva korvataan lohkon kaikki arvot siis tällä tallennetulla arvolla. Näin saadaan spatiaalisesti puretut sisätulokuvat

$$\tilde{P} = (\tilde{p}_1 \quad \tilde{p}_2 \quad \dots \quad \tilde{p}_k)^T = (\tilde{p}_1 \quad \tilde{p}_2 \quad \dots \quad \tilde{p}_{n_m}), \quad (3.9)$$

missä  $\tilde{P}_a$  on  $a$ :s purettu sisätulokuva ja  $\tilde{p}_i = (p_{1,i} \quad p_{2,i} \quad \dots \quad p_{k,i})^T$ . Ensimmäinen sisätulokuva ei pakkauksessa muutu, joten  $\tilde{p}_1 = P_1$ . Näiden sisätulokuvien ja kannan  $B = (\bar{e}_1 \quad \bar{e}_2 \quad \dots \quad \bar{e}_k)$  avulla saadaan purettua  $i$ :s spektri

$$\bar{s}_i = \sum_{j=1}^k \bar{e}_j \cdot \tilde{p}_{j,i}. \quad (3.10)$$

Rekonstruoitu värispektrikuva on siten

$$\tilde{S} = (\bar{s}_1 \quad \bar{s}_2 \quad \dots \quad \bar{s}_{n \cdot m}). \quad (3.11)$$

Virheellisyyttä alkuperäisen ja rekonstruoidun kuvan välillä voidaan mitata useilla eri menetelmillä (luku 4).

### 3.5 Keskiarvomenetelmän aiheuttama virhe

Olen tutkinut kokeellisesti menetelmää, jossa lohko korvataan tämän arvojen keskiarvolla. Käytän tästä nimitystä *keskiarvomenetelmä*. Myös viitteessä [17] on käytetty tätä samaa menetelmää.

Jos lasketaan yhden sisätulokuvan yhdessä mielivaltaisessa lohkoissa syntyvä virhe menetelmissä 4:4:4 tai 4:2:2, voidaan todeta, että keskiarvomenetelmä vastaavassa lohkokokoissa ( $1 \times 1$  tai  $1 \times 2$ ) antaa yhtä suuren virheen (todistukset liitteessä 2).

Vertaillaan vielä 4:2:0-/4:1:1-menetelmää ja  $2 \times 2$ -/ $1 \times 4$ -lohkokokoissa laskettua keskiarvomenetelmää. Lasketaan tietokoneella, kuinka monessa eri yhdistelmässä mikäkin menetelmä antaa paremman tuloksen. Tässä tulee myös huomioida, että sijoitettava keskiarvo joudutaan pyöristämään kokonaisluvuksi sijoituksen yhteydessä. Käydään kaikki vaihtoehdot läpi (8-bittisillä kanava-arvoilla, eli arvoilla 0...255) ja ylläpidetään kolmea laskuria, jotka kasvavat sen mukaan kumpi järjestelmä oli parempi vai ovatko ne yhtä hyviä. Algoritmi 3.8 tekee tämän, ja sen tulokset ovat taulukossa 3.9. Kuvaaja tästä tuloksesta eriteltynä eri kulma-arvoilla on liitteessä 3.

### Algoritmi 3.8: $2 \times 2$ -/ $1 \times 4$ -keskiarvomenetelmä vs. $4:2:0/4:1:1$ -menetelmä

```
p = 0: q = 0: r = 0 (* laskurit *)
M = 255 (* 8-bit luvut 0..255 *)
For a = 0 To M Do (* Lohkon 1. arvo *)
For b = 0 To M Do (* Lohkon 2. arvo *)
For c = 0 To M Do (* Lohkon 3. arvo *)
For d = 0 To M Do (* Lohkon 4. arvo *)

(* Keskiarvomenetelmän virhe *)
k = Abs(a - Round((a + b + c + d) / 4)) + Abs(b - Round((a + b + c + d) / 4))
  + Abs(c - Round((a + b + c + d) / 4)) + Abs(d - Round((a + b + c + d) / 4))
(* 4:2:0/4:1:1-menetelmän virhe *)
v = Abs(b - a) + Abs(c - a) + Abs(d - a)
If k < v Then
  p = p + 1 (* p kasvaa, jos keskiarvomenetelmä parempi *)
Else
  (* q kasvaa, jos 4:2:0/4:1:1 parempi; r kasvaa, jos yhtä hyviä *)
  If k > v Then q = q + 1 Else r = r + 1 End IF
End If
End For
End For
End For
End For
Tulosta arvot p, r ja q (* keskiarvomenetelmä vs. 4:2:0/4:1:1 *)
```

Taulukko 3.9: Algoritmin 3.8 antama tulos

	Yhdistelmiä	Pros.
Keskiarvomenetelmä parempi (p)	2013304050	46,88 %
Molemmat yhtä hyviä (r)	1472200678	34,28 %
$4:2:0$ / $4:1:1$ -menetelmä parempi (q)	809462568	18,85 %

Taulukosta 3.9 nähdään, että  $2 \times 2$ -/ $1 \times 4$ -keskiarvomenetelmä on teoriassa parempi vaihtoehto spatiaaliselle pakkaukselle kuin  $4:2:0/4:1:1$ -menetelmä. Lähes puolet kaikista mahdollisista lohkon arvojen yhdistelmistä (46,88%) olivat sellaisia, missä keskiarvomenetelmä antaa spatiaalisessa pakkauksessa pienemmän virheen. Vastaavasti alle viidennes tapauksista (18,85%) olivat sellaisia, missä  $4:2:0/4:1:1$ -menetelmä antaa pienemmän virheen. Jäljelle jäävistä tapauksista (34,28%) kumpikin menetelmä antaa yhtä suuren virheen.

### 3.6 Mediaanimenetelmän aiheuttama virhe

Olen myös tutkinut matemaattisesti ja kokeellisesti menetelmää, missä lohko korvataan tämän arvojen mediaanilla. Käytän tästä nimitystä *mediaanimenetelmä*.

Olkoon menetelmässä  $V$  lohkon korvaava arvo  $u \in R$ . Seuraavat kohdat pätevät yleisesti (todistus liitteessä 2):

1. Ei ole olemassa arvoa  $u \in R$  (eikä siten menetelmää  $V$ ), joka lohkon korvatessaan aiheuttaisi lohkoissa aidosti pienemmän virheen kuin mediaanimenetelmä.
2. Mediaanimenetelmä antaa lohkoissa aidosti pienemmän virheen verrattuna menetelmään  $V$  jos ja vain jos  $u$  ei ole lohkon keskimäinen arvo lohkon alkioden määrän ollessa pariton, ja kun  $u$  ei ole lohkon keskimäisten alkioden suljetulla välillä lohkon alkioden määrän ollessa parillinen.
3. Muussa tapauksessa mediaanimenetelmä antaa lohkoissa yhtä suuren virheen kuin menetelmä  $V$ .

## 4. VIRHEMITTAUSMENETELMÄT

Virhemittausmenetelmiä on erilaisia. Seuraavassa esittelen yleisesti ne tavat mitata kuvien virheellisyyttä, joita olen itse tässä tutkielmassa käyttänyt.

### 4.1 Virhekaavat

Virhemittauksessa on käytetty useita kaavoja [11,13], jotka esittelen seuraavaksi. Kaavojen symbolien merkitykset ovat seuraavat:

$e_i$  on  $i$ . suurin ominaisarvo (ks. luku 3.2.1)

$n$  on kuvan leveys

$m$  on kuvan korkeus

$n \cdot m$  on pikselien lukumäärä,

$c$  on kanavien lukumäärä,

$s_{j,i}$  on alkuperäisen kuvan  $j$ :s kanava arvo  $i$ :s spektrissä

$\tilde{s}_{j,i}$  on rekonstruoidun kuvan  $j$ :s kanava arvo  $i$ :s spektrissä.

**Fidelity – Kuvan säilyvyys.** Lasketaan kuvan korrelaatiomatriisista (kaava 3.3) saatujen  $k$ :n ensimmäisen (kompressioon mukaan otettavien) ominaisarvojen (kaava 3.5) summa ja jaetaan se kaikkien ominaisarvojen summalla. Tämä mittaa kantavektoreissa säilyvän tiedon määrää (prosentteina) spektritason pakkauksen jälkeen.

$$Fidelity = \frac{\sum_{i=1}^k e_i}{\sum_{j=1}^c e_j} \cdot 100 \quad (4.1)$$

**MAX – Virheen huippuarvo.** Lasketaan alkuperäisestä kuvasta ja rekonstruoidusta kuvasta jokaisen spektrin kanava-arvojen erotusten summa  $\sum_{j=1}^c |s_{j,i} - \tilde{s}_{j,i}|$ , missä  $i=1,2,3,\dots, n \cdot m$ . Virheen huippuarvo on suurin summa eli virheellisin spektri

$$MAX = \max \sum_{j=1}^c |s_{j,i} - \tilde{s}_{j,i}|. \quad (4.2)$$

**MAE – Absoluuttinen keskivirhe** (*mean absolute error*). Lasketaan alkuperäisestä kuvasta ja rekonstruoidusta kuvasta kanava-arvojen erotusten keskiarvo eli kuvan keskimääräinen kanava-arvojen välinen virhe

$$MAE = \frac{1}{n \cdot m \cdot c} \sum_{j=1}^{n \cdot m} \sum_{i=1}^c |s_{i,j} - \tilde{s}_{i,j}|. \quad (4.3)$$

**MSE – Keskimääräinen neliövirhe** (*mean square error*). Lasketaan alkuperäisestä kuvasta ja rekonstruoidusta kuvasta kanava-arvojen erotusten neliön keskiarvo eli kuvan keskimääräinen kanava-arvojen välinen neliövirhe. Tämä korostaa virhettä enemmän kuin *MAE*.

$$MSE = \frac{1}{n \cdot m \cdot c} \sum_{i=1}^{n \cdot m} \sum_{j=1}^c (s_{j,i} - \tilde{s}_{j,i})^2 \quad (4.4)$$

**MSD – Keskimääräinen spektrietäisyys** (*mean spectral distance*). Lasketaan jokaiselle spektrille alkuperäisestä kuvasta ja rekonstruoidusta kuvasta kanava-arvojen neliön summa ja otetaan tästä neliöjuuri. Näistä tuloksista otetaan keskiarvo. Tämä mittaa spektreihin aiheutuva virhettä.

$$MSD = \frac{1}{n \cdot m} \sum_{i=1}^{n \cdot m} \sqrt{\sum_{j=1}^c (s_{j,i} - \tilde{s}_{j,i})^2} \quad (4.5)$$

**SNR – Signaalikohinasuhde** (*signal-to-noise ratio*). Lasketaan keskimääräinen alkuperäisen kuvan kanava-arvon neliö, jaetaan se kuvan keskimääräisellä neliövirheellä ja kerrotaan tämän 10-logaritmi arvolla 10. Tämä mittaa kompressiossa syntyvää kohinaa.

$$SNR = 10 \cdot \log_{10} \frac{\frac{1}{m \cdot n \cdot c} \sum_{i=1}^{m \cdot n} \sum_{j=1}^c s_{j,i}^2}{MSE}, \quad (4.6)$$

missä *MSE* on keskimääräinen neliövirhe.

**PSNR – Signaalikohinasuhteen huippuarvo** (*peak signal-to-noise ratio*). Kuten *SNR*, mutta keskimääräisen alkuperäisen kuvan kanava-arvon sijasta käytetään kanava-arvon teoreettista maksimia.

$$PSNR = 10 \cdot \log_{10} \frac{M^2}{MSE}, \quad (4.7)$$

missä *M* on kanava-arvon teoreettinen maksimi ja *MSE* on keskimääräinen neliövirhe.

**$\Delta E$  – Ihmissilmän näköön perustuva väri virhe.** Muunnetaan alkuperäinen ja rekonstruoitu kuva CIELAB 1976  $L^*a^*b^*$ -kuviksi [6,27], minkä jälkeen virhelaskenta toteutetaan laske-  
malla näiden kuvien komponenttien erotusten neliön summan neliöjuuri. Yleisesti ihminen ei  
voi erottaa virhettä, missä  $\Delta E < 0,5$ . Yhdessä kuvapisteessä tapahtuva väri virhe lasketaan  
seuraavasti:

$$\Delta E = \sqrt{\Delta L^{*2} + \Delta a^{*2} + \Delta b^{*2}} . \quad (4.8)$$

#### 4.2 Väri virheet ja S-CIELAB

Kaikkien kuvapisteiden  $\Delta E$ -virheen keskiarvoa ei sellaisenaan olla pidetty värispektrikuvalle  
oikeellisenä, sillä ihminen ei erota värieroja sinisillä ja punaisilla alueilla niin selvästi kuin  
vihreillä ja keltaisilla alueilla [28]. Tämä johtuu siitä, että sinisen ja punaisen värin aallonpi-  
tuudet sijaitsevat ihmisen näköalueen (400...700 nm) reunoilla, kun taas vihreä ja keltainen  
ovat keskialueilla. Tähän ongelmaan onkin kehitetty menetelmä, missä spektrikuvasta lasket-  
tujen luminanssikuvan ja kahden opponenttimuotoisen krominanssikuvan krominanssikanavia  
suodatetaan spatiaalisesti siten, että sinisten ja punaisten aallonpituusalueiden virheet jäävät  
 $\Delta E$ :n laskemisessa vähemmälle huomiolle. Suodatukseen huomioidaan hyvin käytännöllisiä  
seikkoja, sillä siihen vaikuttavat mm. kuvan katselijan ja tietokoneen monitorin välinen etäi-  
syys sekä kuvaruudun fyysinen koko. Tämän jälkeen lasketaan CIELAB-esitys sekä normaali  
 $\Delta E$ -virhe. Menetelmän nimi S-CIELAB tulee sanoista *Spatial CIELAB*, sillä se on spatiaa-  
linen laajennus CIELAB-esitykseen. Lisätietoa tästä suodatuksesta löytyy viitteestä [28]. Itse  
olen tutkimuksessani käyttänyt suodatuksen tekijöiden Matlab-koodeja [29].

