

KUVIEN LAADUN ANALYSOINTI

Lasse Makkonen

1.7.2003

Joensuun Yliopisto

Tietojenkäsittelytiede

Pro gradu –tutkielma

Tiivistelmä

Tutkielmassa luodaan katsaus kirjallisuudessa esiintyviin digitaalisten kuvien laadullisen analysoinnin menetelmiin. Valikoitua osaa näistä myös sovelletaan kokeellisesti joukolle testikuvia. Tarkoituksena on löytää se kuvissa esiintyvä piirre, joka olisi ominainen erilaisia painopapereita käyttäville painotuotoksille. Analysointityökaluja etsitään kuvien laatumittojen, sisältöpohjaisen kuvatietokantahaun sekä eräiden muiden tunnettujen kuvankäsittelymenetelmien piiristä. Valittujen menetelmien avulla tuotettuja piirrevektoreita visualisoidaan kokeellisessa osassa mm. itseorganisoivan kartan avulla. Lisäksi esitellään uudenlainen sovellettu piirre, niin ikään itseorganisoivan kartan avulla ratkaistava 'vallitsevimpien' värien joukko. Tämä piirre tuottaakin ns. spektrikuvien laatumitta -piirteen ohella kaikkein lupaavimpia tuloksia. Tutkielma on osa laajempaa tutkimusta, jonka päämääränä on parantaa painotuotteiden laatua erilaisissa painatusolosuhteissa.

Avainsanat: kuva-analyysi, laatumitat, itseorganisoiva kartta (SOM).

Sisällysluettelo

1	Johdanto	1
2	Kuvien laadun analysointi	4
2.1	DIGITAALINEN KUVA	5
2.2	VÄRIMALLIT	6
2.3	SISÄLTÖPOHJAINEN KUVATIETOKANTAHAKU	9
2.3.1	Kuvahakujärjestelmät	10
2.3.2	Peruskäsitteitä.....	11
2.3.3	Etäisyysmitat	13
2.3.4	Värihistogrammit	15
2.3.5	Värikkorelogrammit.....	18
2.3.6	Muita tekniikoita.....	20
2.4	KUVIEN LAATUMITAT	24
2.4.1	Ihmisen näköjärjestelmä	28
2.4.2	Havaintokykyyn perustuva kuvien vertailu	30
2.4.3	Yleinen kuvien laatuindeksi.....	32
2.4.4	Laatumitta spektrikuvien tiivistykselle.....	34
2.5	LBP-TEKSTUURIPIIRRE	38
2.6	ITSEORGANISOIVA KARTTA	40
3	Käytetyt menetelmät ja testiaineisto	45
3.1	MENETELMIEN VALINNASTA	45
3.2	TESTIMATERIAALI	46
4	Kokeelliset tulokset	50
4.1	TILASTOLLINEN ANALYYSI	51
4.2	VÄRIAVARUUSPROJEKTIOT	55
4.3	VALLITSEVA VÄRI	57
4.4	SOM-ANALYYSI	59
5	Pohdinta	66
	Viiteluettelo	73
	Liite 1: Laplace-suodinmatriisit	
	Liite 2: Olohuone-kuvasarja	
	Liite 3: Syötekuvat <i>kakku</i> ja <i>parveke</i>	
	Liite 4: SOM-visualisoinnit syötekuvasta <i>kakku</i>	
	Liite 5: SOM-visualisoinnit syötekuvasta <i>parveke</i>	
	Liite 6: Syötekuvien väriavaruusprojektioita	

1 Johdanto

Tämä tutkielma on osa M-realin ja Joensuun yliopiston tietojenkäsittelytieteen laitoksen värikuvautkimusta. Tutkimuksen tavoitteena on löytää sellaisia kuvankäsittelymenetelmiä, joiden avulla mm. erilaisille papereille painettujen kuvien laatu voitaisiin maksimoida riippumatta painatuksessa vallitsevista olosuhteista. Painatusolosuhteilla tässä yhteydessä tarkoitetaan painatusprosessiin liittyviä muuttuvia parametreja, esimerkiksi käytetyn painopaperin tyyppiä ja laatua sekä väripigmenttien levittämiseen käytettyä tekniikkaa.

Ajatellaan hetki kahta painettua kuvaa. Molemmat kuvat ovat peräisin samasta lähteestä, kenties Suomen tietotoimiston uutiskuva-arkistosta. Ne esittävät samaa kohdetta ja ovat alun perin olleet täysin identtisiä keskenään. Kuvat on kuitenkin painettu eri kohteisiin, esimerkiksi halpaa sanomalehtipaperia käyttävään paikallislehteen sekä toiseen, laadukkaampaa paperia hyödyntävään julkaisuun. Tällä tavoin alun perin identtisille kuville on saattanut aiheutua painoprosessista johtuvia, ihmissilmin selvästi havaittavissa olevia laadullisia eroavaisuuksia. Nämä eroavaisuudet voivat olla erittäin merkittäviä, äärimmäisessä tapauksessa kuvassa alun perin vallinnut yleisilme on saattanut muuttua hyvinkin erilaiseksi, jolloin katsojalle välittyvä kuvallinen viesti ei välttämättä olekaan enää entisen kaltainen.

Painettujen kuvien laadullisten ominaisuuksien tutkiminen on tässä työssä keskeisessä osassa. Tavoitteeseen pyritään kuvien digitaalisten analysointimenetelmien tutkimisen ja kehittämisen keinoin. Pääpaino on sellaisen analysointimenetelmän (tai -menetelmien) etsimisessä, jonka avulla voitaisiin paikallistaa se piirre, joka on jollakin tavalla ominainen tiettyä paperilaatua käyttäville painotuotoksille. Toisin sanoen pyritään selvittämään sitä, mikä antaa tietynlaiselle paperilaadulle painetuille kuville niille tyyppillisen yleisilmeen, mikäli sellainen on löydettävissä. On syytä huomata, että tutkimus on vasta alkuvaiheessa. Tätä tutkielmaa voidaankin ajatella eräänlaisena ensiaskeleena kohti digitaalista painokuvien laadun

analysointia, joka lopulta tietämyksen lisääntyessä vie kohti lopullista päämäärää eli painettujen kuvien laadun parantamista.

Luvussa 2 selvitetään lyhyesti digitaalisen kuva-analyysin peruskäsitteistöä. Näin lukijalla on mahdollisuus tutustua perusasioihin ja samalla myös työssä käytettävään termistöön. Tämän jälkeen luodaan katsaus kirjallisuudesta poimittuihin menetelmiin, jotka liittyvät piirteiden irrottamiseen digitaalisista kuvista. Näistä keskitytään välittömästi tutkimuksen kohteeseen liittyvän materiaalin puutteessa erityisesti sisältöpohjaiseen kuvatietokantahakuun ja erilaisiin kuvien laatumittoihin. Lisäksi esitellään kokeellisessa osassa käytetty tekstuuripiirre LBP sekä lopuksi tässäkin työssä runsaasti hyödynnetty tiedon visualisointimenetelmä, prof. Kohosen itseorganisoiva kartta.

Kokeellisessa osassa käytetty testimateriaali esitellään luvussa 3. Tässä yhteydessä nimetään myös lyhyesti ne menetelmät, joita työssä sovellettiin. Testimateriaalina käytettiin joukkoa erilaisilta painopapereilta digitoituja valokuvia, jotka annettiin syötteeksi mainituille analysointimenetelmille. Kokeelliset tulokset raportoidaan luvussa 4.

Kohdassa 2.3 esitellyistä kuvahakuun liittyvistä piirteistä jatkotarkastelun kohteeksi poimittiin itseoikeutetusti värihistogrammi sekä RGB-väriarvojen käyttö piirrevektoreina. Histogrammit tallentavat tietyn tapahtuman – tässä tapauksessa tietyn värisävyn – globaalin esiintymistiheyden kuvan alueella. Testikuvia vertailtiin näiden piirteiden osalta ja myös saavutettiin osin hyviäkin tuloksia. Laatumittojen edustajana tutkimuksessa käytettiin puolestaan laatumittaa spektrikuvien tiivistykselle, joka esitellään tarkemmin kohdassa 2.4.4. Laatumitta koostuu kolmesta virhekomponentista, jotka mittaavat kuvien kontrastia, spatiaalista rakennetta ja rakeisuutta. Myös tätä piirrettä soveltamalla saadut tulokset olivat kohtuullisia.

Tutkituista piirteistä LBP-tekstuuripiirre osoittautui tarkoitukseen heikoimmin soveltuvaksi. Lisäksi tutkittiin myös erilaisia

väriavaruusprojektioita. Niidenkään osalta tulokset eivät olleet erityisen rohkaisevia. Viimeisessä luvussa 5 on yhteenveto kokeellisista tuloksista ja pohdintaa siitä, mitä seikkoja kannattaa huomioida silloin, kun tutkimusta tulevaisuudessa viedään eteenpäin.

Tärkeimmät tekstissä esiintyvät symbolit on esitetty alla olevassa taulukossa. Eri menetelmien yhteydessä käytettäviä termejä, merkintöjä ja symboleja on pyritty johdonmukaisesti yhtenäistämään.

Symboli	Merkitys
a	autokorrelogrammi
g	korrelogrammi
r	painopiste
s	keskihajonta
c	väri
C	kontrastipiirre
D	yleinen etäisyysfunktio
$e_c, e_s,$ e_k	spektrikuvien virhemitan virhekomponentit
E_c	vektorien välinen Canberra-etäisyys
E_G	LBP/C-jakaumien vertailuun sovitettu etäisyysmetriikka
E_M	värien sijaintikuvaajalle sovitettu etäisyysmetriikka
E_N, q	vektorien kulmien erotukseen pohjautuva etäisyysmetriikka
E_S	spatiaalis-kromaattiselle histogrammille sovitettu etäisyysmetriikka
E_Z	vektorien välinen Czekanowski-kerroin
f	itseorganisoivan kartan naapurustofunktio

Symboli	Merkitys
g	Laplace-suodin
G	yleinen kuvien laatuindeksi
h	histogrammi
H	histogrammiavaruus
I, Q	kuva
L_p	yleistetty Minkowski-metriikka, vektorien välinen etäisyys
m	itseorganisoivan kartan 'malli'
n	värien lukumäärä / ominaisuusvektorin ulotteisuus
p	pikseli
P	pikselijoukko
w	painokerroin
X, Y	kuvan dimensiot
x, y, z	koordinaatteja
$\vec{x}, \vec{y}, \vec{z}$	(ominaisuus)vektoreita, alkiot muotoa x_i

Taulukko 1: Käytetyt symbolit.

2 Kuvien laadun analysointi

Tässä luvussa luodaan katsaus kirjallisuudessa esiintyviin kuvien laadullisen analysoinnin menetelmiin. Käsiteltyjen menetelmien kirjavuus selittyy osin sillä, että tämän tutkimuksen tarpeisiin sellaisenaan soveltuvia menetelmiä ei juurikaan ollut saatavilla. Ensimmäisissä kohdissa 2.1 ja 2.2 tutustutaan kuitenkin aluksi aihepiirin peruskäsitteisiin. Niissä määritellään lyhyesti digitaalisen kuvan käsite sekä esitetään yleisimmät käytössä olevat värimallit.

Seuraavaksi kohdassa 2.3 esitellään ainakin näennäisesti lähtökohdiltaan tämän työn aihepiiristä eroava tutkimusala: sisältöpohjainen kuvatietokantahaku. Tämä tutkimuskenttä sisältää kuitenkin yhteisiä tavoitteita laadullisen analyysin kanssa. Kuvatietokantahaussa digitaalisista kuvista irrotetaan tavallisesti jokin piirre – tai piirrejoukko – jota myöhemmin käytetään tietokannan kuvien indeksoinnissa hakuprosessin aikana. Kantava ajatus tämän työn näkökulmasta onkin jatkossa soveltaa eräitä tässä kohdassa kuvattuja piirteitä kuvien laadulliseen analysointiin, toisin sanoen tutkia sitä, millä tavoin erilaisille painopapereille painetut kuvat eroavat toisistaan näiden kuvahaun piiristä lainattujen piirteiden osalta.

Tämän jälkeen kohdassa 2.4 on esitetty joitakin erilaisia ns. kuvien laatumittoja. Nimitys on hivenen harhaanjohtava, sillä tämä tutkimusala on lähinnä keskittynyt mittaamaan digitaalisten kuvien laatua sellaisessa tilanteessa, jossa kuvat ovat altistuneet jollekin niiden laatuun heikentävästi vaikuttavalle tekijälle; tyypillisesti esimerkiksi jonkin tiivistysmenetelmän käytöstä johtuville vääristymille. Näiden menetelmien perimmäisenä tarkoituksena onkin arvioida laskennallisesti sitä degradaation määrää, joka on ihmissilmin havaittavissa alkuperäisen kuvan ja jollekin vääristymälle altistuneen, muutoin oleellisesti saman kuvan välillä. Perusvaatimus tässä yhteydessä yleensä onkin, että sekä alkuperäinen että vääristynyt kuvasignaali on käytettävissä tarkastelua varten. Kuten edellä, myös laatumitoista saattaa olla eriävistä tavoitteistaan huolimatta merkittävää

hyötyä tämän tutkimuksen eteenpäin viemisessä. On kuitenkin samalla todettava, että laatumittojen luonteesta johtuen valtaosaa niistäkään ei voida soveltaa tässä työssä sellaisenaan.

Kohdassa 2.5 esitellään menetelmä, joka pyrkii kuvaamaan kuvissa esiintyviä paikallisia tekstuurikuvioita numeerisilla arvoilla. Myös tekstuurianalyysi eroaa tavoitteiltaan huomattavasti kuvien laadullisesta arvioinnista, mutta sen käyttö voi silti tässä yhteydessä olla perusteltua. Tekstuuripiirre ei vaadi referenssikuvaa ja niinpä sitä voidaan helposti soveltaa yksittäisille kuville. Tällöin on syytä tutkia myös sitä, saataisiinko tämän piirteen avulla mitattua erilaisten painopaperien käytön seurauksena syntyviä laadullisia eroja kuvien välillä.

Luvun lopuksi kohdassa 2.6 esitellään itseorganisoiva kartta, joka on moniulotteisen datan jäsentelyyn ja visualisointiin tarkoitettu menetelmä. Sitä käytetään laajalti hyväksi tutkimuksen kokeellisessa osassa. Monet luvussa 4 esitettävistä tuloksista hyödyntävätkin itseorganisoivan kartan kykyä visualisoida monimutkaisia piirrevektorien välisiä suhteita.

2.1 DIGITAALINEN KUVA

Kuva voidaan määritellä kaksiulotteisena funktiona $I(x,y)$, jossa x ja y ovat avaruudellisia tasokoordinaatteja. Harmaasävykuvien tapauksessa funktion I arvoa minkä tahansa koordinaattiparin (x,y) kohdalla kutsutaan kuvan intensiteetiksi (tai harmaasävyksi) kyseisessä pisteessä. Värikuvat puolestaan koostuvat tavallisesti useammasta kuin yhdestä tämän kaltaisesta komponenttikuvasta, toisin sanoen värikanavasta, jotka yhdessä määrittävät kussakin pisteessä vallitsevan värin ja kirkkauden [10].

Kuvaa kutsutaan digitaaliseksi kuvaksi silloin, kun x ja y samoin kuin myös $I(x,y)$ ovat äärellisiä, diskreettejä suureita. Termillä digitaalinen kuvankäsittely viitataan tilanteeseen, jossa digitaalisia kuvia prosessoidaan tietokoneen avulla. Digitaalinen kuva koostuu siis äärellisestä määrästä elementtejä, joista jokaisella on tietty sijainti ja arvo. Näitä elementtejä

kutsutaan tavallisesti kuvaelementeiksi eli pikseleiksi. Termeistä pikseli on useimmiten käytetty, kun halutaan viitata yhteen digitaalisen kuvan elementtiin [10].

Algoritmit, jotka analysoivat ja prosessoivat kuvia, yleensä käsittelevät ensisijaisesti pikseleitä. Synteettisiä kuvia generoivat algoritmit esimerkiksi tuottavat tulosteen joukon pikseliarvoja. Sovelluksissa pikselit voidaan ajatella hierarkkisen rakenteen alimmaiseksi tasoksi. Näin esimerkiksi digitaalisten kuvien analysointia voidaan ajatella sellaisena prosessina, joka alkaa matalalta tasolta ja joka korkeammalla tasolla pyrkii muodostamaan rakenteellisia riippuvuussuhteita pikselien välille [38].

Digitaalisessa kuvassa esiintyvien pikselien kokonaislukumäärää kutsutaan avaruudelliseksi resoluutioksi. Vastaavasti sitä tarkkuutta, jolla (värikuvan) pikselien eri intensiteettiä voidaan esittää, kutsutaan väriresoluutioksi. Tavallisesti kuvan resoluution ja visuaalisen laadun välillä on tiettyyn rajaan asti voimassa korrelaatio: mitä suurempi resoluutio, sitä parempi on myös kuvan laatu. Digitaalisen kuvankäsittelyn yhteydessä on myös tyypillistä, että niin väri- kuin avaruudellinenkin resoluutio määräytyvät käytettävissä olevan laitteiston vaatimuksia vastaaviksi [38].

2.2 VÄRIMALLIT

Digitaalista kuvankäsittelyä tarkasteltaessa on kuvattavan kohteen lisäksi tärkeää myös valon ominaispiirteiden tunteminen. Valoa kutsutaan akromaattiseksi silloin, kun sillä ei ole värisävyä. Tällöin sen ainoa ominaisuus on intensiteetti (kirkkaus). Tämän kaltaista valoa voi nähdä esimerkiksi mustavalkotelevisiossa. Valon värin määräävä ominaisuus on puolestaan kromaattisuus. Sen kuvaamiseen käytetään kolmea perussuuretta: radianssia, luminanssia ja kirkkautta. Radianssi on valonlähteestä säteilevä kokonaisenergia. Luminanssi puolestaan mittaa sitä energiamäärää, jonka havainnoija havaitsee. Esimerkiksi infrapuna-alueella säteilevällä valonlähteellä voi olla suuri kokonaisenergia (radianssi) mutta havaittu luminanssi siitä huolimatta lähes olematon. Lopuksi kirkkaus on

subjektiivinen piirre, jota on käytännössä mahdotonta mitata. Se ilmentää väri-intensiteetin akromaattista ominaisuutta ja se on yksi avaintekijöistä ihmisen värihavaintoa määriteltäessä [10].

Väriä käytetään usein apuna kuvankäsittelyssä muun muassa seuraavista kahdesta syystä. Ensinnäkin se on vahva piirre, joka usein helpottaa kuvissa esiintyvien objektien tunnistamista ja niiden erottamista taustastaan. Toiseksi ihminen kykenee erottamaan toisistaan tuhansia eri värisävyjä ja intensiteettejä. Eri harmaasävyjä ihminen pystyy sitä vastoin tunnistamaan vain muutamia kymmeniä [10].

Värimallin (tai *väriavaruuden*) tarkoitus on mahdollistaa värien määrittelemisen tietyllä, yleisesti hyväksytyllä ja standardilla tavalla. Pohjimmiltaan värimalli on määritelmä koordinaattijärjestelmälle ja sen aliavaruudelle, jossa tiettyä väriä edustaa yksi piste. Useimmat nykyään käytössä olevat värimallit ovat joko laitteistoa (esimerkiksi näyttöpäätteitä tai printtereitä) taikka sellaisia sovelluksia kohti suunnattuja, joissa värien manipulointi on tarpeen [10]. Digitaalisen kuvankäsittelyn piirissä laitteistosuuntautuneista malleista useimmin käytetään *RGB*-mallia (*red, green, blue*), jota suositaan mm. erilaisissa näyttölaitteissa sekä suuressa joukossa videokameroita. Lisäksi *CMY* (*cyan, magenta, yellow*) ja *CMYK* (*cyan, magenta, yellow, black*) -mallit esiintyvät mm. väritulostuksen ja erilaisten painotekniikoiden yhteydessä. *HSI*-malli (*hue, saturation, intensity*) vastaa edellä mainittuja paremmin sitä tapaa, millä ihminen ymmärtää ja tulkitsee värejä. Tässä värimallissa on lisäksi se etu, että se erottelee kuvan väri- ja harmaasävyinformaation toisistaan. Tällöin esimerkiksi harmaasävyjä hyväkseen käyttäviä tekniikoita voidaan soveltaa sellaisenaan. Monia erilaisia värimalleja on kehitetty, sillä väritutkimus on tänä päivänä laaja tutkimuskenttä, joka pitää sisällään suuren joukon sovellusalueita.

RGB-värimallissa kukin väri koostuu kolmesta primäärisestä spektrikomponentista: punaisesta, vihreästä ja sinisestä [10]. Siinä käytetään karteesisista koordinaattijärjestelmää, jolloin kunkin värin määrittämiseen tarvitaan arvokolmikko, joka siis vastaa

sijaintikoordinaatteja kolmiulotteisessa RGB-avaruudessa. RGB-aliavaruus on kuutio, jonka kahdeksasta kulmasta kolme edustaa ns. päävärejä, kolme ns. sekundäärivärejä ja loput kaksi edustavat valkoista ja mustaa. Harmaasävyt kulkevat viimemainittujen välillä kuution lävistäjää pitkin sen vastakkaisesta kulmasta toiseen.

Käytännössä RGB-arvot ovat usein tietokoneen muistissa 8-bittisiä kokonaislukuja, jolloin niiden arvot sijaitsevat välillä [0,255] matemaattisesti houkuttelevamman välin [0,1] sijasta. Näin ollen tässä ns. full-color järjestelmässä käytettävissä olevien värien lukumäärä on yhteensä noin 16 miljoonaa ($2^{8 \cdot 3}$). Koska nykyisissä näyttölaitteissa useimmiten suositaan RGB-mallia, suurin osa digitaalisista kuvista ja siten myös suuri osa kuvankäsittelyjärjestelmistä käyttää tätä mallia [3][10].

Edellä kuvattu RGB-malli on luonteeltaan additiivinen. Maksimimäärä kaikkia värikomponentteja tuottaa valkoisen värin. Sen sijaan CMY- ja CMYK -värimallit ovat luonteeltaan substraktiivisia. Niitä käytetään yleisesti sellaisissa järjestelmissä, joiden tarkoitus on tuottaa väripigmenttejä paperille. Kun painotuotteiden yhteydessä puhutaan neliväripainatuksesta, viitataan CMY-mallin kolmeen perusväriin mustalla täydennettynä. Jos oletetaan, että RGB-arvot on normalisoitu välille [0,1], voidaan muunnos RGB:n ja CMY:n välillä suorittaa seuraavalla yksinkertaisella kaavalla [10]:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

RGB-malli vastaa hyvin sitä fysikaalista tosiasiaa, että ihmissilmä havainnoi voimakkaasti punaista, vihreää ja sinistä primääriväriä [10]. Tästä huolimatta se ei sovellu erityisen hyvin värien käytännölliseen ja intuitiiviseen kuvailemiseen. Kun ihminen katsoo värillistä esinettä, sen ominaisuuksia ovat sävy, saturaatio ja kirkkaus. Sävy on piirre, joka ilmaisee puhdasta väriä (keltaista, oranssia, punaista jne.). Saturaatio taas kertoo kuinka paljon puhtaaseen väriin on sekoittunut valkoista. Lopuksi

kirkkaus on intensiteetin akromaattinen osa, joka tärkeä tekijä värihavaintoa määriteltäessä. HSI-värimalli erottaa värikuvien intensiteettikomponentin ja värisävyn sisältävät komponentit (sävy ja saturaatio) toisistaan tällä tavalla. Se onkin hyvä väline silloin, kun on tarkoitus kehittää sellaisia kuvankäsittelyalgoritmeja, joissa käsitellään värejä ihmiselle luontaisella tavalla.

Kuten edellä mainittiin, digitaalisen värikuvan pikseleiden väri-informaatio voidaan normaalisti esittää tiettyä 3-ulotteista väriavaruusmallia hyödyntäen. Useita värimalleja käytetäänkin tähän tarkoitukseen, joskaan ainuttakaan niistä ei ole yleisesti tunnustettu parhaimmaksi. Hyvältä värimallilta voidaan kuitenkin edellyttää ominaisuuksia, joita voivat olla täydellisyys, yhtenäisyys, tiiviys ja käyttäjäsuuntautuneisuus [30]. Täydellisyys tarkoittaa sitä, että malli mahdollistaa kaikkien ihmissilmin havaittavissa olevien värien esittämisen. Yhtenäisyydellä tarkoitetaan sitä, että mitattu 'läheisyys' kahden värin välillä on suoraan verrannollinen niiden havaitun samankaltaisuuden kanssa. Tiiviydellä tarkoitetaan sitä, että mallissa kukin yksittäinen väriarvo edustaa sen havaittavissa olevaa eroa muihin nähden. Käyttäjäsuuntautuneisuus puolestaan edellyttää, että kunkin värikomponentin vaikutus lopulliseen väriin on intuitiivinen ja ennalta arvattava. Toisin sanoen, värimallin tulisi mahdollistaa kaikkien ihmisen havainnoimien värien esittäminen siten, että väri-informaatio ja myös sen muutos korreloisi mahdollisimman hyvin ihmisen omien havaintojen kanssa.

2.3 SISÄLTÖPOHJAINEN KUVATIETOKANTAHAKU

Tiedon prosessointi- ja tallennuskapasiteetin jatkuva kasvu viime vuosikymmeninä on mahdollistanut entistä laajempien multimediatekijöiden synnyn. Kun digitaalisten kuvien tuottaminen on nykyään sekä halpaa että helppoa, niiden määrä eri medioissa on kasvanut voimakkaasti [26]. Kun kuva- ja videotietokantojen koko kasvaa suureksi, siitä seuraa myös niiden käytettävyyteen liittyviä ongelmia. Tärkein näistä on visuaalisen tiedon hakuongelma. Informaatiokokoelmien hyödyt jäävät

vähäisiksi, mikäli haluttua tietoa ei pystytä hakemaan riittävällä tarkkuudella ja riittävän nopeasti.

Koska pelkästään käyttäjän määrittämän tekstuaalisen metadatan – esimerkiksi tietueisiin liitettävien avainsanojen – avulla kuvahaku ei ole yleisessä tapauksessa riittävän tehokasta, on syntynyt tarve hakea tietoa myös kuvien visuaalisen sisällön perusteella. Tähän haasteeseen pyrkii vastaamaan sisältöpohjainen tietokantahaku, joka onkin tutkimusalana saanut viime aikoina runsaasti huomiota osakseen [8].

Seuraavassa esitellään joukko olemassa olevia kuvahakumenetelmiä, jotka perustuvat piirteiden irrottamiseen digitaalisista kuvista. Vaikka kuvahaun perimmäinen tavoite, tehokas ja tarkoituksenmukainen haku kuvatietokannasta ei sinänsä kuulu tämän tutkielman kenttään, näissä menetelmissä kuvista irrotetut piirteet saattavat sopia myös kuvien laadun määrittämiseen ja niiden keskinäiseen laadulliseen vertailuun.