## 5. TESTIAINEISTOT

Tutkimuksessani oli käytössä alun perin yksi kuvasarja (Bristol), joka kuitenkin osoittautui hyvin piikikkääksi dataksi, ja oletin tietokannassa olevan kameran aiheuttamia kuvausvirheitä. Tämän jälkeen keskityin kahteen muuhun kuvasarjaan (Forest ja Coral), mitkä osoittautuivat erinomaisiksi – piikkejä ei juuri ollut. Tässä luvussa esittelen nämä kuvasarjat ja niiden ominaisuudet. Kuvat näistä kuvasarjoista löytyvät liitteistä 4a-4c.

### 5.1 Bristol-kuvasarja

Bristol-kuvasarjan kuvat on kuvattu värispektrikameralla, jossa käytetyt 31 värisuodatinta ulottuvat 10 nm välein 400...700 nm valon aallonpituusalueelle. Siten kamera tuottaa eri aallonpituuksilta 31 kpl kuvia, mitkä yhdessä muodostavat yhden värispektrikuvan. Kuvat ovat 256×256-kokoisia, ja sen kanava-arvot ovat välillä 0..255 (8-bittisiä). Kuvausjärjestelmä on selitetty tarkemmin viitteessä [24]. Kuvasarjassa on kuvia 29 kpl, mutta käyttämästäni lähteestä [5] puuttui yksi kuva nimeltään '*fuschia*'. Yhden tällaisen kuvan koko on 256·256·31·1 tavua = 2031616 tavua = n. 1,94 megatavua, joten koko kuvasarjan koko on n. 56,2 megatavua. Ks. liite 4a.

### 5.2 Forest- ja Coral-kuvasarjat

Näiden kuvasarjojen kuvat on kuvattu värispektrikameralla, jossa käytetyt 40 värisuodatinta ulottuvat n. 400...700 nm valon aallonpituusalueelle. Siten kamera tuottaa eri aallonpituuksilta 40 kpl kuvia, mitkä yhdessä muodostavat yhden värispektrikuvan. Kuvat ovat 128×128-kokoisia, ja sen kanava-arvot ovat välillä 0..255 (8-bittisiä). Kuvausjärjestelmä on selitetty tarkemmin viitteessä [4]. Forest-sarjassa on 12 kuvaa ja Coral-sarjassa 10 kuvaa. Yhden tällaisen kuvan koko on 128·128·40·1 tavua = 655360 tavua = 640 kilotavua, joten kuvasarjojen yhteiskoko on n. 13,8 megatavua. Ks. liitteet 4b-4c.

Forest-kuvasarjan aallonpituudet (nm) eri värisuotimissa:

403	410	416	423	431	438	445	452
459	466	473	481	488	496	503	511
518	525	533	541	549	557	565	572
580	588	596	604	612	620	627	635
642	650	658	666	674	682	689	696



Coral-kuvasarjan aallonpituudet (nm) eri värisuotimissa:

405	411	418	425	432	439	446	452
459	466	473	480	488	495	502	509
517	525	532	540	548	556	564	571
579	586	594	603	611	619	626	634
641	649	657	665	673	680	688	696

## 6. VIRHELASKENNAT

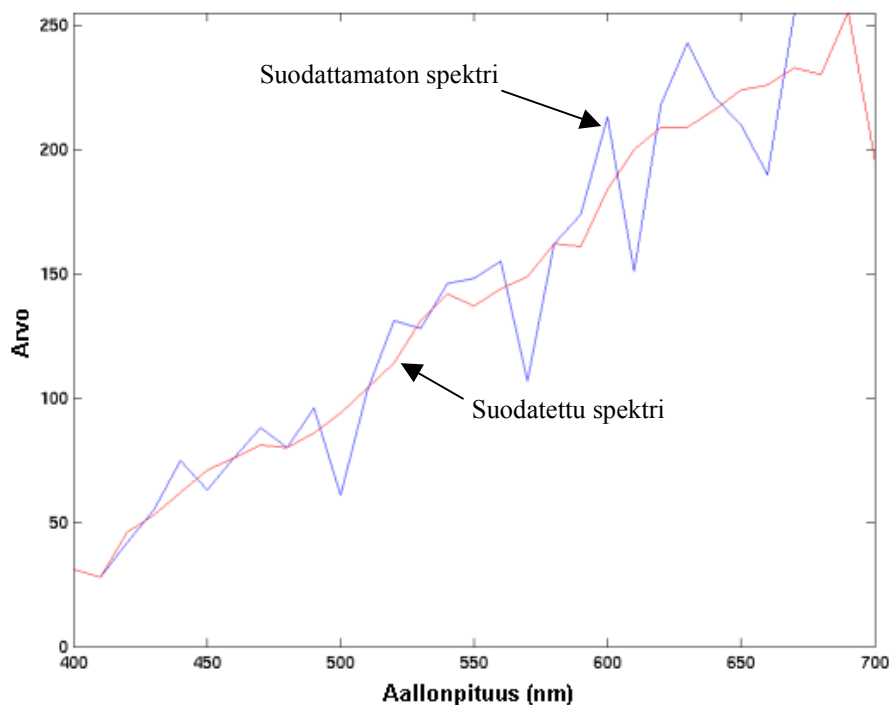
Tässä luvussa selvitän ja vertailen kokeellisesti testiaineistoni avulla eri kompressiomenetelmien aiheuttamia virheitä. Selvitän yksityiskohtaisemmin Forest-kuvasarjan virheitä ja vertaan niitä yleisesti Coral-kuvasarjaan. Lopuksi esittelen vielä yleisesti Bristol-kuvasarjan virheitä.

### 6.1 Spektrien suodatus

Spektrien suodatuksessa olen käyttänyt arvoa  $x=2$ , jolloin  $i$ :s spektrin  $j$ :s kanava-arvon (missä  $j = 3, 4, \dots, c-2$ ) korvaava arvo lasketaan kaavalla 3.2, eli seuraavasti:

$$s_{j,i} = \frac{\sum_{y=j-2}^{j+2} s_{y,i}}{2 \cdot 2 + 1}.$$

Kuvasta 6.1 nähdään, että suodatus tasoittaa spektrissä esiintyviä äkillisiä muutoksia ja piikkejä. Näin päästään lähemmäksi todellista kuvaa, sillä kuvauksen yhteydessä tulleet virheet jäävät vähemmälle huomiolle. Kuvan 6.1 spektri on Bristol-kuvasarjasta, jossa on erittäin piikikkäitä spektrejä. Forest- ja Coral-kuvasarjat ovat huomattavasti virheettömämpiä.



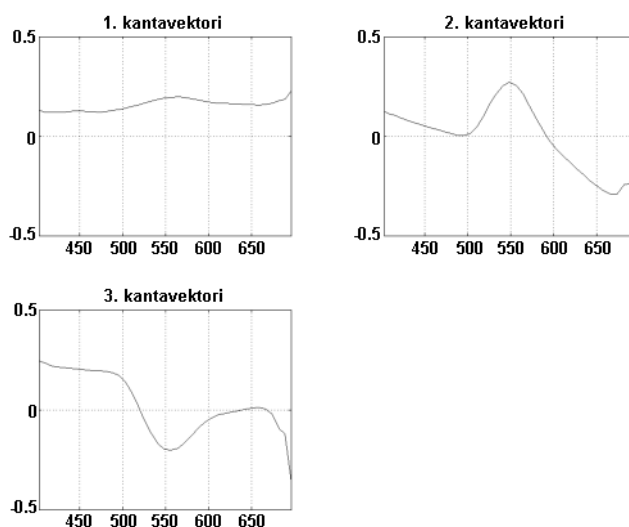
Kuva 6.1: Spektrin suodatus.

## 6.2 Spektritason pakkaus

Pääkomponenttianalyysillä (PCA) tapahtuvassa spektritason pakkauksessa kantavektoreissa säilyvän tiedon määrää (eli informaatioisisältöä) kuvaa mitta *Fidelity*. Tämä voidaan laskea jo ennen varsinaista spektritason pakkausta, sillä laskentaan huomioidaan vain pääkomponenttianalyysillä saadut ominaisarvot. Menetelmällä 4:4:4 (ks. selitys luvusta 3.3, tulokset ovat luvussa 6.3) saadut tulokset ovat vain spektritason pakkauksen aiheuttamia virheitä, koska tällä menetelmällä ei spatiaalista pakkausta tapahdu. Taulukossa 6.2 on esimerkkinä Forest-kuvasarjan *Bank*-niminen kuva (ks. kuva 6.9), jonka *Fidelity*-arvot on laskettu, kun kompressiossa käytetään 1...10 kantavektoria. Laskettaessa kanta kolmella kantavektorilla, saadaan kuvassa 6.3 olevat vektorit. Kuvan 6.3 kantavektoreiden ja alkuperäisen spektrikuvan avulla lasketut sisätulokuvat ovat kuvassa 6.4.

Taulukko 6.2: *Bank*-kuvan kantavektoreissa säilyvä tiedon määrä.

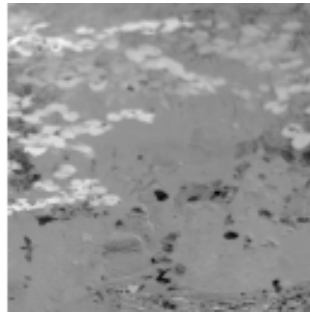
Kantavektoreiden määrä	1	2	3	4	5	6	7	8	9	10
<b>Fidelity</b>	96,94	98,83	99,82	99,92	99,95	99,97	99,98	99,98	99,99	99,99



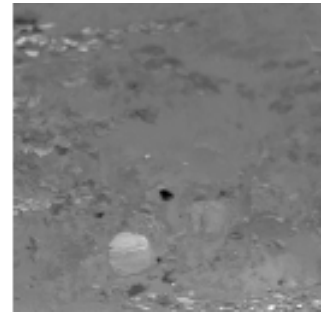
Kuva 6.3: *Bank*-kuvan kantavektorit ( $k=3$ , *Fidelity* = 99,82).



1. sisätulokuva  
(luminanssi)



2. sisätulokuva  
(krominanssi)



3. sisätulokuva  
(krominanssi)

Kuva 6.4: Kantavektoreista muodostetut sisätulokuvat.

*Fidelity*-arvot koko Forest-kuvasarjalle ovat taulukossa 6.5 ja Coral-kuvasarjalle taulukossa 6.6. Coral-kuvasarjassa ei päästä niin suureen tiedon säilyvyyteen kuin Forest-kuvasarjassa, sillä Coral-sarjan kuvissa on enemmän erilaisia värejä.

Taulukko 6.5: Forest-kuvasarjan kantavektoreissa keskimäärin säilyvän tiedon määrä.

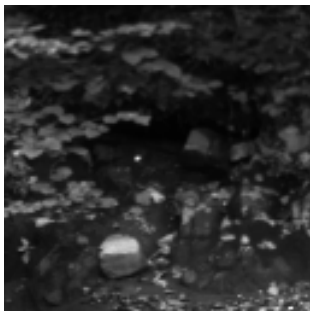
Kantavektoreiden määrä	1	2	3	4	5	6	7	8	9	10
<b>Fidelity, keskiarvo</b>	98,14	99,34	99,76	99,89	99,94	99,96	99,97	99,98	99,98	99,98
<b>Fidelity, mediaani</b>	98,25	99,40	99,80	99,91	99,94	99,96	99,97	99,98	99,98	99,98

Taulukko 6.6: Coral-kuvasarjan kantavektoreissa keskimäärin säilyvän tiedon määrä.

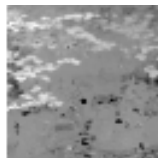
Kantavektoreiden määrä	1	2	3	4	5	6	7	8	9	10
<b>Fidelity, keskiarvo</b>	97,17	99,12	99,59	99,75	99,84	99,88	99,91	99,93	99,95	99,96
<b>Fidelity, mediaani</b>	97,53	99,34	99,65	99,80	99,87	99,91	99,93	99,94	99,96	99,97

### 6.3 Spatiaalitason pakkaus

Olkoon värispektrikuva spektritasossa pakattu pääkomponenttianalyysillä (PCA) 3 kantavektoria käyttäen. Kiinnitetään aluksi huomiota tavallisiin menetelmiin, missä lohkon kulma-arvo (3×3-keskipistemethodessä lohkon keskimääräinen arvo) korvaa lohkon. Kompressoidaan spatiaalisessa tasossa 4:2:0-menetelmällä Forest-kuvasarjan *Bank*-kuva (ks. kuva 6.7).



1. sisätulokuva  
(luminanssi)



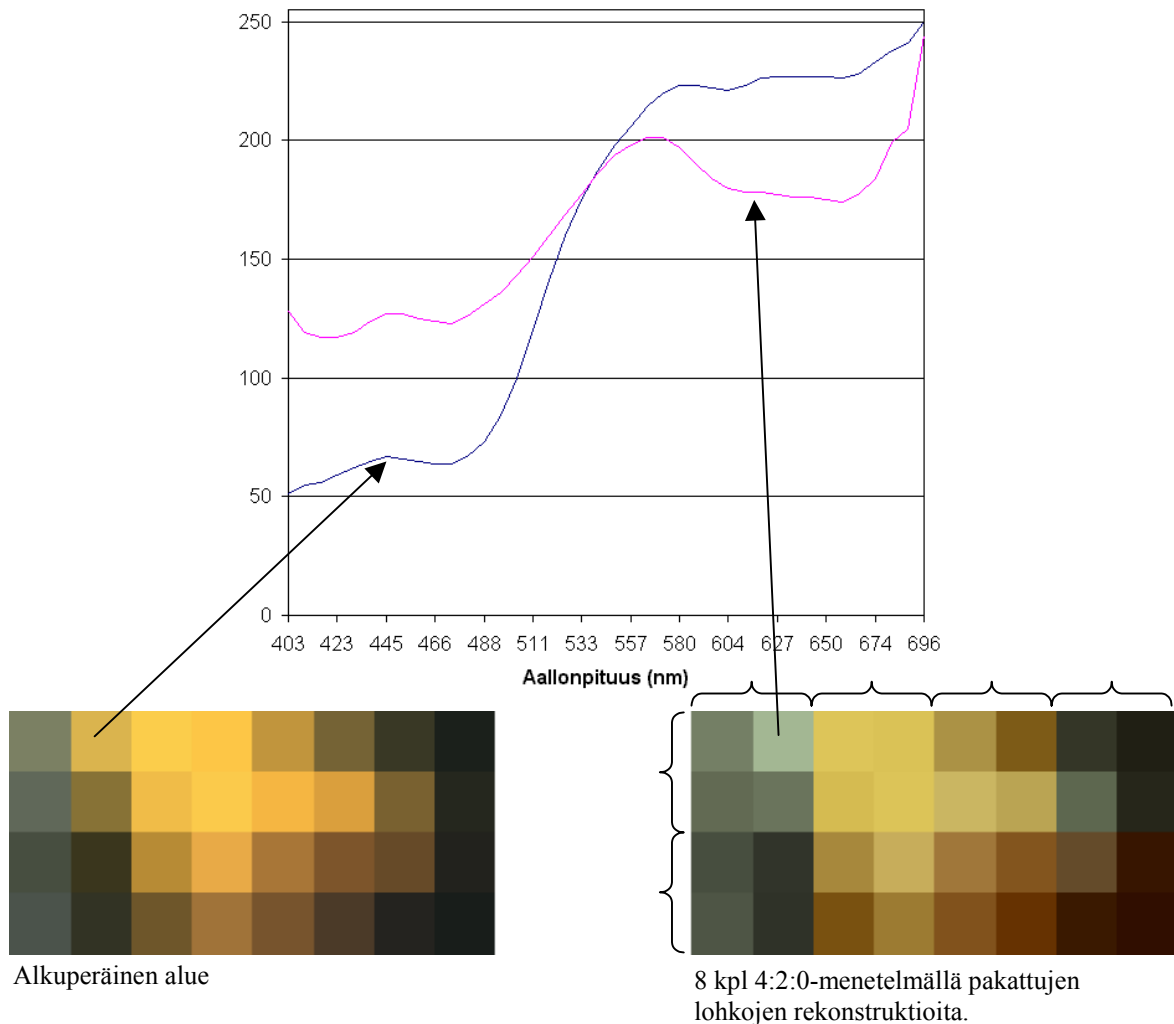
4:2:0-menetelmällä  
pakattu 2. sisätulokuva  
(krominanssi)



4:2:0-menetelmällä  
pakattu 3. sisätulokuva  
(krominanssi)

Kuva 6.7 Spatiaalisesti pakatut sisätulokuvat.

Purkuvaiheessa spatiaalisesti pakatut sisätulokuvat palautetaan luvussa 3.4 esitetyllä tavalla. Näistä palautetuista sisätulokuvista spektrikuva rekonstruoidaan kaavalla 3.10, jolloin alkuperäistä ja rekonstruoitua spektrikuvaa voidaan verrata. Tästä huomataan, että rekonstruktiossa kuva on hieman muuttunut. Esimerkkinä tästä seuraavat lohkot kuvassa 6.8. Kuvan lohkot esitetään spektrikuvasta laskettuina RGB-kuvina.



*Kuva 6.8: Lohkojen pakkaus 4:2:0-menetelmällä ja vertailu.*

Kuvassa 6.8 on lohkon sisällä on hyvin erilaisia värejä. Näin pienten lohkojen arvot ovat kuitenkin usein lähellä toisiaan, jolloin kuvan 6.8 kaltainen virhe ei ole niin suuri. Kuvasta kuitenkin näkee selvästi spatiaalisen pakkauksen vaikutuksen. Käytän jatkossa tätä samaa lohkoa havainnollistaakseni eri menetelmiä. Käsittelemäni lohko on otettu kuvan keskellä olevasta pienestä keltaisesta alueesta (kuva 6.9).

Kaikkiaan alkuperäinen kuva ja rekonstruoitu kuva näyttävät lähes samalta (kuva 6.9). Kuvassa 6.9 oleva erotuskuva on laskettu alkuperäisen kuvan ja rekonstruoidun kuvan erotuksen

itseisarvolla, minkä lisäksi jokainen kanava-arvo on kerrottu arvolla 16, jotta värivirheiden sijainnit korostuisivat. Erotuskuvassa tietty väri tarkoittaa suurempaa virhettä kyseisen värin spektrialueella. Mitä vaaleampi on pikseli, sitä suurempi on virhe. Koska kuvaa on jälkeensä muokattu, se ei yksittäisessä kuvassa näytä oikein virheen suuruutta, mutta sitä voidaan suoraan verrata muihin vastaavalla tavalla muokattuihin erotuskuviin (kuvat 6.17 ja 6.21). Taulukossa 6.10 ovat tämän kuvakompression virheet.



*Kuva 6.9: Alkuperäinen kuva, 4:2:0-menetelmällä pakatun kuvan rekonstruktio sekä näiden erotuskuva.*

*Taulukko 6.10: 4:2:0-menetelmällä pakatun Bank-kuvan virheet.*

	<b>Bank 4:2:0-menetelmä</b>
<b>MAX</b>	1789
<b>MAE</b>	2,91
<b>MSE</b>	27,53
<b>MSD</b>	22,77
<b>SNR</b>	20,13
<b>PSNR</b>	33,73
<b><math>\Delta E</math> keskiarvo CIELAB</b>	4,00
<b><math>\Delta E</math> keskiarvo S-CIELAB</b>	1,72
<b><math>\Delta E</math> mediaani CIELAB</b>	2,64
<b><math>\Delta E</math> mediaani S-CIELAB</b>	1,40

Taulukossa 6.11 esittelen eri spatiaalisilla pakkausmenetelmillä aiheutuneiden virheiden keskiarvoja koko kuvasarjalle. Spektritason pakkaukseen käytetään kuitenkin edelleen PCA-menetelmää ja kolmea sisätulokuvaa. Taulukosta 6.10 nähdään, että valitsemani esimerkkikuva *bank* on muita kuvia huomattavasti virheellisempi.

Taulukko 6.11: Keskimääräiset virheet Forest-kuvasarjan 12 kuvalle.

Forest, PCA 3 sisätulokuvaa	MAX	MAE	MSE	MSD	SNR	PSNR
4:4:4	661	1,45	6,47	12,44	26,75	41,39
4:2:2	1225	2,13	14,61	17,28	23,30	37,94
4:2:0	1337	2,76	23,76	21,76	21,24	35,87
4:1:1	1389	3,00	28,56	23,57	20,33	34,97
3×3 keskipiste	1286	3,05	28,10	23,90	20,54	35,17

Seuraavaksi esitän taulukossa 6.12  $\Delta E$ -virheet suodatetulle Forest-kuvasarjalle. Vertailuna olen laskenut tavalliselle CIELAB-kuville lasketun virheen ja suodatetuille CIELAB-kuville (S-CIELAB) lasketun virheen keskiarvot ja mediaanit. S-CIELAB:in laskemisessa on ollut oletuksena, että kuvaa katsotaan 72 dpi –monitorilla 45 cm päästä.

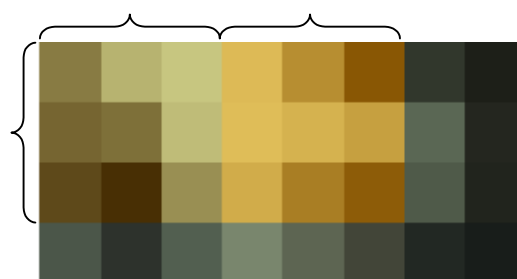
Taulukko 6.12: Keskimääräiset  $\Delta E$ -virheet Forest-kuvasarjalle.

Forest, PCA 3 sisätulokuvaa	$\Delta E$ keskiarvo CIELAB	$\Delta E$ keskiarvo S-CIELAB	$\Delta E$ mediaani CIELAB	$\Delta E$ mediaani S-CIELAB
4:4:4	1,53	1,01	1,28	0,82
4:2:2	2,97	1,31	1,98	1,11
4:2:0	4,23	1,71	3,01	1,44
4:1:1	4,73	2,15	3,26	1,76
3×3 keskipiste	4,82	2,02	3,61	1,69

Kompressiota lisättäessä virhe kasvaa. Mielenkiintoista on, että 4:1:1-menetelmällä ja 3×3-keskipistemenetelmällä saadaan lähes sama virhe, vaikka jälkimmäisessä pakataankin krominanssikanavia vastaavia sisätulokuvia yli 2-kertaisesti. Tämä johtuu siitä, että 4:1:1-menetelmän pikselit ovat spatiaalisesti etäämmällä toisistaan kuin 3×3-keskipistemenetelmässä. Tällöin 4:1:1-menetelmän etäisin pikseli poikkeaa korvaavasta pikselistä jo niin paljon, että virhe kasvaa huomattavasti (kuva 6.13).



8 kpl 4:1:1-menetelmällä pakattujen lohkojen rekonstruktioita. Ensimmäinen pikseli vie muualta lohkoista runsaasti värejä.



2 kpl 3×3-keskipistemenetelmällä pakattujen lohkojen rekonstruktioita. Seuraavat lohkot ottavat korvaavan arvon jo omasta lohkoistaan.

Kuva 6.13: 4:1:1-menetelmä vs. 3×3-keskipistemenetelmä.

Taulukoissa 6.13 ja 6.14 ovat vastaavat virhelaskennat Coral-kuvasarjan 10 kuvalle.

*Taulukko 6.14: Keskimääräiset virheet Coral-kuvasarjalle.*

Coral, PCA 3 sisätulokuvaa	MAX	MAE	MSE	MSD	SNR	PSNR
4:4:4	794	2,29	12,13	18,81	24,33	37,72
4:2:2	1327	2,78	18,95	22,49	22,19	35,58
4:2:0	1526	3,19	26,09	25,59	20,79	34,18
4:1:1	1902	3,40	31,46	27,16	20,05	33,44
3×3 keskipiste	1581	3,42	30,24	27,42	20,14	33,53

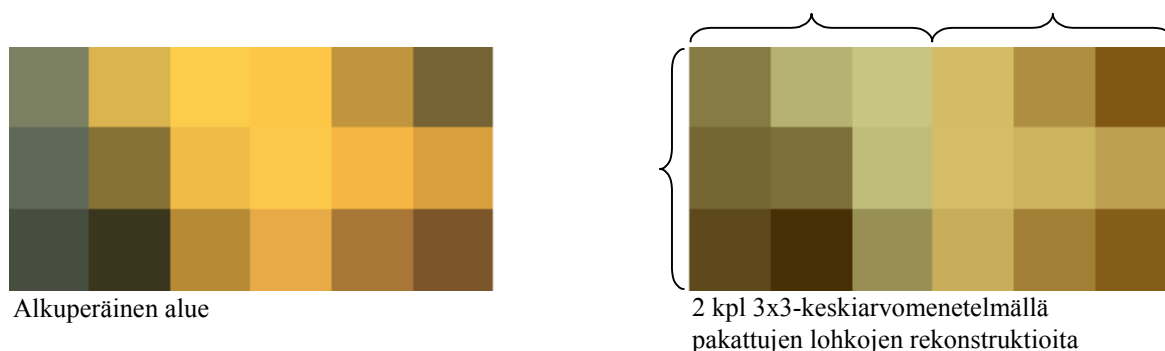
*Taulukko 6.15: Keskimääräiset  $\Delta E$ -virheet Coral-kuvasarjalle.*

Coral, PCA 3 sisätulokuvaa	$\Delta E$ keskiarvo CIELAB	$\Delta E$ keskiarvo S-CIELAB	$\Delta E$ mediaani CIELAB	$\Delta E$ mediaani S-CIELAB
4:4:4	3,00	2,15	2,47	1,75
4:2:2	3,79	2,29	3,06	1,90
4:2:0	4,38	2,50	3,60	2,11
4:1:1	4,63	2,69	3,72	2,26
3×3 keskipiste	4,73	2,66	3,91	2,23

Coral-kuvasarjan virheet ovat huomattavasti suurempia kuin Forest-kuvasarjan virheet. Tämä johtuu siitä, että Coral-kuvasarjassa on enemmän erilaisia värejä, jolloin spatiaalisessa pakkauksessa väritieto muuttuu lohkoissa huomattavasti.