2.3.1 Kuvahakujärjestelmät

Kuvahakuprosessissa kuva etsitään tietokannasta tavallisesti käyttäjän määrittämän kyselyn perusteella. Useimmat olemassa olevat kuvahakujärjestelmät tukevat yhtä tai useampaa seuraavista hakutavoista: satunnainen selaus, haku esimerkin perusteella, haku luonnoksen perusteella, haku avainsanan perusteella ja kategorisoitu navigointi [22][32].

Itse kuvahakuprosessi voidaan jakaa kolmeen erilliseen vaiheeseen: piirteiden irrottamiseen, moniulotteiseen indeksointiin sekä hakujärjestelmän suunnitteluun [32]. Jatkossa keskitytään tarkastelemaan näistä lähinnä kahta ensin mainittua, sillä ne liittyvät tiiviimmin tämän työn aihepiiriin.

Valmiita tai jatkokehityksen kohteena olevia kuvahakujärjestelmiä on olemassa useita erilaisia. Niitä on suunniteltu sekä kaupallisiin tarkoituksiin että myös tutkimuskäyttöön. Olemassa olevista kuvahakujärjestelmistä esimerkkeinä mainittakoon IBM:n *QBIC*-järjestelmä

[8], Kolumbian yliopiston kehittämä *VisualSeek* [33] sekä Helsingin teknillisen korkeakoulun itseorganisoivan kartan käyttöön pohjautuva *PicSOM* [20].

2.3.2 Peruskäsitteitä

Väri on tärkein yksittäinen matalan tason piirre, jota käytetään hyväksi kuvahaussa. Tämä ei sinänsä ole kovinkaan yllättävää, sillä esimerkiksi ihmisen visuaalinen hahmotuskyky ja muistaminen ovat pitkälti riippuvaisia väristä [2].

Värin ohella toinen yleisesti hahmontunnistuksessa ja myös kuvahaussa käytetty piirre on tekstuuri. Tekstuurilla tarkoitetaan pintakuviointia, jossa joukko primitiivisiä elementtejä esiintyy tietyllä kuvan alueella tasaisesti jakautuneena. Tekstuurielementti on yleensä yksinkertaisen muotoinen yhtenäisen intensiteetin omaava alue, joka toistuu ympäristössään. Tekstuuria voidaan analysoida pikseli-ikkuna -tasolla (tilastollinen analyysi) tai tekstuurielementtitasolla (rakenneanalyysi). Jälkimmäinen lähestymistapa sopii tilanteeseen, jossa tekstuurielementit voidaan selkeästi erotella toisistaan, kun sen sijaan tilastollista analyysiä voidaan käyttää hienojakoisemman kuvioinnin tulkitsemisessa [3].

Tilastollisilla mitoilla voidaan mitata sellaisia tekstuurin piirteitä kuten kontrastia, karkeutta ja kuvioinnin suuntaa. Myös *Fourier*-spektrejä käytetään tekstuurien kuvaamiseen. Suorittamalla Fourier-muunnos tekstuuri-ikkunalle saadaan ikkunalle tunnusarvo. Saman tai samankaltaisen tunnusarvon omaavat ikkunat voidaan tämän jälkeen tulkita samaan tekstuurialueeseen kuuluviksi [3].

Rakenneanalyysissä pyritään tunnistamaan kuvasta tekstuurielementit, määrittämään niiden muoto sekä sijoittelusäännöt. Sijoittelusäännöt kertovat, kuinka elementit sijaitsevat kuvassa toisiinsa nähden. Rakenneanalyysissä mitattaviin asioihin voivat kuulua esimerkiksi elementtien naapurien lukumäärä, esiintymistiheys ja säännöllisyys [3].

Yleisesti yksi vaikeimmista kuvahaun osa-alueista on muoto- ja hahmopohjainen haku. Tämä johtuu kuvissa esiintyvien kohteiden segmentoinnin vaikeudesta. Muotojen löytäminen kuvasta edellyttää yleensä kuvan esikäsittelyä, joka vaihtelee sovellusalueesta riippuen. Kun haluttu kohde on paikannettu, sen reuna voidaan etsiä reunanetsintäalgoritmeilla. Tämän jälkeen kohteen muotoa voidaan kuvata esimerkiksi sen kaltaisilla mitoilla kuin eksentrisyys, pyöreys, reunan kaarevuus, etäisyys keskipisteeseen, muodon momentti, reunan fraktaalidimensio jne. Kaikkia edellä mainittuja voidaan kuvata numeerisilla arvoilla [3].

Jotta haku kuvatietokannasta olisi mahdollista, tietokantaan tallennetaan yleensä yksi ominaisuusvektori kuvaa kohden indeksointia varten [34]. Haku tapahtuu soveltamalla tiettyä *etäisyysmetriikkaa* tässä ominaisuusavaruudessa. Kun käytettävä etäisyysmetriikka on valittu, sen avulla voidaan laskea etäisyydet tarkasteltavasta kuvasta irrotetun ominaisuusvektorin ja kunkin tietokannan kuvan ominaisuusvektorin välillä. Näitä etäisyyksiä verrataan sellaiseen kynnyksarvoon, jonka perusteella palautuvat hakutulokset lopulta määräytyvät.

Ominaisuusvektoreita voidaan muodostaa kuvista monella tavalla. Kuvan visuaalista sisältöä voidaan ajatella esimerkiksi abstraktioiden hierarkiana [3]. Alimmalla abstraktiotasolla ovat pikselien intensiteettiarvot, jotka siis sisältävät kirkkaus- ja väri-informaation. Jatkokäsittelyn avulla niiden perusteella voidaan tuottaa erilaisia piirteitä, kuten reunoja, kulmia, viivoja, käyriä ja värialueita. Korkeammalla abstraktiotasolla näitä piirteitä voidaan edelleen yhdistellä ja tulkita kokonaisuutena. Lisäksi voidaan mahdollisesti pyrkiä muodostamaan niistä kuvassa esiintyviä objekteja ja näiden attribuutteja. Kaikkein korkeimmalla abstraktiotasolla ovat ihmiselle luonnolliset käsitteet objekteista ja näiden välisistä vuorovaikutussuhteista, esimerkiksi "ihminen pitämässä puhetta". Yleisessä tapauksessa korkean tason ominaisuuksien tulkitseminen kuvasta automaattisesti ja luotettavasti ei kuitenkaan ole nykyisen tietämyksen nojalla mahdollista.

2.3.3 Etäisyysmitat

Piirteiden irrotusprosessissa saadaan tuloksena yleensä yksi n -ulotteinen ominaisuusvektori kuvaa kohti. Kuten edellä mainittu, näitä ominaisuusvektoreita voidaan kuvahakujärjestelmässä käyttää kuvien indeksoimiseen. Tällöin hakutulosten muodostamista varten on pystyttävä määrittämään kahden ominaisuusvektorin välinen erilaisuus, ts. niiden keskinäinen 'etäisyys' toisistaan [23][39].

Tarkasteltavien ominaisuusvektorien komponenttien ei välttämättä tarvitse olla metrisen avaruuden alkioita. Mikäli ne ovat, niiden välistä etäisyyttä voidaan mitata ns. vektoriavaruusmallia käyttäen [39]. Tässä etäisyyden määrittämiseen käytetään mitä tahansa tarkoitukseen sopivaa n -ulotteisessa avaruudessa määriteltyä metriikkaa. Kuvahaussa yleisimmin käytetty ominaisuusvektorien välinen etäisyysmetriikka on L_2 , Euklidinen normi, mutta myös muita L_P -perheen metriikoita, kuten L_1 , Manhattan normi, voidaan käyttää. Yleistetty vektorien välinen *Minkowski*-metriikka L_P on seuraava [1]:

$$L_P(\vec{x}, \vec{y}) = \left(\sum_i |x_i - y_i|^P \right)^{1/P}, P \geq 1 \quad (2)$$

Alla on lisäksi esitetty kaksi toisenlaista vektorien etäisyysmetriikkaa, *Canberra*-etäisyys (3) ja *Czekanowski-kerroin* (4) [2]. Molempia voidaan käyttää ei-negatiivisille moniulotteisille vektoreille, kuten esimerkiksi RGB-väriavaruuden pisteille (joita [2] käyttää kuvahaun ominaisuusvektoreina):

$$E_C(\vec{x}, \vec{y}) = \sum_i \frac{|x_i - y_i|}{|x_i + y_i|} \quad (3)$$

$$E_Z(\vec{x}, \vec{y}) = 1 - \frac{2 * \sum_i \min(x_i, y_i)}{\sum_i (x_i + y_i)} \quad (4)$$

Androustos (et al.) [2] tutkivat etäisyysfunktion vaikutusta kuvahaun onnistumiseen. Heidän mukaansa tilanteessa, jossa hakuavaimina käytetään RGB-väriavaruuden vektoreita, parhaimmat tulokset ihmisen värihavaintoon nähden saadaan käyttämällä piirrevektorien kulmien erotukseen pohjautuvia metriikoita. Seuraavassa on esitetty kaksi tämän kaltaista metriikkaa (5) ja (6). Näistä jälkimmäinen on erityisesti kuvahakua varten kehitetty metriikka, joka yhdistää vektorien välisen kulman ja näiden välisen etäisyyden. Kun mitattavat vektorit ovat yhdensuuntaiset, eron määrää ainoastaan niiden pituuden erotus.

$$\mathbf{q} = 1 - \frac{2}{\mathbf{p}} \cos^{-1} \left(\frac{\vec{x} \cdot \vec{y}}{|\vec{x}| |\vec{y}|} \right) \quad (5)$$

$$E_N(\vec{x}, \vec{y}) = 1 - \underbrace{\left[1 - \frac{2}{\mathbf{p}} \cos^{-1} \left(\frac{\vec{x} \cdot \vec{y}}{|\vec{x}| |\vec{y}|} \right) \right]}_{\text{kulmatekijä}} \underbrace{\left[1 - \frac{|\vec{x} - \vec{y}|}{\sqrt{3 * 255^2}} \right]}_{\text{pituustekijä}} \quad (6)$$

Yllä olevissa kaavoissa kulma normalisoidaan kertoimella $\frac{2}{\mathbf{p}}$, sillä suurin mahdollinen kulma on $\frac{\mathbf{p}}{2}$. Lisäksi jälkimmäisessä kaavassa käytetty normalisointikerroin $\frac{1}{\sqrt{3 * 255^2}}$ voidaan johtaa siitä, että käytännön sovelluksissa pisin vektorien välinen etäisyys RGB-avaruudessa on tavallisesti (255,255,255), jonka pituus siis on $\sqrt{3 * 255^2}$. Jälkimmäisessä E_N -metriikassa on lisäksi vielä se etu, että tarvittaessa kulma- ja pituuskomponenteille voidaan asettaa toisistaan eriävät painokertoimet mikäli toisen merkitystä halutaan korostaa.

Normaalissa tapauksessa ominaisuusavaruuden ulotteisuus n on kaikkien käsiteltävien vektorien suhteen vakio. Yleisessä tapauksessa näin ei kuitenkaan välttämättä tarvitse olla. Tällöin voidaan käyttää vektoriavaruusmallia rajoittuneempaa ns. metrisen avaruuden mallia [39]. Tässä tapauksessa ominaisuusavaruuteen voidaan määritellä jokin etäisyysfunktio D siten, että sille pätee

- (i) positiivisuus: $D(\vec{x}, \vec{y}) \geq 0$, ja $D(\vec{x}, \vec{x}) = 0$,
- (ii) symmetrisyys: $D(\vec{x}, \vec{y}) = D(\vec{y}, \vec{x})$,
- (iii) kolmioepäyhtälö: $D(\vec{x}, \vec{z}) \leq D(\vec{x}, \vec{y}) + D(\vec{y}, \vec{z})$.

Kuvahaun yhteydessä puhutaan myös usein yleisemmin erilaisuusfunktioista, jotka eivät välttämättä täytä kaikkia edellä mainittuja matemaattiselle metriikalle asetettuja vaatimuksia, kuten esimerkiksi kolmioepäyhtälö [4].

2.3.4 Värihistogrammit

Digitaalisista kuvista voidaan suhteellisen helposti irrottaa erilaisia matalan tason piirteitä, kuten värisävy, kirkkaus, reunaisuus tai esimerkiksi värisävyn tai kirkkauden yhteisesiintyminen [4]. Näistä matalan tason piirteistä voidaan puolestaan muodostaa histogrammeja, tiheysjakaumia, joita voidaan käyttää esimerkiksi sellaisenaan ominaisuusvektoreina kuvien indeksoinnissa ja kuvahaussa.

Perinteisesti kuvahaussa onkin käytetty runsaasti värihistogrammeja [24]. Värihistogrammilla tarkoitetaan siis kuvan värien globaalista esiintymistiheydestä muodostettua vektorimuotoista esitystä. Värihistogrammin käsite voidaan formuloida esimerkiksi seuraavalla tavalla käyttäen apuna todennäköisyysfunktiota Pr . Kuvan I värihistogrammi määritellään siinä esiintyville väreille c_i seuraavasti [11][26]:

$$h_{c_i}(I) = XY \cdot \Pr[p \in I_{c_i}] \quad (7)$$

Yllä I siis vastaa kuvaa, jonka dimensiot ovat $X*Y$. Kuva koostuu pikseleistä p ja väreistä c_1, c_2, \dots, c_n . Mille tahansa kuvan pikselille $\frac{h_{c_i}(I)}{XY}$ antaa todennäköisyyden sille, että kyseessä oleva pikseli edustaa väriä c_i .

Tilastollisessa mielessä digitaalisen kuvan voidaan ajatella rakentuvan kahdesta komponentista: paikka-avaruudesta ja väriavaruudesta [4]. Paikka-avaruus koostuu kaikista mahdollisista pikselien sijainneista ja väriavaruus vastaavasti kaikista mahdollisista käytettävissä olevista väreistä. Jotta värihistogrammeja voitaisiin käyttää, on molemmat avaruudet aluksi kvantisoitava sopivaan tarkkuuteen.

Kvantisointiprosessissa kuvan väri-informaatio muunnetaan diskreettiin väriavaruuteen, jossa on yhteensä n väriä. Värihistogrammi on tällöin n -ulotteinen vektori $(h_{c_1}, h_{c_2}, \dots, h_{c_n})$, jossa kukin elementti h_{c_i} vastaa värin c_i esiintymien lukumäärää kuvassa. Sovellusalueen kaikkien mahdollisten histogrammien joukkoa kutsutaan histogrammiavaruudeksi, joka on n -ulotteisen vektoriavaruuden aliavaruus. Jos oletetaan yksinkertaisuuden vuoksi, että pikselien lukumäärä sovellusalueen jokaisessa kuvassa on $X*Y$, voidaan histogrammiavaruuden käsite formuloida seuraavasti [34]:

$$H = \left\{ (h_{c_1}, h_{c_2}, \dots, h_{c_n}) \mid h_{c_i} \geq 0, (1 \leq i \leq n), \sum_{i=1}^n h_{c_i} = XY \right\} \quad (8)$$

Kuvien keskinäistä vertailua varten niiden histogrammien välistä etäisyyttä määrittämään valitaan jokin etäisyysmetriikka. Kun kahden histogrammin välinen etäisyys t on tiedossa, sen avulla voidaan määritellä, että kyseiset histogrammit ovat t -samanlaiset, mikäli näiden välinen etäisyys on t tai vähemmän [34]. Vastaavasti histogrammien voidaan sanoa olevan t -erilaiset, mikäli näiden välinen etäisyys on enemmän kuin t . Näin ollen histogrammipohjaisen kuvahaun näkökulmasta tietokannassa oleva kuva voidaan palauttaa hakutuloksena, mikäli se on t -samanlainen kyselykuvan kanssa. Tässä t on siis haussa käytettävä kynnsarvo, jonka avulla voidaan vaikuttaa palautuvien tulosten lukumäärään. Kynnsarvon valintaan on syytä kiinnittää huomiota, sillä liian suuri arvo voi näkyä suurena määränä ei-toivottuja hakutuloksia ja vastaavasti liian pieni arvo voi jättää haluttuja kuvia tulosjoukon ulkopuolelle.

Kuvien indeksointi kuvahaussa histogrammien avulla toimii tehokkaasti ainoastaan siinä tapauksessa, että sovellusalueen kuvatietokannan histogrammit sijaitsevat histogrammiavaruudessa harvassa [34]. Tätä ominaisuutta voidaan tutkia ns. histogrammikapasiteetin avulla. Käsitteellä tarkoitetaan sitä suurinta lukumäärää t -erilaisia histogrammeja, jotka mahtuvat n -ulotteiseen histogrammiavaruuteen silloin, kun käytössä on tietty etäisyysmitta D . Tällä tavoin voidaan arvioida histogrammivertailun teoreettista erottelukykä, eli toisin sanoen histogrammipiirteiden kykyä erotella tietokannan kuvia toisistaan.

Kuvassa esiintyvät värit luokitellaan kvantisoinnin yhteydessä luokkiin. Luokan koko voi olla joko vakio, tai se voi myös vaihdella luokasta riippuen [4]. Tällä tavoin muodostuva histogrammiresoluutio vaikuttaa tietokantaan tallennettavien ominaisuusvektorien tilan tarpeeseen sekä myös laskennan kompleksisuuteen. Mitä pienempiä ominaisuusvektorit ovat, sitä nopeampi niitä on käsitellä. Niinpä hakuprosessin kannalta histogrammiresoluution tulisi olla niin matala kuin mahdollista ilman, että hakutulokset heikkenevät merkittävästi.

Mikäli histogrammiresoluutio ei ole optimaalinen, tuloksena saavutettu kvantisointi voi olla joko yli- tai alitarkka kuvaus kyseessä olevasta hajonnasta. Eräs tapa arvioida optimaalista histogrammin luokkien vakiokokoa s_h on käyttää esimerkiksi alla olevaa Scottin kaavaa [4]:

$$s_h = 3.5sN^{-1/3} \quad (9)$$

Tässä N vastaa näytteen kokoa ja s on arvio sen keskihajonnasta.

Värihistogrammien käyttö kuvahaussa on varsin yleistä, sillä ne täyttävät kohtuullisen hyvin vaatimukset, joita kuvista irrotettaville piirteille voidaan tässä yhteydessä asettaa. Tärkein yksittäinen seikka, joka tukee histogrammien käyttöä tähän tarkoitukseen on se, että niiden muodostaminen ja käsitteleminen ei vaadi suurta laskennallista tehoa. On arvioitu, että nykypäivänä jo vaatimatonkin laitteisto kykenee

värihistogrammien avulla vertailemaan keskenään yli miljoonaa kuvaa sekunnissa [26].

Värihistogrammeilla on kuitenkin myös puutteensa [11][24][29]. Ne kuvaavat ainoastaan värien globaalia jakaumaa, jolloin sijainti-informaatio jää kokonaan vaille huomiota. Kahdella visuaalisesti hyvinkin erinäköisellä kuvalla voi olla lähes identtinen histogrammi, mikäli ne sisältävät samankaltaisia värejä ainoastaan eri tavoin ryhmittyneenä. Tämä onkin ongelma erityisesti silloin, kun sovellusalue käsittää lukumäärällisesti paljon erilaisia kuvia. Tällöin virheellisten hakutulosten todennäköisyys kasvaa huomattavasti. Lisäksi esimerkiksi kuvakulman vaihtaminen voi aiheuttaa merkittäviä muutoksia kuvan histogrammiin, vaikka kuva itse edelleen esittäisikin samaa kohdetta.

Edellä kuvattuun ongelmaan eräs ratkaisu on esimerkiksi segmentoida kuva tasaisesti samankokoisiin osakuviin (ikkunoihin) ja käyttää ominaisuusvektorina useampaa, kustakin ikkunasta erikseen muodostettua histogrammia [11]. Tätä lähestymistapaa on käytetty mm. tämän työn kokeellisessa osassa.

2.3.5 Värikorrelogrammit

Värikorrelogrammeja on esitetty paikkaamaan histogrammien puutteita. Värikorrelogrammi tallentaa värien spatiaalisen informaation ja sitä voidaan käyttää kuvaamaan globaalia jakaumaa värien paikallisten sijaintien korrelaatiosta. Lisäksi korrelogrammin kehittäjien (Huang et al.) [11] mukaan korrelogrammi on myös helppo laskea ja sen tilan tarve on suhteellisen pieni. Se onkin eri tutkimuksissa osoittautunut histogrammeja ja ns. värikoherenssivektoreita (ks. kohta 2.3.6) tehokkaammaksi menetelmäksi kuvien indeksointiin ja tietokantahakuun [11][21][26].

Käytännön sovelluksissa kuvan värikorrelogrammi on taulukko, jota indeksoidaan väripareilla (c_i, c_j) siten, että t_s alkio rivillä (i, j) vastaa sitä todennäköisyyttä, jolla c_i - ja c_j väriset pikselit esiintyvät kuvassa

etäisyydellä t toisistaan. Korrelogrammi voidaan määritellä histogrammin määritelmää (7) vastaavalla notaatiolla seuraavasti [11][26]:

$$\mathbf{g}_{c_i, c_j}^{(t)}(I) = \Pr_{p_1 \in I_{c_i}, p_2 \in I_{c_j}} [p_2 \in I_{c_j} \mid |p_1 - p_2| = t] \quad (10)$$

Toisin sanoen, $\mathbf{g}_{c_i, c_j}^{(t)}$ vastaa sitä todennäköisyyttä, että pikseli tietyn etäisyyden t päässä mistä tahansa väriä c_i ilmentävästä pikselistä kuvassa I ilmentää väriä c_j .

Pikselien etäisyyttä korrelogrammissa mitataan yksinkertaisuuden vuoksi L_∞ -normilla, jolloin pikseleiden $p_1 = (x_1, y_1)$ ja $p_2 = (x_2, y_2)$ etäisyys saadaan kaavalla $|p_1 - p_2| = \max(|x_1 - x_2|, |y_1 - y_2|)$.

Korrelogrammin tilavaatimus on $n^2 \cdot k$, jossa n vastaa värien määrää kuvassa ja k puolestaan vastaa kaikkien mahdollisten pikselien etäisyyksien joukon (t_1, t_2, \dots, t_k) alkioden lukumäärää [26]. Mikäli pikselien etäisyyksiä ei rajoiteta eikä enempää kvantisoida, on k tällöin yhtä kuin $\max(X, Y)$, jossa X ja Y on kuvan leveys ja korkeus pikseleinä.

Monien käytännön sovellusten tarpeisiin riittää kuitenkin yleensä suppeampi korrelogrammin osajoukko, *autokorrelogrammi*, joka on tila- ja laskentavaativuudeltaan huomattavasti korrelogrammia kevyempi. Kuvan I autokorrelogrammi \mathbf{a} antaa todennäköisyyden sille, että kaksi *identtistä* väriä c löydetään kuvasta etäisyydellä t toisistaan [26]:

$$\mathbf{a}_c^{(t)}(I) \equiv \mathbf{g}_{c,c}^{(t)}(I) \quad (11)$$

Autokorrelogrammilla saavutetaan merkittäviä laskennallisia etuja yleiseen korrelogrammiin verrattuna. Ensinnäkin sen tilan tarve on enää $n \cdot \max(X, Y)$, joka on olennaisesti vähemmän kuin yleisessä korrelogrammissa. Toiseksi tilastollinen keskimääräinen 'osumien' lukumäärä yhtä histogrammin solua kohti kasvaa. Autokorrelogrammissa tämä arvo on $\frac{XY}{n \cdot k}$, kun taas

korrelogrammissa vastaava arvo voi olla huomattavasti pienempi, $\frac{XY}{n^2 \cdot k}$.

Näin korrelogrammien tilastollinen luotettavuus paranee, joka myös helpottaa niiden keskinäistä vertailua. Kirjallisuudessa yleensä edellytetään histogrammin solun sisältävän keskimäärin ainakin 5-10 osumaa, jotta histogrammi olisi tilastollisesti luotettava [26].

Valmiita korrelogrammeja voidaan vertailla samaan tapaan kuin histogrammejakin. Näin ollen siis edellä mainitut histogrammien etäisyysmetriikat, mukaan lukien Euklidinen normi, ovat suoraan sovellettavissa myös korrelogrammeihin.

2.3.6 Muita tekniikoita

Värikoherenssivektori on jaettu värihistogrammi, jossa histogrammin elementit jakautuvat vielä kahteen aliluokkaan pikselien avaruudellisen koherenssin perusteella [5][21][29]. Pikselin katsotaan olevan koherentti siinä tapauksessa, että se kuuluu tiettyä kynnyсарvoa suurempaan yhtenäiseen alueeseen. Muussa tapauksessa pikselin katsotaan olevan epäkoherentti. Näin saadaan histogrammi, joka on kaksi kertaa perinteistä värihistogrammia isompi, mutta tallentaa sisäänsä tietoa myös pikselien keskinäisistä suhteista kuvassa.

Cinquen (et al.) [5] kehittämä menetelmä, spatiaalis-kromaattinen histogrammi, pyrkii laajentamaan tätä ajatusta. Tämä piirre pyrkii tuottamaan vastauksen seuraaviin kolmeen kysymykseen: kuinka monta merkitsevää väriä kuvassa on, missä toisiaan muistuttavat pikselit kuvassa likimäärin sijaitsevat ja miten nämä pikselit ovat avaruudellisesti jakautuneet.

Piirteen laskennan aluksi kuvan I värit redusoidaan jollakin kvantisointimenetelmällä kokoelmaan, joka käsittää yhteensä n väriä. Spatiaalis-kromaattinen histogrammi on vektori, jossa on n elementtiä. Jokainen elementti $j = \{1..n\}$ sisältää toisaalta väriä c_j ilmentävien pikselien

lukumäärän kuvassa mutta myös tietoa kyseisten pikselien spatiaalisesta rakenteesta.

Määritellään joukko $P_c(I) = \{(x, y) \in I \mid I(x, y) = c\}$ sellaisten pikselien joukoksi kuvassa I , jotka edustavat väriä c . Sen avulla voidaan määritellä h_c väriä c edustavien pikselien suhteelliseksi lukumääräksi kuvassa I (jonka dimensiot ovat $X \times Y$) seuraavasti [5]:

$$h_c(I) = \frac{|P_c(I)|}{XY} \quad (12)$$

Painopiste \mathbf{r}_c on spatiaalis-kromaattisen histogrammin komponentti ja se on määritelty pikseleille joukossa P_c seuraavalla tavalla [5]:

$$\mathbf{r}_c(I) = \left(\frac{1}{X} \frac{1}{|P_c(I)|} \left[\sum_{(x,y) \in P_c(I)} x \right], \frac{1}{Y} \frac{1}{|P_c(I)|} \left[\sum_{(x,y) \in P_c(I)} y \right] \right) \quad (13)$$

Painopisteen avulla saadaan viitettä samanväristen pikselien sijainnista. Se on hyödyllinen väline kuvaamaan hyvin määriteltyjä, yhtenäisiä värialueita. Yksinään se on kuitenkin vielä varsin karkea kuvaus pikselien spatiaalisesta järjestyksestä koko kuvan alueella. Tämän vuoksi spatiaalis-kromaattista histogrammia varten lasketaan lisäksi vielä pikselien keskihajonta \mathbf{s}_c , joka kuvaa pikselien hajontaa painopisteen ympärillä [5]:

$$\mathbf{s}_c(I) = \sqrt{\frac{1}{|P_c(I)|} \sum_{p \in P_c(I)} L_2(p, \mathbf{r}_c(I))^2} \quad (14)$$

Yllä p tarkoittaa joukkoon P_c kuuluvaa pikseliä, jonka suhteelliset koordinaatit ovat (x_p, y_p) . Lisäksi L_2 on kahden pikselin paikkakoordinaattien välinen Euklidinen etäisyys.