#### 6.4 Keskiarvomenetelmä

Kompressoidaan Bank-kuva (spektritasossa pakattu PCA-menetelmällä ja kolmella kantavektorilla) 3×3-kokoisille lohkoille keskiarvomenetelmällä. Kuvassa 6.16 on tarkasteltu samoja lohkoja kuin kuvassa 6.13, mutta alin rivi ja kaksi oikeanpuoleisinta saraketta on poistettu, sillä ne ovat jo osia seuraavista lohkoista. Kuva luonnollisesti tasoittuu keskiarvomenetelmällä hieman enemmän, sillä korvaavaan lohkoon huomioidaan nyt kaikki lohkon värit.



*Kuva 6.16: Keskiarvomenetelmällä kompressoituja 3×3-lohkoja.*



Kuvassa 6.17 erotuskuva näyttää tasaisemmalta, mikä viittaa absoluuttisen virheen tasaantumiseen spatiaalitasossa. Kuvan keskellä oleva keltainen alue on kuitenkin silmännähtävästi tummempi. Bank-kuvan virheet 3×3-keskiarvomenetelmällä pakattuna ovat taulukossa 6.18.



Kuva 6.17: Alkuperäinen kuva, 3×3-keskiarvomenetelmällä pakatun kuvan rekonstruktio sekä näiden erotuskuva.

Taulukko 6.18: 3×3-keskiarvomenetelmällä kompressoitun Bank-kuvan virheet.

	<b>Bank 3×3-keskiarvo</b>
<b>MAX</b>	1422
<b>MAE</b>	2,99
<b>MSE</b>	25,11
<b>MSD</b>	23,36
<b>SNR</b>	20,53
<b>PSNR</b>	34,13
<b><math>\Delta E</math> keskiarvo CIELAB</b>	4,18
<b><math>\Delta E</math> keskiarvo S-CIELAB</b>	1,59
<b><math>\Delta E</math> mediaani CIELAB</b>	3,23
<b><math>\Delta E</math> mediaani S-CIELAB</b>	1,29

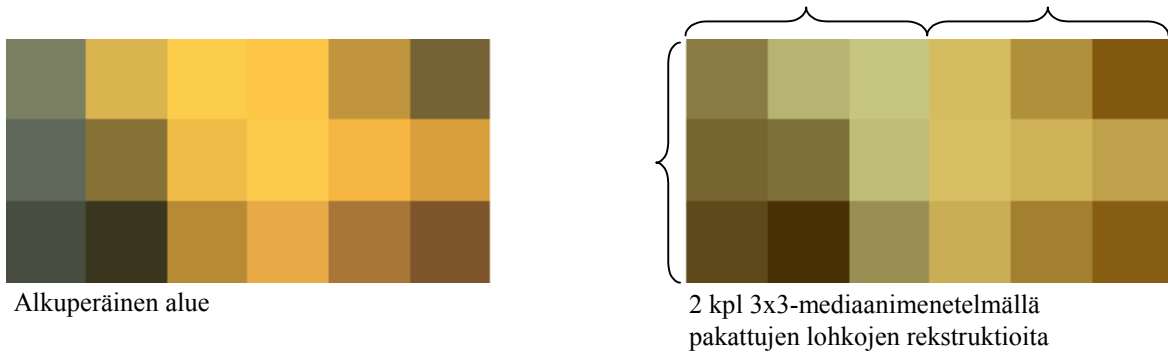
Taulukossa 6.19 ovat keskimääräiset virheet koko Forest-kuvasarjalle ja lisäksi vertailun vuoksi 3×3-kulmapiste- ja 3×3-keskipistemenetelmien aiheuttamat virheet. Keskiarvomene- telmä pienentää virhettä huomattavasti. Lisäksi olen laskenut keskimääräiset virheet myös 3×3-keskipistemenetelmille, missä sisätulokuvien määrä on 4 ja 5. Keskiarvomenetelmä antaa 3 sisätulokuvalla edelleen parempia tuloksia kuin keskipistemenetelmä 5 sisätulokuvalla.

Taulukko 6.19: Keskimääräiset eri 3×3-menetelmien aiheuttamat virheet Forest-kuvasarjalle.

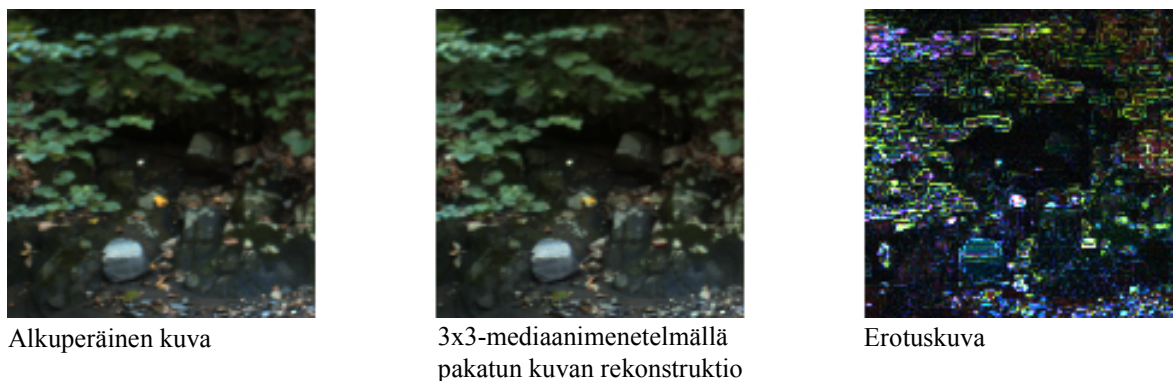
Forest, PCA 3 sisätulokuvaa	MAX	MAE	MSE	MSD	SNR	PSNR	$\Delta E$ keskiarvo CIELAB	$\Delta E$ keskiarvo S-CIELAB	$\Delta E$ mediaani CIELAB	$\Delta E$ mediaani S-CIELAB
3×3 keskiarvo	1091	2,68	19,95	21,18	21,97	36,60	4,14	1,47	3,33	1,25
3×3 kulmapiste	1381	3,46	36,03	26,80	19,45	34,08	5,62	2,50	4,13	2,04
3×3 keskipiste	1286	3,05	28,10	23,90	20,54	35,17	4,82	2,02	3,61	1,69
3×3 keskipiste, 4 sisätulokuvaa	1294	2,94	27,51	22,97	20,65	35,28	4,67	1,81	3,42	1,49
3×3 keskipiste, 5 sisätulokuvaa	1344	2,90	27,53	22,64	20,66	35,30	4,61	1,73	3,36	1,41

## 6.5 Mediaanimenetelmä

Mediaanimenetelmässä kompressio toteutetaan samalla tavalla kuin keskiarvomenetelmässäkin, mutta nyt keskiarvon sijaan käytetään mediaania. Kuvasta 6.20 huomataan, että lohossa ei juuri ole eroa keskiarvomenetelmään verrattuna (vrt. kuva 6.16). Kuvassa 6.21 on alkuperäinen kuva, mediaanimenetelmällä pakatun kuvan rekonstruktio sekä näiden erotuskuva (vrt. kuva 6.17).



Kuva 6.20: Mediaanimenetelmällä kompressoitu 3×3-lohko.



Kuva 6.21: Alkuperäisen Bank-kuvan ja 3×3-mediaanimenetelmällä pakatun kuvan rekonstruktio sekä näiden erotuskuva.

Taulukoista 6.22 ja 6.23 havaitaan, että Bank-kuvalle ja koko kuvasarjalle lasketut virheet eivät juuri poikkea keskiarvomenetelmästä (vrt. taulukot 6.18 ja 6.19).

*Taulukko 6.22: 3×3-mediaanimenetelmällä kompressoidun Bank-kuvan virheet.*

	<b>Bank 3×3-mediaani</b>
<b>MAX</b>	1509
<b>MAE</b>	2,95
<b>MSE</b>	27,42
<b>MSD</b>	23,10
<b>SNR</b>	20,15
<b>PSNR</b>	33,75
<b>ΔE keskiarvo CIELAB</b>	4,03
<b>ΔE keskiarvo S-CIELAB</b>	1,63
<b>ΔE mediaani CIELAB</b>	2,96
<b>ΔE mediaani S-CIELAB</b>	1,28

*Taulukko 6.23: Keskimääräiset 3×3-mediaanimenetelmän aiheuttamat virheet Forest-kuvasarjalle.*

	<b>Forest-kuvasarja 3×3-mediaani</b>
<b>MAX</b>	1190
<b>MAE</b>	2,67
<b>MSE</b>	21,31
<b>MSD</b>	21,14
<b>SNR</b>	21,70
<b>PSNR</b>	36,34
<b>ΔE keskiarvo CIELAB</b>	4,06
<b>ΔE keskiarvo S-CIELAB</b>	1,58
<b>ΔE mediaani CIELAB</b>	3,11
<b>ΔE mediaani S-CIELAB</b>	1,32

## 6.6 Keskiarvomenetelmä vs. mediaanimenetelmä

Kuvassa 6.24 havainnoidaan Bank-kuvan lohkoa, jossa keskiarvomenetelmä eroaa mediaanimenetelmästä.



Kuva 6.24: Keskiarvomenetelmä vs. mediaanimenetelmä.

Kuvan 6.24 alueella keskiosan ruskeat lohkot ovat muuttuneet mediaanimenetelmässä huomattavasti vihreämmäksi kuin keskiarvomenetelmässä. Ruskea alue on neljän lohkon ympäröimänä juuri siten, että lohkoissa on vihreitä pikseleitä enemmän kuin ruskeita. Näin ollen lohkon mediaanikin on lähempänä vihreää. Keskiarvomenetelmässä pikselit ovat taas haaleampia, sillä siinä ruskeat ja vihreät alueet yhdessä muodostavat uuden lohkon.

Todistin liitteessä 2, että mediaanimenetelmä antaa spatiaalisessa tasossa aina pienimmän virheen. Kuitenkin näitä arvoja vertailemalla (taulukot 6.25 ja 6.26) löydetään mediaanimenetelmässä isompia virheitä kuin keskiarvomenetelmässä. Oletan tämän johtuvan siitä, että mediaanimenetelmässä virhe suurenee enemmän rekonstruoitaessa kuva takaisin sisältölu-  
vistä spektrikuvaksi kuin keskiarvomenetelmässä.

Taulukko 6.25: Keskiarvo- ja mediaanimenetelmän vertailu.

Virheet ovat Forest-kuvasarjan keskiarvoja.

	Forest-kuvasarja 3x3-keskiarvo	Forest-kuvasarja 3x3-mediaani
<b>MAX</b>	1091	1190
<b>MAE</b>	2,68	2,67
<b>MSE</b>	19,95	21,31
<b>MSD</b>	21,18	21,14
<b>SNR</b>	21,97	21,70
<b>PSNR</b>	36,6	36,34
<b><math>\Delta E</math> keskiarvo CIELAB</b>	4,14	4,06
<b><math>\Delta E</math> keskiarvo S-CIELAB</b>	1,47	1,58
<b><math>\Delta E</math> mediaani CIELAB</b>	3,33	3,11
<b><math>\Delta E</math> mediaani S-CIELAB</b>	1,25	1,32

Taulukko 6.26: Keskiarvo- ja mediaanimenetelmän vertailu.

Virheet Coral-kuvasarjan keskiarvoja.

	Coral-kuvasarja 3x3-keskiarvo	Coral-kuvasarja 3x3-mediaani
<b>MAX</b>	1201	1385
<b>MAE</b>	3,09	3,10
<b>MSE</b>	23,30	24,39
<b>MSD</b>	24,96	25,01
<b>SNR</b>	21,33	21,14
<b>PSNR</b>	34,72	34,53
<b><math>\Delta E</math> keskiarvo CIELAB</b>	4,20	4,19
<b><math>\Delta E</math> keskiarvo S-CIELAB</b>	2,33	2,36
<b><math>\Delta E</math> mediaani CIELAB</b>	3,56	3,50
<b><math>\Delta E</math> mediaani S-CIELAB</b>	1,93	1,98

Näin pienellä lohkokokoolla on vaikea järjestää keskiarvo- ja mediaanimenetelmä paremmuusjärjestykseen. Keskiarvomenetelmän virhe riippuu pitkälti siitä, minkälaisia jakaumia lohkon eri arvot noudattavat luonnonkuvissa. Keskinäinen paremmuus voisi olla mahdollista laskea näiden todennäköisyysjakaumien avulla [2].

### 6.7 Bristol-kuvasarjan virheitä

Aikaisemmin mainitsin käyttäneeni myös Bristol-kuvasarjaa (28 kuvaa), johon en kuitenkaan datan piikkisyyden vuoksi keskittynyt. Taulukossa 6.27 on Bristol-kuvasarjan keskimääräisiä virheitä.

Taulukko 6.27: Bristol-kuvasarjan virheiden keskiarvoja.

	MAX	MAE	MSE	MSD	SNR	PSNR	$\Delta E$ keskiarvo CIELAB
<b>4:4:4</b>	888	3,38	37,58	26,19	22,26	33,25	3,59
<b>4:2:2</b>	1151	3,96	47,89	29,56	21,05	32,04	5,12
<b>4:2:0</b>	1647	4,58	62,40	33,29	19,86	30,86	6,53
<b>4:1:1</b>	2012	5,21	82,59	37,21	18,53	29,52	7,87

## 7. KOMPRESSIOSUHDE JA SIIRTO TIETOVERKOISSA

Kompressiosuhde eri menetelmille lasketaan seuraavalla kaavalla [11]:

$$CR = \frac{\text{pakkaamattoman kuvan koko}}{\text{pakatun kuvan koko}} \quad (7.1)$$

Bristol-kuvien kompressiosuhteet on kuvattu taulukossa 7.1.