Spatiaalis-kromaattinen histogrammi C voidaan nyt edellisten perusteella määritellä n -alkioiseksi joukoksi seuraavasti [5]:

$$C_{c_i}(I) = (h_{c_i}(I), \mathbf{r}_{c_i}(I), \mathbf{s}_{c_i}(I)), i = \{1..n\} \quad (15)$$

Valmis piirrejoukko on pieni kooltaan ja se voidaan laskea nopeasti. Koska sen laskennassa käytetyt piirteet normalisoidaan, se on myös invariantti skaalan vaihdoksille. Piirteen kehittäjät ehdottavat kahden spatiaalis-kromaattisen histogrammin välisen etäisyyden määrittämiseen seuraavanlaista etäisyysmetriikkaa [5]:

$$E_S(I, Q) = \sum_{i=1}^n \min(h_{c_i}(I), h_{c_i}(Q)) \cdot \left(\frac{\sqrt{2} - L_2(\mathbf{r}_{c_i}(I), \mathbf{r}_{c_i}(Q))}{\sqrt{2}} + \frac{\min(\mathbf{s}_{c_i}(I), \mathbf{s}_{c_i}(Q))}{\max(\mathbf{s}_{c_i}(I), \mathbf{s}_{c_i}(Q))} \right) \quad (16)$$

Yllä I on kyselyssä käytettävä kuva, lisäksi Q on toinen, tietokannassa esiintyvä kuva. Kuvien spatiaalis-kromaattiset histogrammit $C(I) = (h(I), \mathbf{r}(I), \mathbf{s}(I))$ ja $C(Q) = (h(Q), \mathbf{r}(Q), \mathbf{s}(Q))$ oletetaan tunnetuiksi.

Merkittävin ero perinteisten etäisyysfunktioiden ja yllä esitetyn välillä on siinä, että väri- ja avaruudellinen informaatio käsitellään tässä erillään toisistaan. Painofunktio vahvistaa avaruudellista komponenttia sellaisessa tilanteessa, jossa kuvassa esiintyy suuri määrä yhtäläisen värisiä pikseleitä. Komponentin arvo kasvaa tilanteessa, jossa painopisteet sijaitsevat lähellä toisiaan.

Spatiaalis-kromaattisen histogrammin kehittäjät ovat kokeilleet sitä käytännön kuvahaussa ja myös saavuttaneet rohkaisevia tuloksia.

Värien sijaintikuvaaja [13] on hieman toisenlainen menetelmä, joka myös pyrkii esittämään värien avaruudellista (spatiaalista) jakaumaa kuvassa. Se on kehitetty osaksi MPEG-7 -standardia, joka pyrkii luomaan yhtenäisen välineistön multimedian sisällön kuvaamiseen. MPEG-7 -standardi sisältääkin mm. tehokkaita kuvaustekniikoita, jotka on suunniteltu juuri samankaltaisuuden perusteella tapahtuvaan kuvahakuun [22].

Kasutanin ja Yamadan kehittämän värien sijaintikuvaajan [13] ominaisuuksiin kuuluu mm. erittäin vaatimaton tilantarve. Valmiin ominaisuusvektorin koon ei nimittäin tarvitse olla enempää kuin 63 bittiä (8

tavua). Sen avulla onkin mahdollista suorittaa kuvahaku erittäin nopeasti ja lisäksi myös suhteellisen tarkasti. Sijaintikuvaajan pienen koon ansiosta sillä voidaan sen kehittäjien mukaan käydä läpi jopa 24 tuntia videokuvaa sekuntia lyhyemmässä ajassa.

Itse sijaintikuvaajan laskentaprosessi voidaan jakaa seuraaviin neljään vaiheeseen. Ensimmäiseksi kuva ositetaan sen koosta riippumatta 64 samankokoiseen alueeseen. Kunkin alueen koko on tällöin siis $(X/8, Y/8)$, jossa X ja Y vastaavat alkuperäisen kuvan leveyttä ja korkeutta pikseleinä. Osituksen jälkeen jokaisesta alueesta valitaan yksi ns. dominoiva väri edustamaan kyseistä aluetta. Tähän tehtävään voidaan soveltaa mitä tahansa tunnetuista menetelmistä dominoivan värin löytämiseksi. Kasutani ja Yamada käyttävät työssään yksinkertaista väriarvojen keskiarvolaskua. Näin saadaan pieni, 8×8 kokoinen kuva, jossa esiintyy alkuperäisen kuvan tärkeimmät värit alkuperäistä kuvaa vastaavalla tavalla sijoiteltuina. Tämän jälkeen kaikkiin kolmeen komponenttikuvaan sovelletaan 8×8 diskreetti kosinimuunnos, jonka tuloksena saadaan kolme DCT-kertoimien joukkoa. Laskennan päätteeksi tästä kerroinjoukoista irrotetaan muutama matalataajuuksinen kerroin zigzag-skannauksen osoittamassa järjestyksessä. Kertoimet vielä kvantisoidaan haluttuun tarkkuuteen ja lopuksi yhdistetään nämä piirteet muodostamaan valmis sijaintikuvaaja [13].

Sijaintikuvaajan laskennassa alkuperäisen kuvan oletetaan olevan $YCbCr$ -väriavaruudessa. Kun pienennetylle 8×8 kuvalle suoritetaan edellä kuvatun prosessin mukaisesti diskreetti kosinimuunnos, saadaan tuloksena yksi DC-kerroin ja 63 AC-kerrointa kullekin kolmelle komponenttikuvulle. Kokeellisten tulosten perusteella koko-optimaaliseen sijaintikuvaajaan sisällytetään kuusi ensimmäistä luminanssikanavan (Y) kerrointa ja vastaavasti kolme ensimmäistä kummankin krominanssikanavan (Cb ja Cr) kerrointa. Kertoimien kvantisointi suoritetaan siten, että DC-kerroin saa kuusi bittiä ja AC-kertoimet viisi bittiä kukin. Näin ollen valmiin värien sijaintikuvaajan koko on ainoastaan $(6+5 \times 2) \times 2 + (6+5 \times 5) = 63$ bittiä [13].

Kahden kuvan välisen eron – tai kuvan ja luonnoksen välisen eron – määrittäminen voidaan suorittaa laskemalla etäisyysfunktion arvo kahden sijaintikuvaajan välillä. Piirteen kehittäjät ehdottavat, että tähän tarkoitukseen voidaan käyttää esimerkiksi painotettua L_2 -normia seuraavaan tapaan [13]:

$$E_M(I, Q) = \sqrt{\sum_{i \in (Y)} w1_i (Y_i(I) - Y_i(Q))^2} + \sqrt{\sum_{i \in (Cb)} w2_i (Cb_i(I) - Cb_i(Q))^2} + \sqrt{\sum_{i \in (Cr)} w3_i (Cr_i(I) - Cr_i(Q))^2} \quad (17)$$

Edellä merkinnät $Y_i(I)$, $Cb_i(I)$ ja $Cr_i(I)$ tarkoittavat i :nnettä DCT-kerrointa komponenteille Y, Cb ja Cr kuvassa I. Lisäksi $w1_i$, $w2_i$ ja $w3_i$ edustavat parametreina olevia, kullekin tekijälle ennalta määrättyjä painokertoimia. Painokertoimien arvojen tulisi olla laskevia zigzag skannauksen järjestyksen mukaisesti. Painokertoimina myös suositellaan käytettävän 2:n potensseja, jolloin kehittäjien mukaan voidaan saavuttaa laskentatehon kasvua [13].

2.4 KUVIEN LAATUMITAT

Kirjallisuudessa esiintyy runsaasti sellaisia digitaalisten kuvien laadullisen arvioinnin menetelmiä, jotka on tarkoitettu jollekin vääristymälle altistuneen kuvan ja alkuperäisen kuvan välisen degradaation mittaamiseen [9][12][19][35][37]. Pääasiallinen sovelluskohde näille menetelmille löytyy kuvien tiivistysmenetelmien tutkimuksen piiristä, jossa erilaisten tiivistysmenetelmien suoriutumista pyritään arvioimaan käyttäen näitä laatumittoja. Tässä luvussa esitellään muutamia tämän kaltaisia laatumittoja.

Kuvankäsittelyn piirissä eräs merkittävimmistä tutkimuskohteista on perinteisesti ollut kuvien tiivistäminen. Vaikka tiedon tallennus- ja siirtokapasiteetit ovatkin jo pitkään olleet kasvussa, on myös kuvien määrällinen tarve sekä niiltä vaadittava tarkkuus (ts. pikseli- ja väriresoluutio) jatkuvasti kasvanut [28]. Digitaaliset kuvat vaativatkin tyypillisesti huomattavan paljon tilaa ja siksi niitä joudutaan usein

käytännön sovelluksia varten tiivistämään käyttämällä jotain tiivistysmenetelmää.

Ideaalitapauksessa tiivistykseen voidaan käyttää häviötöntä menetelmää. Tällä tavoin saavutettavissa olevat tulokset ovat kuitenkin varsin rajallisia. Mikäli halutaankin pakkaussuhde, joka on suurempi kuin noin 2:1, on normaalisti turvauduttava häviölliseen tiivistykseen [19]. Häviöllisessä tiivistyksessä osa alkuperäisestä kuvainformaatiosta jätetään tarkoituksellisesti pois tiivistysprosessin aikana. Tästä johtuen kuvassa esiintyvän mielivaltaisen pikselin p intensiteetin $I(p)$ arvo tiivistysprosessin läpikäynnin jälkeen rekonstruoidussa kuvassa $Q(p)$ saattaa olla erilainen alkuperäiseen intensiteettiin verrattuna.

Useimmissa häviöllisissä tiivistysmenetelmissä käytetään hyväksi jotakin seuraavista tekniikoista: muunnos-koodausta, vektorikvantisointia, fraktaaleja, alitaajuuskoodausta tai wavelet-koodausta [6]. Tavoitteena on poistaa kuvasta psykovisuaalisesti ja tilastollisesti tarpeetonta tietoa. Kun bittimäärää tarpeeksi lasketaan, kukin tiivistysmenetelmä lopulta myös aiheuttaa vääristymiä kuvan laatuun. Näiden vääristymien tyyppi ja luonne on tavallisesti etukäteen tiedossa. Tuloksena voi syntyä menetelmästä riippuen esimerkiksi rosoisia, sumeita, laikullisia tai tuhruisia kuvia.

Erilaisia standardeja kuvantiivistysmenetelmiä on kehitetty, näistä esimerkkinä laajalti käytetty *JPEG*-pakkaus [19]. Standardien tiivistysmenetelmien jatkoksi on olemassa myös suuri joukko eri sovellusalueille erityisesti räätälöityjä menetelmiä. Eri sovellusalueella tiivistysmenetelmiltä voidaankin edellyttää erilaisia ominaisuuksia. Tästä syystä useita erilaisia tiivistysmenetelmiä käytetään yhä samanaikaisesti. Useimmat menetelmistä tukevat lisäksi yhtä tai useampaa parametria, jonka avulla lopputulokseen voidaan myös vaikuttaa. Siksi oikean tiivistysmenetelmän valinta ja sen parametointi sovellusalueelle sopivaksi on tärkeää parhaan mahdollisen lopputuloksen saavuttamiseksi.

Voidaan ajatella, että häviöllisen tiivistysmenetelmän suoriutumista tehtävästään voidaan arvioida sen perusteella, kuinka paljon tiivistysprosessin läpikäynyt kuva muistuttaa alkuperäistä kuvaa jotain tiettyä kriteeriä apuna käyttäen. Toisin sanoen, käyttämällä haluttua mittaa $d(I,Q)$ kuvajoukossa voidaan määrittää pakkauksen jälkeen rekonstruoidun kuvan ja alkuperäisen kuvan 'etäisyys' toisiinsa nähden. Tämän kaltaisia mittoja kutsutaan laatumitoiksi [19].

Kun käytettävä laatumitta on tiedossa, voidaan optimaalisin tiivistysmenetelmä valita sen perusteella, minkä menetelmän tuloksista mitattu keskimääräinen etäisyys kuvien kesken on pienin [19]. Tässä lähestymistavassa parhaan tiivistysmenetelmän valinta siis edellyttää aluksi sopivan laatumitan löytämistä. Joissakin tapauksissa tämä voi olla jopa triviaali tehtävä: esimerkiksi jos ollaan kiinnostuneita ainoastaan yhdestä tietystä kuvissa esiintyvistä piirteistä, voidaan mitaksi valita sellainen, joka mittaa kuvien eroavaisuuden määrän juuri tämän kyseisen ominaisuuden osalta. Monissa käytännön tilanteissa ei kuitenkaan etukäteen voida tietää, mistä kuvien piirteistä ollaan kiinnostuneita.

Laatumittoja voidaan luokitella eri perustein. Koska tavallisten kuvien tapaan myös digitaaliset kuvat on yleensä tarkoitettu ihmissilmällä katseltaviksi, luotettavin arvio niiden laadusta saadaan käyttämällä arvioinnissa ihmisen omaa havainnointia [6]. Tämän kaltaisia subjektiivisia mittoja on kahden tyyppisiä, absoluuttisia ja vertailevia. Absoluuttinen arviointi on prosessi, jossa havainnoija antaa kullekin kuvalle arvosanan ennalta määrätyn asteikon mukaisesti. Vertailevassa arvioinnissa puolestaan verrataan kuvapareja keskenään ja näistä valitaan parempi kunnes paremmuusjärjestys on selvillä.