*Taulukko 7.1: Eri menetelmien kompressiosuhteita Bristol-kuville.*

Menetelmä	Pakkaamattoman kuvan koko (tavua)	Kompressoitun kuvan koko (tavua)	CR
4:4:4 (3 sisätulokuvaa)	2031616	196701	10,3
4:2:2 (3 sisätulokuvaa)	2031616	131165	15,5
4:2:0 (3 sisätulokuvaa)	2031616	98397	20,6
4:1:1 (3 sisätulokuvaa)	2031616	98397	20,6
3×3 (3 sisätulokuvaa)	2031616	80421	25,3
3×3 (4 sisätulokuvaa)	2031616	87848	23,1
3×3 (5 sisätulokuvaa)	2031616	95275	21,3

Kuvan tiedonsiirtoon kuluva aika sekunteina lasketaan kaavalla:

$$Aika = \frac{\text{Kuvan koko} \cdot 8 \text{ bittiä / tavu}}{\text{Tiedonsiirtonopeus}} \quad (7.2)$$

Tällöin teoreettisiksi siirtoajoiksi eri pakkausmenetelmillä pakatuille Bristol-kuville saadaan taulukon 7.2 mukaiset ajat.

*Taulukko 7.2: Eri menetelmillä kompressoitujen kuvien tiedonsiirtoon kuluvia aikoja Bristol-kuville.*

Tiedonsiirtoon kuluva aika	28 800 bps	56 000 bps	128 000 bps	512 000 bps
Pakkaamaton kuva	~9,4 min	~4,8 min	~2,1 min	31,7 s
4:4:4 (3 sisätulokuvaa)	54,6 s	28,1 s	12,3 s	3,1 s
4:2:2 (3 sisätulokuvaa)	36,4 s	18,7 s	8,2 s	2,0 s
4:2:0 (3 sisätulokuvaa)	27,3 s	14,1 s	6,1 s	1,5 s
4:1:1 (3 sisätulokuvaa)	27,3 s	14,1 s	6,1 s	1,5 s
3×3 (3 sisätulokuvaa)	22,3 s	11,5 s	5,0 s	1,3 s
3×3 (4 sisätulokuvaa)	24,4 s	12,5 s	5,5 s	1,4 s
3×3 (5 sisätulokuvaa)	26,5 s	13,6 s	6,0 s	1,5 s

Bristol-kuvien siirtonopeus putoaa alle 10-osaan jo pelkällä spektritason pakkauksella, eikä yhden kuvan siirtoon kulu enää kuin muutama sekunti. Käytännön sovelluksissa joudutaan kuitenkin usein käyttämään huomattavasti isompia kuvia, jolloin spatiaalinenkin pakkaus tulee siirtonopeudessa merkittävään asemaan.

Forest- ja Coral-kuvien kompressiosuhteet ovat taulukossa 7.3 ja tiedonsiirtoon kuluvat teoreettiset ajat eri kompressiomenetelmille sekunteina ovat taulukossa 7.4.

*Taulukko 7.3: Eri menetelmien kompressiosuhteita Forest- ja Coral-kuville.*

Menetelmä	Pakkaamattoman kuvan koko (tavua)	Kompressoitun kuvan koko (tavua)	CR
4:4:4 (3 sisätulokuvaa)	655360	49272	13,3
4:2:2 (3 sisätulokuvaa)	655360	32888	19,9
4:2:0 (3 sisätulokuvaa)	655360	24696	26,5
4:1:1 (3 sisätulokuvaa)	655360	24696	26,5
3×3 (3 sisätulokuvaa)	655360	20202	32,4
3×3 (4 sisätulokuvaa)	655360	22091	29,7
3×3 (5 sisätulokuvaa)	655360	23980	27,3

*Taulukko 7.4: Eri menetelmillä kompressoitujen kuvien tiedonsiirtoon kuluvia aikoja Forest- ja Coral-kuville.*

Tiedonsiirtoon kuluva aika	28 800 bps	56 000 bps	128 000 bps	512 000 bps
Pakkaamaton kuva	~3,0 min	~1,5 min	41,0 s	10,2 s
4:4:4 (3 sisätulokuvaa)	13,7 s	7,0 s	3,1 s	0,8 s
4:2:2 (3 sisätulokuvaa)	9,1 s	4,7 s	2,1 s	0,5 s
4:2:0 (3 sisätulokuvaa)	6,9 s	3,5 s	1,5 s	0,4 s
4:1:1 (3 sisätulokuvaa)	6,9 s	3,5 s	1,5 s	0,4 s
3×3 (3 sisätulokuvaa)	5,6 s	2,9 s	1,3 s	0,3 s
3×3 (4 sisätulokuvaa)	6,1 s	3,2 s	1,4 s	0,3 s
3×3 (5 sisätulokuvaa)	6,7 s	3,4 s	1,5 s	0,4 s

Forest- ja Coral-kuvat ovat spatiaalisesti niin pienikokoisia (128×128), että tiedonsiirtoon kuluvan ajan vertailu keskenään eri menetelmillä on vaikeaa. Ajatellaan kuitenkin, että meillä olisi 1024×1024-kokoinen kuva, jossa olisi 61-kanavaa. Tällöin, jos sisätulokuvien määrä on 3, niin eri menetelmillä tiedonsiirtoon kuluvat ajat olisivat taulukon 7.5 mukaiset.

*Taulukko 7.5: Eri menetelmillä kompressoidun  
 ison kuvan tiedonsiirtoon kuluvia aikoja.*

<b>Tiedonsiirtoon kuluva aika</b>	<b>Kuvan koko (tavua)</b>	<b>28 800 bps</b>	<b>56 000 bps</b>	<b>128 000 bps</b>	<b>512 000 bps</b>
Pakkaamaton kuva	63963136	~4,9 h	~2,5 h	~1,1 h	16,7 min
1×1-kokoinen lohko, 3 sisätulokuvaa	3145911	14,6 min	7,5 min	3,3 min	49 s
2×2-kokoinen lohko, 3 sisätulokuvaa	1573047	7,3 min	3,7 min	1,6 min	25 s
3×3-kokoinen lohko, 3 sisätulokuvaa	1282687	5,9 min	3,1 min	1,3 min	20 s
4×4-kokoinen lohko, 3 sisätulokuvaa	1179831	5,5 min	2,8 min	1,2 min	18 s
8×8-kokoinen lohko, 3 sisätulokuvaa	1081527	5,0 min	2,6 min	1,1 min	17 s



## 8. POHDINTA

Lopuksi kerron lyhyesti työn kulun sekä analysoin saamiani tuloksia.

### 8.1 Työn kulku

Värispektrikuvat suodatettiin aluksi keskiarvostuksella, minkä jälkeen suodatettua kuvaa käsiteltiin alkuperäisenä kuvana. Spektrin äkillisiä muutoksia ja piikkejä saatiin näin vähennettyä.

Työssäni spektritason pakkaus tehtiin lähinnä 3 kantavektorilla ja niin ikään 3 sisätulokuvaa käyttäen. Esittelin lyhyesti myös 4 ja 5 kantavektorin ja sisätulokuvan käyttöä 3×3-keskipistemenetelmässä, ja MacDonald et al. [18] sekä Liu [17] käyttävätkin spektritason pakkauksessa eri määriä kantavektoreja/sisätulokuvia. Selvää on, että mitä enemmän kantavektoreja ja sisätulokuvia pakkaukseen otetaan, sitä enemmän pakatussa kuvassa on tietoa alkuperäisestä kuvasta. Tällöin virhe vastaavasti pienenee.

Spatiaalisessa pakkauksessa käsittelin ensin erikseen JPEG- ja MPEG-muotoon perustuvia menetelmiä 3 sisätulokuvalla, missä kromaattisia kanavia vastaavat 2. ja 3. sisätulokuva jaettiin menetelmästä riippuen 1×2-kokoisiin (4:2:2-menetelmä), 2×2-kokoisiin (4:2:0-menetelmä) ja 1×4-kokoisiin (4:1:1-menetelmä) lohkoihin, minkä kulma-arvo (lohkon vasen yläkulma) korvasi koko lohkon. Lisäksi otin näihin vertailuksi menetelmät, jossa spatiaalista pakkausta ei tehty lainkaan (4:4:4-menetelmä) sekä 3×3-keskipistemenetelmän, missä lohko korvattiin ko. lohkon keskellä olevalla arvolla. Selvitin myös keskiarvo- ja mediaanimenetelmää 3×3-lohkokoolla ja vertailun vuoksi laskin vielä 3×3-kulmapistemenetelmän. Lisäksi kokeilin 3×3-keskipistemenetelmää vielä 4 ja 5 sisätulokuvalla. Spatiaalista pakkausta voi myös tutkia käyttämällä isompiakin lohkoja, ja käyttää lohkon korvaavana arvona esim. lohkon kulma-arvoa, keskellä sijaitsevaa arvoa, keskiarvoa tai mediaania. Lisäksi 1. sisätulokuva voitaisiin myös pakata esim. diskreetillä kosinimuunnoksella tai häviöttömällä pakkauksella kompressiosuhteen parantamiseksi.

### 8.2 Työn analysointi

Spektrin äkilliset muutokset johtuvat usein kuvausvirheistä, eli esimerkiksi siitä, että aurinko on juuri menossa pilveen kuvaa otettaessa. Valoisuuden muutos aiheuttaa spektrissä äkillisiä

muutoksia. Suodatuksen avulla saadaan näitä kuvauksen yhteydessä tulleita virheitä pienemmäksi.

Spektritason pakkaus PCA-menetelmällä pienentää kuvan kokoa huomattavasti säilyttäen kuitenkin hyvin kuvainformaatiota. Jo muutamalla sisätulokuvalla päästään niin kelvolliseen lopputulokseen, että kuvan väriä voi hyvin kompressoida spatiaalitasossakin. Testiaineistossa jo pelkkä spektritason pakkaus 3 sisätulokuvalla pudotti kuvan kokoa alle 10 prosenttiin alkuperäisestä, ja taulukon 7.5 mukaisessa isossa kuvassa 5 prosenttiin. Kantavektoreissa säilyvän tiedon määrä on Forest-kuvasarjassa keskimäärin 99,80%. Coral-kuvasarjassa saatiin hieman pienempi tulos (99,65%), sillä Coral-kuvissa on enemmän erilaisia värejä.

4:4:4-menetelmän virhe on 3 sisätulokuvalla luonnollisesti kaikkein pienin, sillä siinä ei tehdä spatiaalista pakkausta lainkaan. Vastaavasti, mitä suurempaa lohkoa spatiaalisessa pakkauksessa käytetään, sitä suuremmaksi virhe kasvaa. Tarkastellaan aluksi menetelmiä 4:4:4, 4:2:2, 4:2:0, 4:1:1 sekä 3x3-keskipistemenetelmää. Jos tarkastellaan S-CIELAB  $\Delta E$ -virhettä, saadaan Forest-kuvasarjalle keskimäärin virheeksi 1,0...2,0 ja Coral-kuvasarjalle 2,2...2,7 menetelmästä riippuen. Tämä ei kuitenkaan ole kovin suuri kompressiosuhteeseen nähden. Virheitä tosin näkee ihmissilmällä, ainakin jos alkuperäistä ja rekonstruoitua kuvaa katselee vierekkäin, mutta rekonstruoitu kuva on edelleen kelvollinen moneen tarkoitukseen. Mielenkiintoista on, että 3x3-keskipistemenetelmän ja 4:1:1-menetelmän virheet olivat keskenään lähes samat, vaikka 3x3-keskipistemenetelmässä pakataan kromaattisia kanavia vastaavia sisätulokuvia yli kaksinkertaisesti. Lohkon etäisin korvattava pikseli on 4:1:1-menetelmässä jo niin kaukana korvaavasta pikselistä, että se muuttaa kuvaa huomattavasti (kuva 6.13). Virheellisyys ja tilavaativuus keskenään tekevätkin 4:1:1-menetelmän tarpeellisuudesta hyvin kyseenalaisen.

Yleisesti koko Forest-kuvasarjalle keskimääräinen  $\Delta E$ -virheen suuruus on spatiaalisessa pakkauksessa 3x3-keskiarvomenetelmällä ja 3x3-mediaanimenetelmällä parempi kuin 4:2:0-menetelmällä, vaikka kromaattisia kanavia vastaavia sisätulokuvia pakataankin yli kaksinkertaisesti. Keskiarvomenetelmä kuitenkin haalistaa kuvaa enemmän, sillä korvaavassa pikselissä on tietoa kaikista lohkon väreistä. Lisäksi keskiarvomenetelmän virheellisyys riippuu siitä, minkälaista jakaumaa lohkon arvot noudattavat. Mediaanimenetelmässä taas korvaava pikseli huomioi lohkon jakauman, mitä muut menetelmät eivät tee. Edelleen, jos

kuva pakataan spektritasossa 5 sisätulokuvalla ja spatiaalitasossa 3×3-keskipistemenetelmällä, huomataan tuloksista, että 3 sisätulokuvalla vastaava keskiarvomenetelmä ja mediaanimenetelmä antavat edelleen pienempiä virheitä. Liitteessä 5 on vielä vertailtu eri 3×3-menetelmiä keskenään kahden hyvin yksinkertaistetun esimerkin kautta. Näistä esimerkeistä nähdään, että valittaessa oikea pakkausmenetelmä oikeaan tarkoitukseen voidaan mahdollisesti myös eliminoida joitakin kuvausvirheitä tai vastaavasti purkuvaiheessa sopivalla korjausalgoritmilla mahdollisesti palauttaa kuvasta tietoa vielä lisää.