Subjektiivisissa arvioinneissa on kuitenkin myös haittapuolensa, joista eräs on toistettavuusongelma. Tuloksiin nimittäin vaikuttavat muun muassa kuvien tyyppi, niiden koko sekä käytetty kuvavalikoima. Lisäksi tuloksiin vaikuttaa myös havainnoijien tausta, motivaatio sekä tutkimusolosuhteet, joihin lukeutuu esimerkiksi valaistus ja näyttölaitteen laatu [6].

Subjekttiivisten mittojen vastakohtana ovat objektiiviset laatumitat, jotka ovat nykyään tärkeä osa erilaisia kuvankäsittelysovelluksia [7][37]. Ne voidaan jakaa edelleen kahteen luokkaan lähestymistavan perusteella. Ensimmäisenä ovat matemaattisesti määritellyt kvantitatiiviset mitat, joihin lukeutuu perinteisesti paljon käytetty neliöllinen keskivirhe (MSE), sen johdannaiset signaali-kohinasuhde (SNR) ja huippu signaali-kohinasuhde ($PSNR$) sekä näiden lisäksi muun muassa keskivirhe ($RMSE$) ja keskimääräinen absoluuttinen virhe (MAE). Toisessa luokassa ovat puolestaan sellaiset mitat, joissa pyritään huomioimaan ja käyttämään hyväksi ihmisen näköjärjestelmän rajoituksia ja erityispiirteitä.

Kvantitatiiviset laatumitat ovat tärkeitä, kun halutaan arvioida ja testata erilaisia kuvien tiivistysmenetelmiä [6]. Ideaalisesti paras laatumitta olisi sellainen, joka on helppo muodostaa, riippumaton katseluetäisyydestä ja kykenee mittaamaan kaiken tyyppisiä vääristymiä kuvissa.

Oletetaan, että I on referenssikuva, jonka koko on $X \cdot Y$. Vastaavasti Q on saman kokoinen, vääristymille altistettu kuva. Kyseisten kuvien välinen MSE-arvo voidaan määrittää seuraavalla tavalla [35]:

$$MSE = \frac{1}{XY} \sum_{x,y} (I(x,y) - Q(x,y))^2 \quad (18)$$

Useissa yhteyksissä käytetään myös edellisten sukulaismittoja SNR ja $PSNR$, jotka määritellään seuraavasti [12]:

$$SNR = 10 \log_{10} \frac{E^o}{E^{cr}}, PSNR = 10 \log_{10} \frac{XY \cdot s^2}{E^{cr}} \quad (19)$$

Tässä s on kuvan signaalin huippuarvo, käytännössä normaalisti $s = 2^8 - 1 = 255$. E^o on alkuperäisen kuvan energia ja E^{cr} puolestaan on alkuperäisen kuvan ja vääristyneen kuvan energioiden erotus.

Edellä esitetyn kaltaisten matemaattisten mittojen heikkouksiin voidaan lukea muun muassa se, että ne muodostetaan tavallisesti pikselitasolla, jolloin kuvien rakenteellinen informaatio jää lähes kokonaan vaille

huomiota. Tällä tavoin mitatut arvot korreloivatkin ihmisen näköhavainnon kanssa varsin huonosti [7][35]. Ongelmana on se, että jokainen eroavaisuus visuaalisesta kontekstista riippumatta painotetaan laskennassa samalla painoarvolla. Jokainen pikseli digitaalisessa kuvassa ei kuitenkaan ole ihmissilmän kannalta samanarvoinen. Näin ollen tarvitaankin ns. kvalitatiivisia mittoja, joiden avulla pyritään määrittämään kuvien laatu sellaisena kuin ihminen sen havaitsee.

Vaikka matemaattisten mittojen heikkoudet on yleisesti tunnustettu, niiden käyttö saattaa vielä edelleenkin olla houkutteleva vaihtoehto lähinnä kahdesta syystä. Ensinnäkin ne ovat tavallisesti huomattavasti yksinkertaisempia ja laskennaltaan kevyempiä monimutkaisiin havaintojärjestelmään perustuviin mittoihin verrattuna. Toiseksi yhdenkään ihmisen havaintojärjestelmään perustuvan mitan suoriutumista ei ole tähän mennessä voitu osoittaa kriittisessä ja yleisluontoisessa testitilanteessa oleellisesti esimerkiksi PSNR -tyyppisiä mittoja paremmaksi [37].

2.4.1 Ihmisen näköjärjestelmä

Useimpien visuaalista laatua mittaavien mallien lähtökohtana on pyrkiä määrittämään sitä, millä tavoin ihmisen näköjärjestelmään liittyvä matalan tason fysiologia rajoittaa ihmisen havaintokykyä kuvien havaitsemisen suhteen [28]. Malleihin voidaan sisällyttää sen kaltaisia ilmiöitä kuin valaistustaso, paikkataajuus, signaalisältö ja liikkuvan kuvan tapauksessa myös ajallinen muutos. Ihmisen näköjärjestelmän ominaisuuksista pyritään muodostamaan kynnsarvoja, joiden perusteella määräytyy se vääristymän maksimimäärä, jonka lisääminen kuvaan aiheuttaa ihmiselle havaittavissa olevan aistimuksen. Tätä kutsutaan joskus termillä juuri havaittavissa oleva vääristymä (*JND level*). Toisin sanoen ihmissilmä ei kykene erottamaan kuvassa esiintyvää vääristymää jos vääristymän määrä alittaa tämän kynnsarvon.

Valoisuuden aistimus ihmissilmässä määräytyy epälineaarisesti luminanssin funktiona. Ilmiö tunnetaan myös näköjärjestelmän amplitudin

epälineaarisuutena. Tähän liittyy niin sanottu Weberin laki, joka voidaan esittää kaavalla $\frac{\Delta I}{I} = k$ [36]. Tässä ΔI on sellainen intensiteetin muutoksen kynnyisarvo, jonka ihmissilmä juuri ja juuri kykenee erottamaan. I vastaavasti on alkuperäinen intensiteettiarvo ja k puolestaan vakio, jonka suuruuteen I :n arvolla ei ole vaikutusta.

Kontrastiherkkyysfunktio pyrkii kuvaamaan ihmisen näköjärjestelmän taajuusvastetta [28]. Tätä funktiota voidaan ajatella päästökaistasuotimena ja se voidaan määrittää kokeellisesti. On huomattava, että tietyllä taajuudella esiintyvien yksityiskohtien erotettavuus riippuu myös katseluetäisyydestä. Näin ollen kuvien laatumitoilta voidaan edellyttää sellaisen lähimmän katseluetäisyyden kiinnittämistä, jota apuna käyttäen vääristymän suuruus määritetään.

Luonnolliset kuvat sisältävät tavallisesti laajan taajuuksien kirjon monissa eri skaaloissa. Tästä aiheutuu ilmiö, joka tunnetaan kontrastipeittona. Tilannetta voidaan havainnollistaa esimerkiksi, jota varten tarkastellaan kahta kuvitteellista kuvaa. Toisessa kuvassa intensiteetti on vakio ja toinen esittää esimerkiksi hiekkarantaa. Molempiin lisätään sellainen satunnaista kohinaa muistuttava signaali, joka vakiointensiteettikuvassa on juuri havaittavissa. Koska hiekkarantakuvassa kuitenkin esiintyy eri taajuuskomponentteja runsain mitoin jo valmiiksi, nämä peittävät häiriösignaalin alleen siten, että se ei olekaan enää havaittavissa. Kontrastipeitto viittaa siis tilanteeseen, jossa yhden kuvakomponentin havaittavuus vähenee toisen, samantyyppisen komponentin esiintymisen johdosta [28].

Liikkuvan kuvan laatua määritettäessä tärkeä käsite on myös ajallinen peittyminen [28]. Kirjallisuudessa ilmiötä käsitellään yleensä kahdessa muodossa, jotka ovat ympäristönvaihdos ja ajallinen kontrastiherkkyysfunktio.

2.4.2 Havaintokykyyn perustuva kuvien vertailu

Kuten edellä todettu, neliölliseen keskivirheeseen (MSE) pohjautuvat mallit painottavat jokaista eroa kahden kuvan välillä samanarvoisena. Tällöin esimerkiksi selvästi taka-alalla olevat, kohinasta johtuvat virheet kuvan taustalla vaikuttavat mitattuun laatuun yhdenvertaisesti sellaisten virheiden kanssa, jotka esiintyvät ihmissilmän kannalta tärkeissä objekteissa. Kun laatua mitataan kvalitatiivisesti eli laadullisesti, eroavaisuuksia (virheitä) ihmisen havaintokykyyn kannalta tärkeämissä kuvan osissa pyritään painottamaan muita osia enemmän. Seuraavassa esitellään Tompan, Mortonin ja Jerniganin paikallisen mielenkiinnon kohde-energia -menetelmä [35], joka pyrkii tähän tavoitteeseen laajentamalla perinteistä MSE-mittaa tähän suuntaan.

Paikallista mielenkiinnon kohdetta kuvassa voidaan mitata useilla eri tavoilla, esimerkiksi käyttämällä apuna harmaasävytasoa, paikallista varianssia, reunaisuutta tai spatiaalista hitausmomenttia. Harmaasävytasoa ja paikallista varianssia voidaan käyttää sellaisenaan MSE:n painottamiseen, koska havainnoija saattaa kiinnittää enemmän huomiota kirkkaisiin tai suuren kontrastin omaaviin kuvan osiin. Lisäksi katsojan huomio voi kiinnittyä tavallista enemmän myös reunaisuuden määrään ja reunojen suuruuteen, joita voidaan mitata esimerkiksi Sobel -reunanilmaisimella. Niinpä myös tätä ominaisuutta voidaan käyttää [35].

Toinen hitausmomentti lasketaan kohtelemalla kuvan harmaasävytasoja massa-arvoina. Tämä piirre pyrkii kuvaamaan pikselien avaruudellista jakaumaa ja se on määritelty sekä x - että y -suuntaan. Myös sitä voidaan käyttää piirteenä MSE:n painottamisessa. Seuraavassa on hitausmomentin kaava y -suunnassa [35]:

$$F(x, y) = \sum_{i=-\frac{a}{2}}^{\frac{a}{2}} \sum_{j=-\frac{a}{2}}^{\frac{a}{2}} j^2 I(x+i, y+j) \quad (20)$$

Tässä I on siis kuvan osa ja a on käytettävän ikkunan koko molemmissa ulottuvuuksissa.

Todennäköisyys tietyn harmaasävyn esiintymiselle kuvassa on määritelty kyseisen sävyn suhteellisena osuutena kaikista kuvan pikseleistä. Tämä todennäköisyys lasketaan jakamalla harmaasävyn esiintymien lukumäärä h_c kuvan pikselien kokonaismäärällä $X*Y$. Tätä kutsutaan Shannonin itseinformaatiomitaksi, jonka avulla voidaan siis määrittää tietyn kuvaelementin 'ainutlaatuisuus' koko kuvan alueella [35]:

$$S_c(I) = \frac{h_c(I)}{XY} \quad (21)$$

Tästä todennäköisyydestä voidaan puolestaan määrittää se informaation määrä, joka sisältyy kuhunkin kuvassa esiintyvään tapahtumaan, edellä siis esimerkiksi tietyn harmaasävyn esiintymiseen. Tämän informaatioarvon täytyy olla esiintymistodennäköisyyden suhteellinen käänteisluku, jolloin tulokseksi saadaan suuri informaatioarvo harvoin esiintyville tapahtumille ja kääntäen myös pieni arvo usein esiintyville tapahtumille. Jotta säilytetään tärkeät informaatioarvojen väliset suhteet, lopullinen arvo määritetään ottamalla logaritmi tästä esiintymistodennäköisyyden käänteisluvusta [35]:

$$S'_c(I) = \log \frac{1}{S_c(I)} \quad (22)$$

Näin saadaan positiivinen lukuarvo, joka ilmaisee sitä informaation määrää, joka liittyy tiettyyn tapahtumaan. Tapahtuma voi olla paitsi harmaasävytaso, myös jokin muu edellä mainituista energiamitoista. Käyttämällä näitä energiamittoja tapahtumina saadaan joukko erilaisia mitta-arvoja, jotka pyrkivät kuvaamaan havainnoitua paikallista mielenkiintoa kuvassa [35].

Kuvien vertailu tämän menetelmän avulla tapahtuu seuraavalla tavalla. Aluksi sekä alkuperäinen että vääristynyt kuva painotetaan niiden vastaavilla paikallista mielenkiintoa kuvaavilla informaatiokartoilla. Tämän jälkeen lasketaan MSE normaaliin tapaan. Näin varmistetaan se, että niitä vääristymiä jotka ovat havainnoinnin kannalta tärkeitä myös painotetaan muita enemmän [35].

Ennen informaatiokartan muodostamista kuva käsitellään eksponenttioperaattorilla. Näin voidaan vähentää algoritmin herkkyyttä pieniin vaihteluihin kuvan tummilla alueilla. Esikäsitteilyn jälkeen kuva muunnetaan piirrekartaksi käyttäen piirteenä haluttua kuvan ominaisuutta tai tapahtumaa. Sen jälkeen Shannonin itseinformaatiomittaa sovelletaan piirrekartalle. Prosessin tuloksena saadaan kuvalle informaatiokartta $I'(x,y)$, jota käytetään lopuksi MSE-mitan painokertoimena. Alla on esitetty tällä tavoin syntyvä paikallisen mielenkiinnon kohde-energia -piirre [35]:

$$IMSE = \frac{1}{XY} \sum_{x,y} (I(x,y)I'(x,y) - Q(x,y)Q'(x,y))^2 \quad (23)$$

Yllä I' on alkuperäisen kuvan I informaatiokartta. Vastaavasti Q' on vääristymälle altistetun kuvan Q informaatiokartta.