Värispektrikuvien pakkaussuhde riippuu alkuperäisen kuvan kanavien lukumäärästä, kanta-vektorien (ja sisätulokuvien) määrästä sekä käytettävästä spatiaalitason pakkausmenetelmästä. Kaikkiaan pakkaussuhde on kaikilla menetelmillä erinomainen. Bristol-kuville tämä on pienempi kuin Forest-/Coral-kuville, koska alkuperäisissä Bristol-kuvissa on vähemmän kanavia. Bristol-kuvien pakkaussuhde  $CR$  vaihtelee välillä  $10,0...25,3$  menetelmästä riippuen, eli kuva saadaan  $4...10$  prosenttiin alkuperäisestä. Vastaavasti Forest-/Coral-kuville  $CR$  on  $13,3...32,4$  menetelmästä riippuen, jolloin kuva pakkautuu  $3...8$  prosenttiin alkuperäisestä. Pakatessa isoa (taulukon 7.5 mukaista) kuvaa tämä pakkautuu  $2...5$  prosenttiin alkuperäisestä.

Kaiken kaikkiaan keskiarvomenetelmässä ja mediaanimenetelmässä on siis runsaasti etuja (jo huomattavasti pienemmän virheellisyydenkin vuoksi), mutta tässä vaiheessa ei voida todeta, että nämä olisivat aina parempia kuin kulmapiste- ja keskipistemenetelmät. Pakattaessa  $n$ -alkioinen lohko keskiarvomenetelmällä, on lohkon keskiarvon laskemisen aikavaativuus  $O(n)$ . Mediaanimenetelmällä pakatessa joudutaan mediaani hakemaan käytännössä lajittelemalla lohko, minkä aikavaativuus on  $O(n \cdot \log_2 n)$ . Muut menetelmät ovat vakioaikaisia  $O(1)$  ja siten nopeampia. Jatkossa tulisikin keskittyä erityisesti spatiaalisen pakkausmenetelmän valintaan sekä myös sisätulokuvien määrään. Lisäksi eri menetelmille voitaisiin jo nyt erilaisiin tarkoituksiin luoda korjausalgoritmeja esim. heuristisesti, kuten liitteen 5 yksinkertaistetuissa esimerkeissä tehtiin.

## VIITELUETTELO

- [1] Ajito, T., Obi, T., Yamaguchi, M. and Ohyama, N.: Six-primary color projection display for expanded color gamut reproduction. *In Proceedings, International Symposium on Multispectral Imaging and Color Reproduction, Chiba, Japan, October 21-22*, pp. 135-138, 1999.
- [2] Astola, J. and Kuosmanen, P.: *Fundamentals of Nonlinear Digital Filtering*. CRC Press Inc., USA, 1997.
- [3] Berns, R.S.: *Billmeyer and Saltzman's Principles of Color Technology, third edition*. John Wiley & Sons, Inc., USA, 2000.
- [4] Chiao, C.-C., Cronin T.W. and Osorio, D.: Color signals in natural scenes: characteristics of reflectance spectra and effects of natural illuminants. *Journal of the Optical Society of America A 17(2)*, pp. 218-224, 2000.
- [5] Frydrych, M.: *Multispectral Images: JOSA*. Internet WWW-sivu, URL: <http://www.it.lut.fi/research/ip/images/Spectral/JOSA/index.html> (07.02.2002)
- [6] Giorgianni, E.J. and Madden, T.E.: *Digital Color Management Encoding Solutions*. Addison Wesley Longman, Inc., USA, 1998.
- [7] Hauta-Kasari, M., Lehtonen, J., Parkkinen, J. and Jaaskelainen, T.: Spectral Image Compression for Data Communications. *In Proceedings of SPIE, Electronic Imaging 2001, San Jose, California, USA, January 21-26, vol. 4300*, pp. 42-49, 2001.
- [8] Hauta-Kasari, M.: *Computational Techniques for Spectral Image Analysis*. Väitöskirja. Lappeenrannan teknillinen korkeakoulu, Lappeenranta, 1999.
- [9] Hyvärinen, A. and Oja, E.: Independent Component Analysis: Algorithms and Applications. *Neural Networks, 13(4-5)*, pp. 411-430, 2000.

- [10] Jaaskelainen, T., Parkkinen, J. and Toyooka, S.: Vector-subspace model for color representation. *Journal of the Optical Society of America A* 7(4), pp. 725-730, 1990.
- [11] Kaarna, A.: *Multispectral Image Compression Using the Wavelet Transform*. Väitöskirja. Lappeenrannan teknillinen korkeakoulu, Lappeenranta, 2000.
- [12] Kondou, W., Miyata, K., Haneishi H. and Miyake, Y.: An Evaluation of Image Quality for Compressed Multi-spectral Image. In *Proceedings, International Symposium on Multispectral Imaging and Color Reproduction, Chiba, Japan, October 21-22*, pp. 143-146, 1999.
- [13] Koponen, P.: *Spektrivideo*. Diplomityö. Lappeenrannan teknillinen korkeakoulu, Lappeenranta, 1999.
- [14] Koponen, P., Kälviäinen, H., and Parkkinen, J.: Multispectral Video. *15th International Conference on Pattern Recognition, Barcelona, Spain, September 3-8, Vol. 3: Image, Speech and Signal Processing*, pp. 190-193, 2000.
- [15] König, F. and Praefcke, W.: A Multispectral Scanner. *Colour Imaging Vision and Technology* (Ed. MacDonald, L. and Luo, M.R.), John Wiley & Sons ltd., pp. 129-143, 1999.
- [16] König, F. and Praefcke, W.: Multispectral Image Encoding. In *Proceedings, IEEE International Conference on Image Processing, ICIP, Kobe, Japan, October 24-28, vol. 3*, pp. 123-126, 1999
- [17] Liu, D.: *Image Compression through Principal Components*. M.Sc. Thesis. Colour & Imaging Institute, University of Derby, Iso-Britannia, 2001.
- [18] MacDonald, L., Westland, S. and Liu, D.: Multispectral Image Encoding and Compression. In *Proceedings, 9th CIC Color Imaging Conference, USA, 2001*.
- [19] MacDonald, L., Westland, S. and Shaw, J.: Colour Image Reproduction: Spectral vs. Spatial. In *Proceedings, International Symposium on Multispectral Imaging and Color Reproduction, Chiba, Japan, October 21-22*, pp. 81-91, 1999.

- [20] Oja, E.: *Subspace methods of Pattern Recognition*. John Wiley & Sons, Inc., USA, 1983.
- [21] Parkkinen, J. and Jaaskelainen, T.: Color representation using statistical pattern recognition. *Applied Optics* 26(19), pp. 4240-4245, 1987.
- [22] Parkkinen, J., Hauta-Kasari, M., Kaarna, A., Lehtonen, J., Koponen, P. and Jaaskelainen, T.: Multispectral Image Compression. *In Proceedings, Second International Symposium on Multispectral Imaging and High Accurate Color Reproduction, Chiba, Japan, October 10-11*, pp. 49-58, 2000.
- [23] Parkkinen, J.P.S., Hallikainen, J. and Jaaskelainen, T.: Characteristic spectra of Munsell Colors. *Journal of the Optical Society of America A* 6(2), pp. 318-322, 1989.
- [24] Párraga, C.A., Brellstaff, G., Troscianko, T. and Moorehead, I.R.: Color and luminance information in natural scenes. *Journal of the Optical Society of America A* 15(3), pp. 563-569, 1998.
- [25] Wachtler, T., Lee, T.-W. and Sejnowski, T.J.: Chromatic structures of natural scenes. *Journal of the Optical Society of America A* 18(1), pp. 65-77, 2001.
- [26] Wu, C.-H. and Irwin J.D.: *Emerging Multimedia Computer Communication Technologies*. Prentice-Hall, Inc., USA, 1998.
- [27] Wyszecki, G. and Stiles, W.S.: *Color science : concepts and methods, quantitative data and formulae, 2nd ed.* John Wiley & Sons, Inc., USA, 1982.
- [28] Zhang, X. and Wandell, B.A.: A Spatial Extension of CIELAB for Digital Color Image Reproduction. *In Proceedings, Society for Information Display Symposium Technical Digest, vol. 27*, pp. 731-734, 1996.
- [29] Zhang, X.: S-CIELAB: A Spatial Extension to the CIE  $L^*a^*b^*$   $\Delta E$  Color Difference Metric. Internet WWW-sivu, URL: <http://white.stanford.edu/~brian/scielab/scielab.html> (07.02.2002).

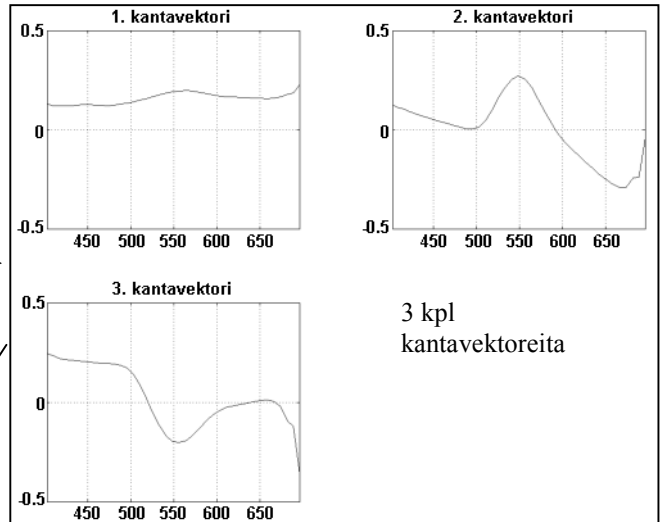
# Liite 1: Kuvien pakkaus ja purku

Spektritason pakkaus:

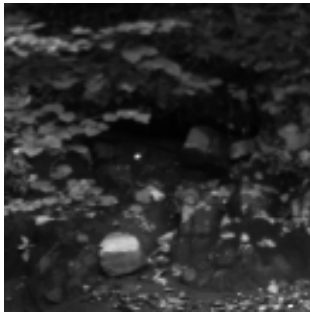


Värispektrikuva

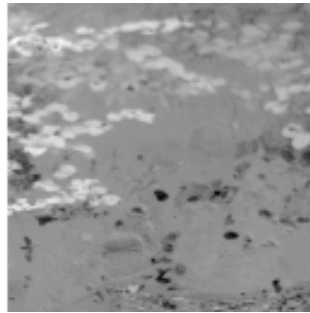
Pääkomponentti-analyysi PCA



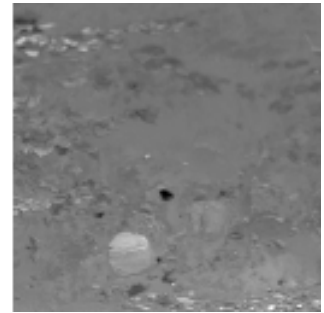
Lasketaan kantavektorien ja alkuperäisen kuvan avulla sisätulokuvat.



1. sisätulokuva



2. sisätulokuva

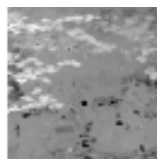


3. sisätulokuva

Spatiaalitason pakkaus:



1. sisätulokuva



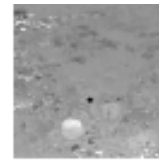
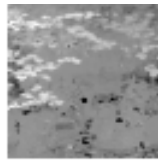
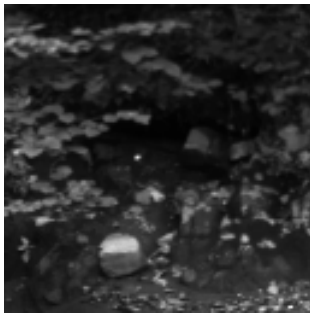
4:2:0-menetelmällä pakattu 2. sisätulokuva



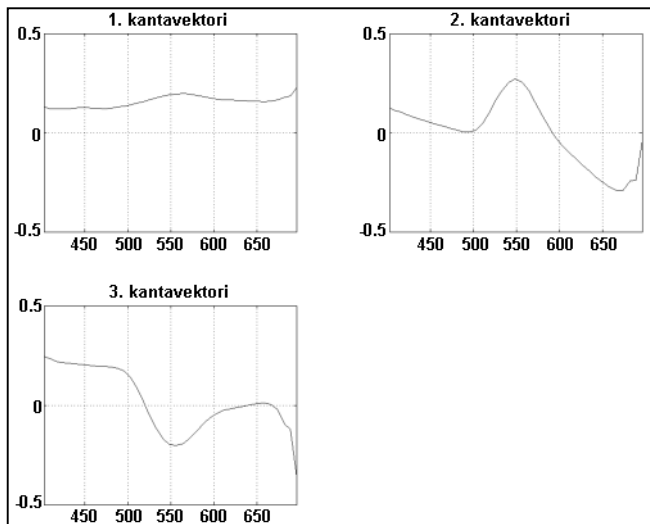
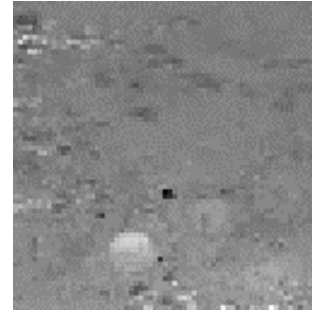
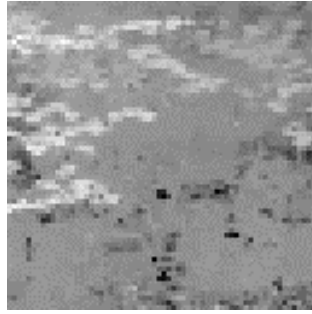
4:2:0-menetelmällä pakattu 3. sisätulokuva

Kompressoidun kuvan muodostavat kantavektorit ja pakatut sisätulokuvat (3 kpl).

Pakatun värispektrikuvan purku:

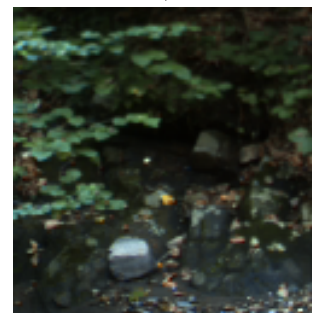


2. ja 3. sisätulokuva laajennetaan alkuperäisien kuvien kokoiseksi siten, että kukin arvo kopioidaan 2x2-kokoisen lohkon puuttuviin arvoihin.



$$\bar{s}_i = \sum_{j=1}^k \bar{e}_j \cdot \tilde{p}_{j,i}$$

jokaiselle pikselille.



Purettu värispektrikuva



## Liite 2: Spatiaalisen pakkauksen aiheuttamaan virheeseen liittyvät todistukset.

**Väite:** Menetelmät 4:4:4 ja 1×1-keskiarvo antavat yhtä suuret virheet lohkoissa.

**Todistus:** Triviaali. Olkoon  $P$  spektritasossa pakattu, mutta spatiaalitasossa pakkaamaton sisätulokuvista ( $k$  kpl) muodostettu kuva. Olkoon  $A$  1×1-kokoinen lohko  $P$ :n jostakin sisätulokuvasta 2.. $k$ , joka sisältää ainoastaan mielivaltaisen arvon  $a$ . Tällöin 4:4:4-menetelmän pakkauksen jälkeen lohkoissa  $A$  ei tapahdu muutoksia, lohkoissa säilyy sama arvo  $a$ . Keskiarvo lohkoista  $A$  on sama arvo  $a$ , sillä se on ainoa lohkon arvo. Näin ollen väite on tosi.

**Väite:** Menetelmät 4:2:2 ja 1×2-keskiarvo antavat yhtä suuret virheet lohkoissa.

**Todistus:** Olkoon  $P$  spektritasossa pakattu, mutta spatiaalitasossa pakkaamaton sisätulokuvista ( $k$  kpl) muodostettu kuva. Olkoon  $P$ :n jostakin sisätulokuvasta 2.. $k$  otetussa 1×2-kokoisessa lohkoissa  $A$  ainoastaan mielivaltaiset arvot  $a$  ja  $b$ . Tällöin 4:2:2-menetelmällä tehdyn pakkauksen jälkeen ko. lohkon virhe on  $|a - b|$ , sillä arvo  $a$  ei muutu, kun taas arvo  $b$  korvautuu arvolla  $a$ .

Keskiarvomenetelmällä laskettu virhe on  $\left|a - \frac{a+b}{2}\right| + \left|b - \frac{a+b}{2}\right|$ , sillä arvot  $a$  ja  $b$  korvataan näiden keskiarvolla. Virheet ovat yhtä suuret, sillä

$$\left|a - \frac{a+b}{2}\right| + \left|b - \frac{a+b}{2}\right| = \left|\frac{b-a}{2}\right| + \left|\frac{a-b}{2}\right| = \left|\frac{a-b}{2}\right| + \left|\frac{a-b}{2}\right| = |a-b|.$$

Näin ollen väite on tosi.

**Väite:** Olkoon menetelmässä  $V$  lohkon korvaava arvo  $u \in R$ . Seuraavat kohdat pätevät yleisesti:

1. Ei ole olemassa arvoa  $u \in R$  (eikä siten menetelmää  $V$ ), joka lohkon korvatessaan aiheuttaisi lohkoissa aidosti pienemmän virheen kuin mediaanimenetelmä.
2. Mediaanimenetelmä antaa lohkoissa aidosti pienemmän virheen verrattuna menetelmään  $V$  jos ja vain jos  $u$  ei ole lohkon keskimäinen arvo lohkon alkioiden määrän ollessa pariton, ja kun  $u$  ei ole lohkon keskimäisten alkioiden suljetulla välillä lohkon alkioiden määrän ollessa parillinen.
3. Muussa tapauksessa mediaanimenetelmä antaa lohkoissa yhtä suuren virheen kuin menetelmä  $V$ .

**Todistus:** Olkoon mielivaltainen  $n$ -alkioinen lohko  $A = (a_1 \ a_2 \ \dots \ a_n)$  nousevasti järjestetty siten, että  $a_1 \leq a_2 \leq \dots \leq a_n$ . Oletetaan myös, että  $a_1 \leq a_2 \leq \dots \leq a_{x-1} \leq u \leq a_x \leq \dots \leq a_n$  ja  $a_1 \leq a_2 \leq \dots \leq a_{y-1} \leq m \leq a_y \leq \dots \leq a_n$  missä  $u$  on lohkon korvaava pikseli menetelmässä  $V$

ja  $m$  lohkon arvojen mediaani. Lisäksi  $x \in \{1, 2, 3, \dots, n+1\}$  ja  $y = \begin{cases} \frac{n}{2} + 1, & \text{kun } n = 2, 4, 6, \dots \\ \frac{n+1}{2} & \text{kun } n = 1, 3, 5, \dots \end{cases}$

Määritetään vielä, että nämä indeksit  $x$  ja  $y$  on sijoitettava mahdollisimman lähelle toisiaan.

Tällöin mediaanimenetelmän aiheuttama virhe lohkossa on

$$M = \sum_{i=1}^n |a_i - m| = \sum_{i=1}^{y-1} (m - a_i) + \sum_{i=y}^n (a_i - m) = M_1 + M_2$$

ja menetelmän  $V$  aiheuttama virhe

$$U = \sum_{i=1}^n |a_i - u| = \sum_{i=1}^{x-1} (u - a_i) + \sum_{i=x}^n (a_i - u) = U_1 + U_2.$$

**Vaihe 1:** Olkoon  $u < m$ . Tällöin  $x \leq y$  sekä

$$U_1 - M_1 = M_1 - (x-1) \cdot |u - m| - M_1$$

ja

$$U_2 - M_2 = M_2 + (n - y + 1) \cdot |u - m| + \sum_{i=x}^{y-1} (a_i - u) - M_2,$$

mistä seuraa, että

$$U - M = U_1 - M_1 + U_2 - M_2 = (n - y + 1 - (x-1)) \cdot |u - m| + \sum_{i=x}^{y-1} (a_i - u), \text{ missä } \sum_{i=y}^{x-1} (u - a_i) \geq 0.$$

Tästä johtopäätökset:

$$n - y + 1 > x - 1 \Leftrightarrow \begin{cases} x < \frac{n}{2} + 1 = y, & \text{missä } n = 2, 4, 6, \dots \\ x < \frac{n+3}{2} = y+1, & \text{missä } n = 1, 3, 5, \dots \end{cases} \Rightarrow U - M > 0$$

$$n - y + 1 = x - 1 \Leftrightarrow \begin{cases} x = \frac{n}{2} + 1 = y, & \text{missä } n = 2, 4, 6, \dots \Rightarrow U - M = 0 \\ x = \frac{n+3}{2} = y+1, & \text{missä } n = 1, 3, 5, \dots \Rightarrow RR \end{cases}$$

$$n - y + 1 < x - 1 \Leftrightarrow \begin{cases} x > \frac{n}{2} + 1 = y, & \text{missä } n = 2, 4, 6, \dots \\ x > \frac{n+3}{2} = y+1, & \text{missä } n = 1, 3, 5, \dots \end{cases} \Rightarrow RR$$

**Vaihe 2:** Olkoon  $u > m$ . Tällöin  $x \geq y$  sekä

$$U_1 - M_1 = M_1 + (y-1) \cdot |u-m| + \sum_{i=y}^{x-1} (u-a_i) - M_1$$

ja

$$U_2 - M_2 = M_2 - (n-x+1) \cdot |u-m| - M_2,$$

mistä seuraa, että

$$U - M = U_1 - M_1 + U_2 - M_2 = (y-1 - (n-x+1)) \cdot |u-m| + \sum_{i=y}^{x-1} (u-a_i), \text{ missä } \sum_{i=y}^{x-1} (u-a_i) \geq 0.$$

Tästä johtopäätökset:

$$y-1 > n-x+1 \Leftrightarrow \begin{cases} \frac{n}{2} + 1 = y < x, & \text{missä } n = 2, 4, 6, \dots \\ \frac{n+3}{2} = y+1 < x, & \text{missä } n = 1, 3, 5, \dots \end{cases} \Rightarrow U - M > 0$$

$$y-1 = n-x+1 \Leftrightarrow \begin{cases} \frac{n}{2} + 1 = y = x, & \text{missä } n = 2, 4, 6, \dots \\ \frac{n+3}{2} = y+1 = x, & \text{missä } n = 1, 3, 5, \dots \end{cases}$$

$$\Leftrightarrow \begin{cases} \frac{n}{2} + 1 = y = x \Rightarrow U - M = 0 \\ \frac{n+3}{2} = y+1 = x \Rightarrow U - M = u - a_{x-1} = u - m > 0 \end{cases}$$

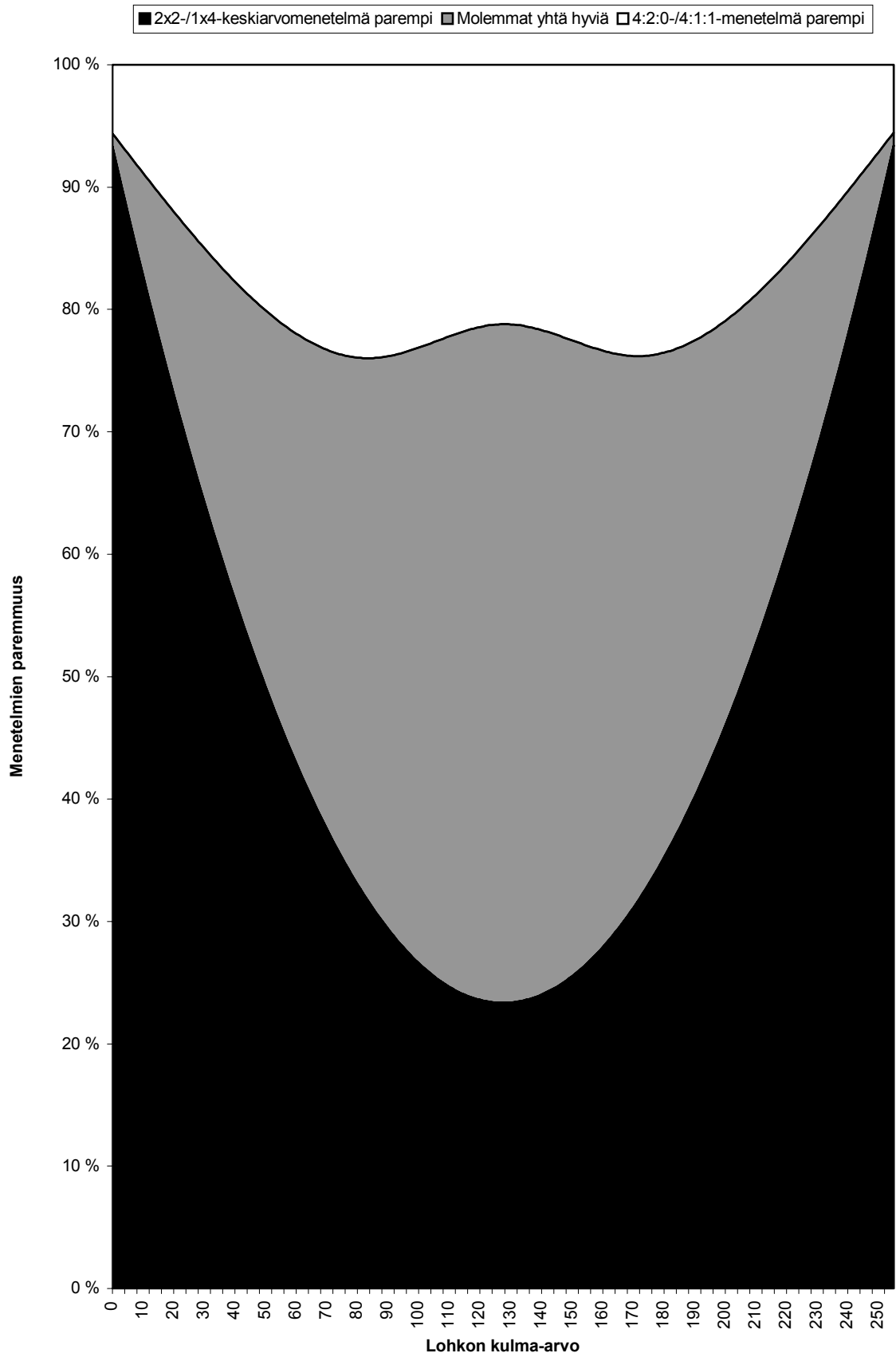
$$y-1 < n-x+1 \Leftrightarrow \begin{cases} x < \frac{n}{2} + 1 = y, & \text{missä } n = 2, 4, 6, \dots \\ x < \frac{n+3}{2} = y+1, & \text{missä } n = 1, 3, 5, \dots \end{cases} \Rightarrow RR$$

**Vaihe 3:** Olkoon  $u = m$ . Tällöin menetelmä  $V$  vastaa mediaanimenetelmää ja siten

$$U - M = 0.$$

Tästä seuraa, että väite on tosi.