Menetelmän kehittäjät ovat tutkineet sen suoriutumista erilaisten kuvien ja vääristymien arvioinnissa. Saadut tulokset ovat käytettyjen testitapausten osalta parempia kuin mitä saataisiin käyttämällä painottamatonta MSE -mittaa sellaisenaan.

2.4.3 Yleinen kuvien laatuindeksi

Wang ja Bovik [37] ovat myös kehittäneet vaihtoehdon MSE:n käytölle kuvan laadun määrittämisessä. Yleinen kuvien laatuindeksi on matemaattisesti määritelty mitta, jota voidaan käyttää kuvien laadun mittaamiseksi. Kehittäjien mukaan se on riippumaton käytetyistä kuvista, katseluolosuhteista ja havaitsijoista. Lisäksi sitä voidaan soveltaa useissa erilaisissa kuvankäsittelysovelluksissa. Se myös tuottaa mielekkään tuloksen vertailtaessa monen tyyppisiä vääristymiä kuvissa.

Oletetaan, että I on alkuperäinen kuvasignaali, joka koostuu intensiteettiarvoista $I(i)$, $i = 1,2,\dots,XY$. Lisäksi Q on vääristynyt kuvasignaali, joka koostuu vastaavalla tavalla intensiteettiarvoista $Q(i)$. Yleinen kuvien laatuindeksi G voidaan nyt määritellä seuraavasti [37]:

$$G = \frac{s_{IQ}}{s_I s_Q} \cdot \frac{2\bar{I}\bar{Q}}{(\bar{I})^2 + (\bar{Q})^2} \cdot \frac{2s_I s_Q}{s_I^2 + s_Q^2} \quad (24)$$

jossa

$$\bar{I} = \frac{1}{XY} \sum_{i=1}^{XY} I(i) \quad \bar{Q} = \frac{1}{XY} \sum_{i=1}^{XY} Q(i) \quad s_I^2 = \frac{1}{XY-1} \sum_{i=1}^{XY} (I(i) - \bar{I})^2 \quad s_Q^2 = \frac{1}{XY-1} \sum_{i=1}^{XY} (Q(i) - \bar{Q})^2$$

$$s_{IQ} = \frac{1}{XY-1} \sum_{i=1}^{XY} (I(i) - \bar{I})(Q(i) - \bar{Q})$$

Laatuindeksin G ensimmäinen komponentti on korrelaatiokerroin I :n ja Q :n välillä, joka mittaa näiden kahden välisen lineaarisen korrelaation määrää välillä $[-1,1]$. Paras arvo, 1, saavutetaan siinä tilanteessa, jossa $Q(i) = a \cdot I(i) + b$ ($a > 0$ ja b ovat vakioita) kaikille $i = 1, 2, \dots, XY$.

Vaikka I ja Q olisivatkin lineaarisessa suhteessa, suhteellisia vääristymiä voi kuitenkin esiintyä niiden välillä. Näitä pyritään mittaamaan kahdella jälkimmäisellä virhekomponentilla. Toinen komponentti mittaa sitä, kuinka lähellä keskimääräinen luminanssi on I :n ja Q :n välillä. Se saa arvoja väliltä $[0,1]$ ja saavuttaa huippuarvon 1 ainoastaan siinä tapauksessa, että $\bar{I} = \bar{Q}$. Lopuksi viimeinen komponentti mittaa sitä, miten lähellä kuvien kontrastit ovat toisiaan. Myös sen arvoalue on $[0,1]$, josta 1 saavutetaan ainoastaan siinä tapauksessa, että $s_I = s_Q$ [37].

Näin ollen laatuindeksin G dynaaminen arvoalue on $[-1,1]$. Paras arvo, 1, voidaan saavuttaa vain ja ainoastaan siinä tapauksessa, että $I(i) = Q(i)$ kaikille $i = 1, 2, \dots, XY$. Sen sijaan heikoin arvo, -1, saavutetaan tilanteessa, jolloin ehto $Q(i) = 2\bar{I} - I(i)$ on voimassa kaikille $i = 1, 2, \dots, XY$. Kyseinen laatuindeksi siis mittaa millaista tahansa yhdistelmää kolmen tyyppisiä vääristymiä: korrelaation häviämistä sekä vääristymiä kuvien luminanssissa ja kontrastissa [37].

Kuvan laatua halutaan usein käytännössä ilmaista käyttäen siihen ainoastaan yhtä lukuarvoa huolimatta siitä, että laatu kuvassa vaihtelee yleensä paikallisesti. Siksi onkin enemmän tarkoituksenmukaista mitata tilastollisia piirteitä ainoastaan paikallisella tasolla ja muodostaa niiden pohjalta yksi laatuarvo vasta prosessin päätteeksi.

Myös yleinen laatuindeksi lasketaan globaalin lähestymistavan sijaan paikallisesti, käyttäen apuna $a \times a$ pikselin kokoista 'liukuvaa ikkunaa'. Ikkunaa siirretään kuvan alueella yksi pikseli kerrallaan. Kuvan rivit ja sarakkeet käydään läpi, aloittaen vasemmasta yläkulmasta ja päätyen oikeaan alakulmaan. Tässä prosessissa askeleen j aikana määritetään paikallinen laatuindeksi G_j kyseessä olevan ikkunan alueella. Jos oletetaan, että koko prosessissa on yhteensä M askelta, niin lopullinen laatuindeksi lasketaan paikallisten arvojen keskiarvona seuraavasti [37]:

$$\bar{G} = \frac{1}{M} \sum_{j=1}^M G_j \quad (25)$$

Yleisen kuvien laatuindeksin kehittäjät vertasivat sen suoriutumista MSE:n suoriutumiseen tilanteessa, jossa testikuvat altistettiin erilaisille vääristymille, mukaan lukien *JPEG*-tiivistys. Tuloksia verrattiin koehenkilöillä teetettyihin subjektiivisiin arvioihin samoista vääristymistä. Osoittautui, että laatuindeksi korreloi subjektiivisten arvioiden kanssa oleellisesti MSE:tä paremmin.

2.4.4 Laatumitta spektrikuvien tiivistykselle

Laatumitta spektrikuvien tiivistykselle on kvalitatiivinen menetelmä, jonka avulla voidaan mitata erityisesti tiivistysprosessin läpikäyneiden spektrikuvien laatua [12]. Se on kehitetty laajentamalla harmaasävykuville tarkoitettua ruuduittaista virhemittaa [9] siten, että sillä voidaan käsitellä harmaasävykuvien lisäksi myös spektrikuvia. Kehittäjien mukaan tätä laatumittaa voidaan käyttää millaisten tahansa spektrikuvien arviointiin. Mitta huomioi sekä suhteelliset että absoluuttiset virheet kuvissa. Tulokset ovat myös keskenään vertailukelpoisia riippumatta käytetystä kuvajoukosta

tai tiivistysmenetelmästä (ts. vääristymän luonteesta). Lisäksi laatumitta perustuu kuvan tärkeisiin piirteisiin, jolloin myös visuaaliset havainnot sopivat yhteen sen antamien tulosten kanssa [12].

Harmaasävykuvien ruuduittainen virhemitta käyttää 3*3 pikselin kokoista liukuvaa ikkunaa kuvan läpikäyntiin. Kunkin ikkunan alueelta lasketaan kolme eri komponenttia: kontrasti, spatiaalinen rakenne sekä toisistaan poikkeavien harmaasävyjen lukumäärä. Kontrasti mittaa sitä, kuinka paljon kukin pikseli eroaa taustastaan. Reuna-alueet kuvissa ovat tärkeitä, sillä ne saattavat sumentua tai muuttua rakeisiksi tiivistysprosessin aikana. Tätä mitataan spatiaalisen rakenteen avulla. Lisäksi rosoisuus on tyypillinen tiivistyksen tuloksena syntyvä vääristymä. Sitä pyritään mittaamaan ikkunan alueella esiintyvien ainutlaatuisten intensiteettiarvojen lukumäärän avulla [12][9].

Spektrikuvaa voidaan ajatella kolmiulotteisena kuvana, jolla on kaksi spatiaalista ulottuvuutta ja lisäksi yksi spektriulottuvuus [12]. Niinpä harmaasävyvirhemitan laskennassa käytetty kaksiulotteinen liukuva ikkuna muutetaan kolmiulotteiseksi kuutioksi tämän menetelmän tarpeita vastaavasti.

Kontrasti on paikallinen kirkkauden muutos. Se määritellään kunkin pikselin kirkkauden suhteena sen taustan kirkkauteen. Koska ihmissilmän herkkyys valoisuudelle on luonteeltaan eksponentiaalinen, harmaasävykuvan kontrastia voidaan mitata käyttämällä keskihajontaa. Spektrikuville kontrasti lasketaan harmaasävykuvia vastaavalla tavalla, ottaen huomioon kaikki kuution sisällä olevat pikselit [12]:

$$s(I) = \frac{1}{a^3} \sum_{i=-\frac{a}{2}}^{\frac{a}{2}} \sum_{j=-\frac{a}{2}}^{\frac{a}{2}} \sum_{k=-\frac{a}{2}}^{\frac{a}{2}} \left(I(x+i, y+j, z+k) - \bar{I} \right)^2 \quad (26)$$

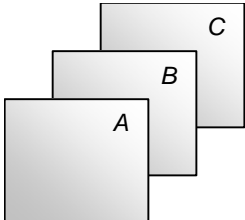
Yllä $I(x,y,z)$ on pikselin intensiteettiarvo pisteessä (x,y,z) , \bar{I} on sitä ympäröivän naapuruston keskimääräinen intensiteetti ja a on käytetyn ikkunan koko kaikissa ulottuvuuksissa.

Ensimmäinen virhekomponentti, kontrastivirhe e_c , lasketaan keskihajontojen erotuksen neliönä alkuperäisen kuvan I ja mitattavan kuvan Q vastinalueiden välillä [12]:

$$e_c = \frac{(\mathbf{s}(I) - \mathbf{s}(Q))^2}{\max(1, \mathbf{s}(I))} \quad (27)$$

Komponentin arvo normalisoidaan alueen kontrastilla $\mathbf{s}(I)$ siitä syystä, että ihmissilmä on herkempi muutoksille matalan kontrastin alueilla verrattuna suuremman kontrastin omaaviin alueisiin.

Toinen virhekomponentti perustuu pikselien spatiaaliseen rakenteeseen. Sen arvo saadaan ikkunalle suoritetun reunanetsintäoperaation tuloksena. Tähän käytetään Laplace-suodinta, joka on ensin muunnettu kolmiulotteiseksi. Tällöin se mittaa reunaisuuden määrää myös spektriulottuvuuden suuntaisesti. Reunasuodinoperaatiot g_i , $i=x,y,z$ kolmiulotteiselle $3 \times 3 \times 3$ kuutiolle noudattavat seuraavanlaista rakennetta [12]:

$$g_i = \frac{1}{16} \times \begin{array}{c} \text{C} \\ \text{B} \\ \text{A} \end{array} \quad (28)$$


Tässä A , B ja C ovat Laplace-suotimesta johdettuja 3×3 reunasuotimia. Kyseiset Laplace-suodinmatriisit on esitetty liitteessä 1.

Itse spatiaalinen virhekomponentti, e_s , on arvoltaan kaikkien reunaoperaatioiden tulosten summa, lopuksi vielä ensimmäisen virhekomponentin tapaan alueen paikallisella kontrastilla normalisoituna [12]:

$$e_s = \frac{|g_x(I) - g_x(Q)| + |g_y(I) - g_y(Q)| + |g_z(I) - g_z(Q)|}{3 \max(1, \mathbf{s}(I))} \quad (29)$$

Edellä $g(I)$ on reunasuotimen tulos alkuperäiselle kuvalle ja $g(Q)$ puolestaan on vastaava arvo vääristyneestä kuvasta laskettuna.

Kolmas ja viimeinen virhekomponentti, kvantisointivirhe e_K , määritellään molemmissa kuvissa kuution alueella esiintyvien ainutlaatuisten intensiteettiarvojen lukumäärien, $q(I)$ ja $q(Q)$, erotuksen neliönä [12]:

$$e_K = (q(I) - q(Q))^2 \quad (30)$$

Alkuperäisen ja vääristyneen kuvan väliset kokonaisvirheet, \bar{e}_C , \bar{e}_S ja \bar{e}_K , määritellään paikallisten virhearvojen e_C , e_S ja e_K keskiarvona koko kuvan alueella [12]:

$$\bar{e}_C = \frac{1}{N} \sum_{i=1}^N (e_C)_i, \quad \bar{e}_S = \frac{1}{N} \sum_{i=1}^N (e_S)_i, \quad \bar{e}_K = \frac{1}{N} \sum_{i=1}^N (e_K)_i \quad (31)$$

Tässä N on paikallisten $3*3*3$ kuutioiden kokonaislukumäärä kuvien alueella. Summattavat e_C , e_S ja e_K on määritelty edellä. Virhemitan lopullinen arvo e_{lin} on keskimääräisten virheiden painotettu summa [12]:

$$e_{lin} = w_C f(\bar{e}_C) + w_S f(\bar{e}_S) + w_K f(\bar{e}_K) \quad (32)$$

Edellä w_C , w_S ja w_K ovat painokertoimia, joiden suuruus määritetään empiirisesti siten, että korrelaatio mitan e_{lin} ja subjektiivisten arvioiden välillä on maksimaalinen. Funktio f toimii skaalausfunktiona, joka on määritelty skaalauskerroimien k_i avulla seuraavasti [12]:

$$f(\bar{e}_i) = 1 - \min\left(1, \frac{\bar{e}_i}{k_i}\right), \quad i = C, S, K \quad (33)$$

Kertoimet k_i skaalaavat funktion f arvot välille $[0,1]$ ja samalla ne toimivat myös kynnyksarvoina. Mikäli $\bar{e}_i > k_i$, niin $f(\bar{e}_i) = 0$.