### Liite 3: 2x2-/1x4-keskiarvomenetelmä vs. 4:2:0-/4:1:1-menetelmä



**Liite 4a: Bristol-kuvasarja**

ashton2



ashton2b



ashton3



cold1



cold3



fern1



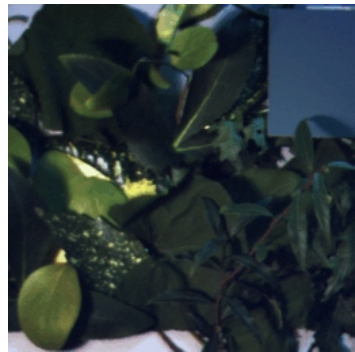
ferns2



fort04



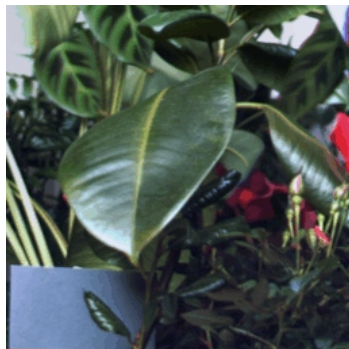
gleaves



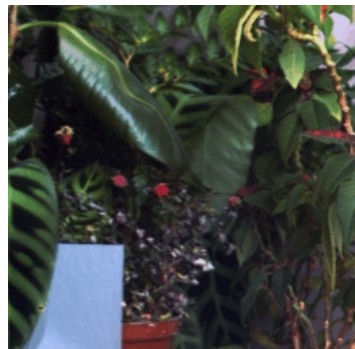
inlab1



inlab2



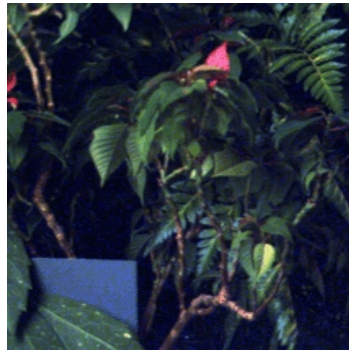
inlab3



inlab4



inlab5



inlab7



jan10pm



jan13am



moss



pink7



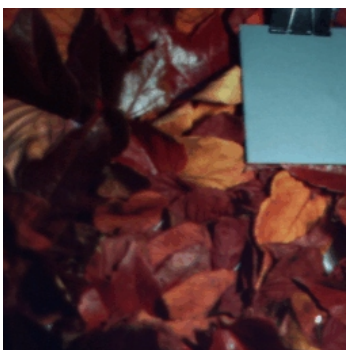
plaza



red1



rleaves



rocks



rwood



valley



windy



yellow1

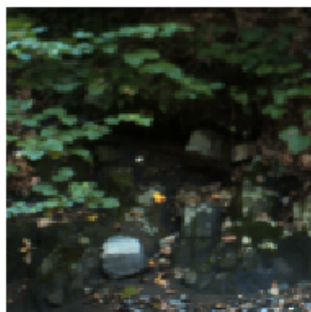


yleaves

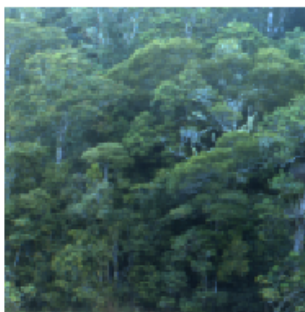


**Liite 4b: Forest-kuvasarja**

bank



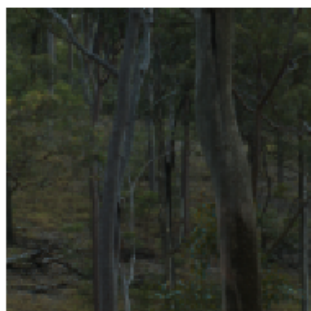
barrine2



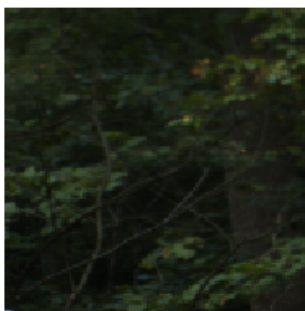
cooth12



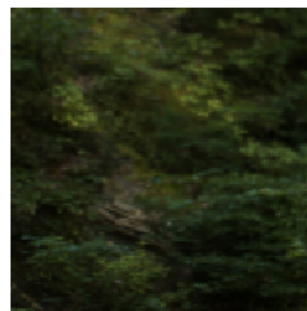
coottha8



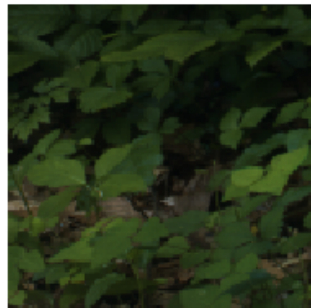
glade



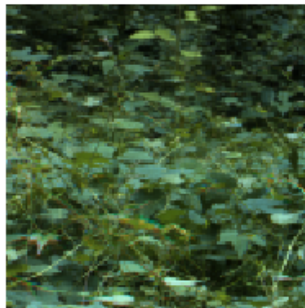
hillside



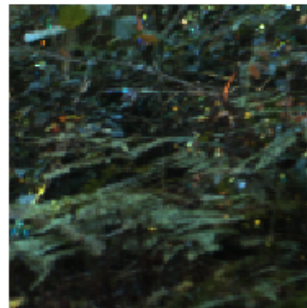
park2



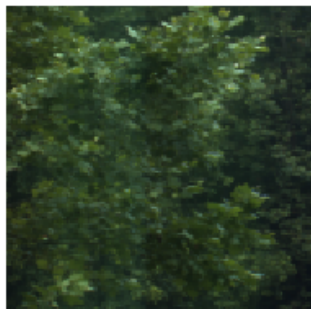
park4



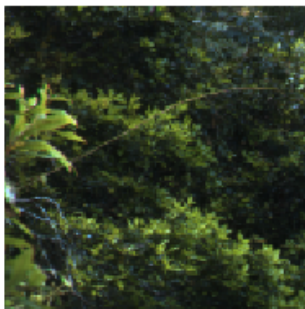
rainfor5



riverrd2



treetop4



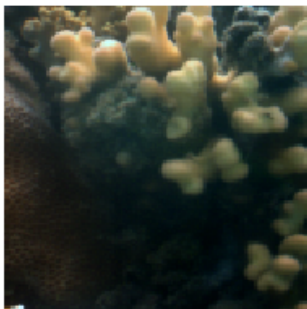
wynnam4





**Liite 4c: Coral-kuvasarja**

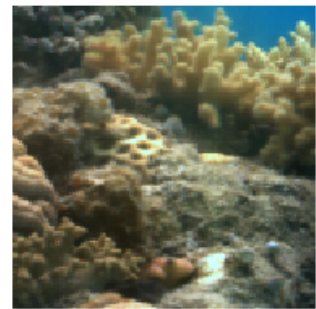
horshe10



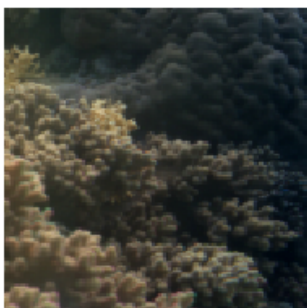
horshe19



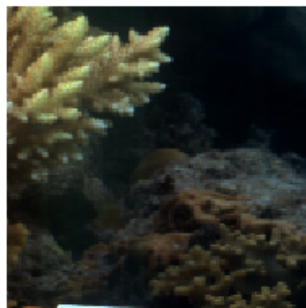
horshe29



horshe32



horshe5



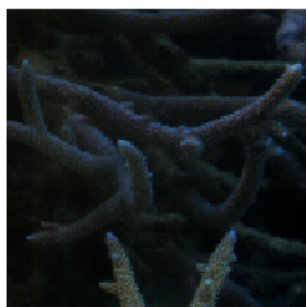
hoshoe12



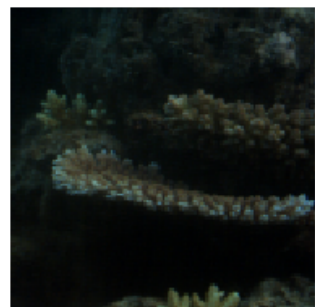
hoshoe24



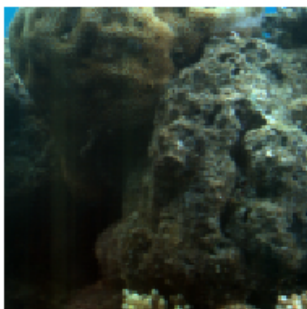
hoshoe27



hoshoe30

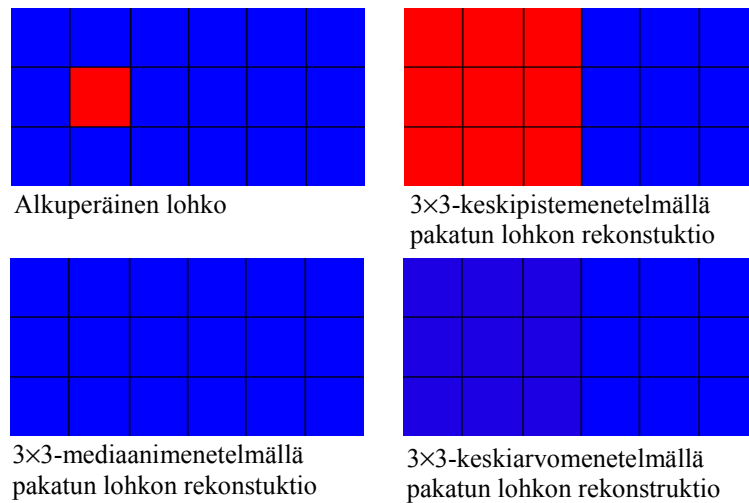


hoshoe36



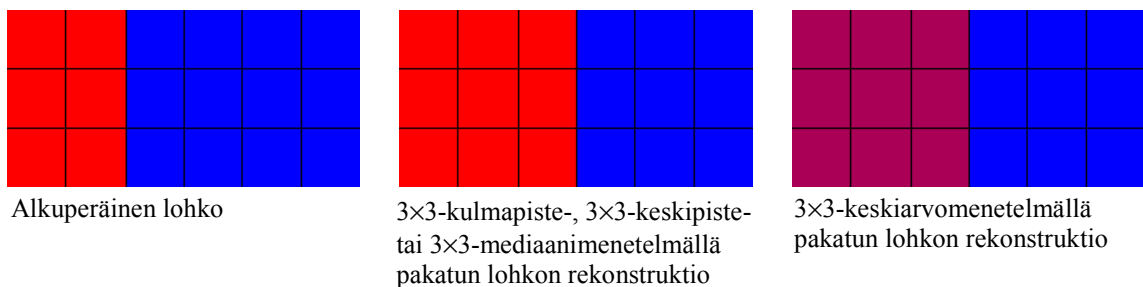
## Liite 5: Pohdintaesimerkkejä

Oletetaan, että meillä olisi sininen alue, missä yksi pikseli olisi punainen esim. kuvausvirheen vuoksi (kuva 1). Kulmapiste- tai keskipistemenetelmässä virheellisen pikselin ollessa juuri lohkon korvaava pikseli, on koko lohko punainen pakattua kuvaa purettaessa. Koska mediaanimenetelmä huomioi lohkon värijakauman, saadaan vain tällä menetelmällä varmasti punainen piste eliminoitu. Vastaavasti, mikäli punainen piste olisikin kuvattavassa kohteessa oikeasti, jättäisi ainoastaan keskiarvomenetelmä tästä (varmasti) pakattuun kuvaan tietoa, sillä se levittäisi punaisen pisteen koko lohkolle sinisen sekaan.



Kuva 1: Piste-esimerkki.

Toisena esimerkkinä voidaan ajatella, että meillä olisi kuvattuna punaisen kappaleen ja sinisen alustan välinen raja (kuva 2). Lisäksi tietäisimme, että raja on pystysuora, ja että rajan vasen puoli on punainen ja oikea puoli on sininen, mutta emme tiedä rajan sijaintia.



Kuva 2: Linja-esimerkki.

Kulmapistemenetelmällä pakatusta kuvasta ei voida tietää, oliko alkuperäisessä kuvassa vain ensimmäinen sarake punainen ja muut sinisiä. Vastaavasti keskipiste-/mediaanimenetelmällä pakatusta kuvasta on mahdotonta jälkeinpäin selvittää, oliko alkuperäisessä kuvassa koko 3×3-lohko punainen, vai vain sen kaksi ensimmäistä saraketta. Ainoastaan keskiarvomenetelmällä pakatessa voidaan violetti lohko mahdollisesti palauttaa alkuperäiseksi, sillä punaisen ja sinisen raja riippuu violetin punaisuudesta/sinisyydestä.