Spektrikuvien virhemitan avulla saavutettavaan tulokseen vaikuttaa oleellisesti kunkin virhekomponentin painottamiseen käytetyt kertoimet. Kertoimien arvot määritettiin subjektiivisten testien perusteella [12], jotka

jakautuivat kahteen osaan. Ensimmäisessä osassa pyrittiin määrittämään kolmen käytetyn testikuvan kuvakohtainen laatu eri suuruusluokkaa olevien vääristymien kohdalla. Tässä yhteydessä vääristyminä käytettiin erilaisten häviöllisten tiivistysmenetelmien aiheuttamia virheitä kuvan laadussa. Lopuksi kuvista saadut arviot pyrittiin yhdistämään samaan mittakaavaan siten, että vertailut myös eri kuvien kesken olisivat mielekkäitä saman asteikon puitteissa. Kuvien subjektiivisessa arvioinnissa käytettiin ns. mean opinion score (MOS) –menetelmää. Prosessin tulokset on nähtävissä lähteessä [12].

2.5 LBP-TEKSTUURIPIIRRE

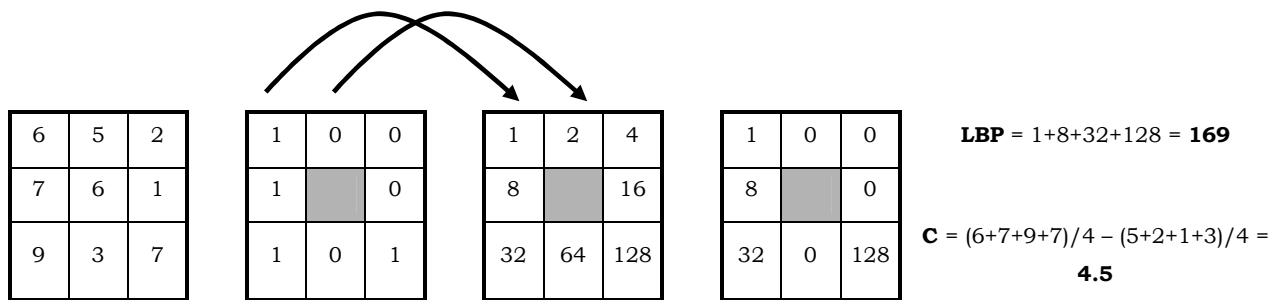
Kuvien segmentointi niissä esiintyvien tekstuuripintojen perusteella voi olla haastava tehtävä. Useinkaan ei voida etukäteen tietää, minkä tyyppisiä tekstuureja kuvissa esiintyy, kuinka monta erilaista tekstuuria lukumäärällisesti voidaan erotella ja mitkä kuvan alueista ilmentävät mitään näistä tekstuureista. Jotta voitaisiin luotettavasti erottaa kaksi tekstuuria toisistaan, täytyy tarkastella varsin suurta näytettä molemmista, toisin sanoen suhteellisen suurta kuvan aluetta. Kuitenkaan suuri kuva-alue ei enää välttämättä kuulu kokonaan samaan tekstuurialueeseen, jolloin ongelmaksi muodostuu alueiden välisten rajojen määrittäminen [27].

Korkean erottelukyvyn omaavien tekstuuripiirteiden valinta on tärkein tekijä tekstuurisegmentoinnin onnistumisessa. Valittujen piirteiden tulisi kyetä erottelemaan monen tyyppisiä tekstuureja. Lisäksi niiden laskennassa käytettävän ikkunan koon tulisi olla kyllin pieni, jotta se olisi käyttökelpoinen myös pienille kuvan osille. Pienikokoisen ikkunan käyttö saattaa myös tuottaa vähemmän virheitä alueiden reunoilla. Tekstuurin rakennetta pyritään seuraavassa kuvaamaan käyttämällä siihen kahden yhdistetyn piirteen jakaumaa: paikallista binäärikuviota (*LBP*) sekä kontrastipiirrettä (*C*) [27].

LBP-piirteiden laskentaprosessi tietyn pikselin naapurustossa on seuraava. Kyseisen pikselin alkuperäinen 3*3 naapurusto kynnystetään keskimmäisen

pikselin arvolla. Tuloksena saadaan binäärimatriisi, jonka alkiot koostuvat arvoista 0 ja 1. Arvoista 0 esiintyy siinä tapauksessa, että kyseistä alkiota vastaava pikseliarvo oli keskustan pikseliarvoa aidosti pienempi ja vastaavasti arvo 1 muutoin. Seuraavaksi tämän binäärimatriisin alkiot kerrotaan binomisista painokertoimista koostuvalla kerroinmatriisilla. Lopullinen ikkunan LBP-arvo saadaan ottamalla näin syntyneen tulomatriisin alkioiden summa.

Kontrastiarvon C laskenta puolestaan suoritetaan seuraavasti. Mikäli \overline{C}_1 on ikkunan alkuperäisten pikseliarvojen keskiarvo niiden alkioiden osalta, jotka binäärimatriisissa saivat arvon 1, ja vastaavasti \overline{C}_0 niiden alkuperäisten pikseliarvojen keskiarvo, jotka binäärimatriisissa saivat arvon 0, niin ikkunan kontrastiarvo määritellään näiden erotuksena $C = \overline{C}_1 - \overline{C}_0$. LBP:n ja kontrastipiirteen laskentaprosessia on havainnollistettu kuvassa 1 [27].



Kuva 1: LBP- ja kontrastipiirteen laskenta pikselin naapurustossa

LBP/C-jakaumaa kuvataan diskreetin kaksiulotteisen histogrammin avulla, jonka koko on $256 * n_C$. Tässä n_C on C :n eri luokkien lukumäärä C :n arvoalueen kvantisoinnin jälkeen. Histogrammien keskinäinen vertailu on mahdollista esimerkiksi alla esitetyn logaritmisen todennäköisyysuhteen E_G avulla (34). Kyseinen etäisyysmetriikka kuvastaa sitä todennäköisyyttä, jolla kaksi näytejakaumaa ovat peräisin samasta populaatiosta [27]:

$$\begin{aligned}
E_G = 2^* & \left[\sum_{h_1, h_2} \sum_{i=1}^n h_i \log h_i \right] - \left[\sum_{h_1, h_2} \left(\sum_{i=1}^n h_i \right) \log \left(\sum_{i=1}^n h_i \right) \right] \\
& - \left[\sum_{i=1}^n \left(\sum_{h_1, h_2} h_i \right) \log \left(\sum_{h_1, h_2} h_i \right) \right] + \left[\left(\sum_{h_1, h_2} \sum_{i=1}^n h_i \right) \log \left(\sum_{h_1, h_2} \sum_{i=1}^n h_i \right) \right]
\end{aligned} \tag{34}$$

Tässä h_1 ja h_2 ovat kaksi näytehistogrammia, jotka molemmat koostuvat yhteensä n alkiosta. Merkintä h_i tarkoittaa histogrammin i :nnen alkion arvoa. Mitä suurempi logaritmisien todennäköisyysuhteen E_G arvo, sitä epätodennäköisempää on, että kyseiset kaksi näytettä edustavat samaa populaatiota.

Ojala ja Pietikäinen [27] käyttävät kolmivaiheista algoritmia kuvien segmentointiin. Vaiheet ovat hierarkkinen osittaminen, alueiden yhdistäminen sekä pikselitason luokittelu. Hierarkkisessa osittamisessa suorakaiteen muotoinen kuva-alue jaetaan neljään tasakokoiseen alialueeseen siinä tapauksessa, että koko alue ei edusta yhtenäistä tekstuuria. Näin jatketaan iteratiivisesti, kunnes tietty lopetusehto täyttyy. Yhdistämisprosessissa yhdistetään samaa tekstuuria ilmentävät alueet. Lopuksi alueiden rajat pyritään pehmentämään pikselitasolla.

Kehittäjät ovat tutkineet edellä kuvatun menetelmän suoriutumista mm. erilaisten mosaiikkitekstuurikuvien ja myös tavallisia luonnonmaisemia esittävien kuvien segmentoinnissa [25][27]. Tutkimustulokset ovat olleet lupaavia, vaikka kehittäjät myöntävätkin, että hyvin eri kokoluokkaa olevien tekstuurien tunnistaminen kuvista luotettavasti ilman etukäteisinformaatiota niiden sisällöstä saattaa olla jopa epärealistinen vaatimus.

2.6 ITSEORGANISOIVA KARTTA

Itseorganisoiva kartta (SOM) on tehokas työväline moniulotteisen tiedon visualisointiin. Perusmuodossaan sen avulla voidaan tuottaa eräänlainen samankaltaisuuskuvaaja syötteenä käytettävälle moniulotteiselle datalle. Syötteen epälineaariset tilastolliset suhteet muuntuvat itseorganisoivassa

kartassa sen kuvapisteidien väliseksi yksinkertaisiksi geometrisiksi suhteiksi, jotka esitetään havainnollisen, matalaulotteisen esityksen avulla [14][16][17].

Koska itseorganisoiva kartta tiivistää tietoa samalla kun se säilyttää tärkeimmät esityksen primääristen elementtien väliset topologiset suhteet, sen voidaan ajatella myös muodostavan eräänlaisen abstraktion syötedatalle. Näitä kahta sen edellä mainittua ominaisuutta, visualisointia ja abstrahointia, voidaan käyttää hyväksi monissa erilaisissa sovelluksissa. Näihin lukeutuvat esimerkiksi prosessien analysointi, konenäkö, kontrollointi ja kommunikaatio [17].

Sellaiset verkkoarkkitehtuurit ja signaaliprosessit, jotka mallintavat hermostollisia järjestelmiä, voidaan jakaa niiden lähtökohtien perusteella karkeasti kolmeen ryhmään. Myötäkytkentäverkot muuntavat syötesignaalien joukon tulossignaalien joukoksi. Muunnos edellyttää yleensä ulkoista ja valvottua järjestelmän parametrien muuntelua. Takaisinkytkentäverkkojen tapauksessa alkuperäinen syöte määrittää erityisen palautejärjestelmän alkutilan. Tilanvaihdosten jälkeen muodostuva asymptoottinen lopputila tulkitaan laskennan tulokseksi. Kolmannessa ja viimeisessä hermostollisia järjestelmiä mallintavien prosessien ryhmässä neuraalisen verkon naapurisolut 'kilpailevat' aktiviteeteissaan keskenään ja vuorovaikuttavat toistensa kanssa, kunnes niistä adaptiivisesti kehittyy eräänlaisia detektoreja erilaisille signaalisarjoille. Tämän tyyppistä oppimista kutsutaan kilpailevaksi, ohjaamattomaksi tai itseorganisoivaksi. Näin ollen myös itseorganisoiva kartta kuuluu tähän viimeksi mainittuun kategoriaan [14].

Formaalisti itseorganisoiva kartta voidaan määritellä epälineaarisenä, järjestettynä ja tasaisena kuvauksena, jossa moniulotteinen syötedata kuvautuu säännöllisen, matalaulotteisen tietorakenteen soluille. Tämä kuvaus määritellään seuraavalla tavalla, joka muistuttaa hieman perinteistä vektorikvantisointia. Oletetaan yksinkertaisuuden vuoksi, että syötemuuttujien joukko $\{x_j\}$ voidaan määritellä reaalivektorina

$x = [x_1, x_2, \dots, x_n]^T \in \mathfrak{R}^n$. Itseorganisoiva kartta rakentuu verkkotopologiaa noudattavista soluista. Tämän tietorakenteen kuhunkin soluun liitetään parametrinen reaalivektori $m_i = [m_{i1}, m_{i2}, \dots, m_{in}]^T \in \mathfrak{R}^n$, jota tässä kutsutaan *malliksi*. Oletetaan lisäksi tunnetuksi yleinen etäisyysmetriikka x :n ja m_i :n välillä ja merkitään sitä $D(x, m_i)$. Syötevektorin *kuva* itseorganisoivan kartan solussa voidaan nyt määritellä siksi elementiksi m_a , jonka etäisyys x :n kanssa on kaikkein pienin. Toisin sanoen, kunkin syötevektorin kuva on se itseorganisoivan kartan solu, jonka indeksi a saadaan seuraavasti [15][16][17]:

$$a = \arg \min_i \{ D(x, m_i) \} \quad (35)$$

Perinteisestä vektorikvantisoinnista poiketen seuraava tehtävä on määrittää m_i sillä tavoin, että lopputuloksena saadaan järjestetty ja deskriptiivinen kuvaus x :n jakaumasta. On huomattava, että itseorganisoivan kartan mallien ei tarvitse välttämättä olla vektorityyppisiä. Riittää, että yleinen etäisyysmetriikka $D(x, m_i)$ on määritelty kaikkien x :n alkioden sekä riittävän suuren mallijoukon m_i välillä. Myöskään mallien alkuarvojen valinta ei ole lopputuloksen kannalta merkitsevä. Alkuarvot voidaan valita esimerkiksi satunnaisesti, mieluiten kuitenkin samasta joukosta syötteen kanssa [17].

Tarkastellaan seuraavaksi syötevektorien listaa $x(t)$, jossa t on kokonaislukuarvoinen indeksi. Jokaista syötevektoria $x(t)$ verrataan kaikkiin kartan malleihin m_i ja kopioidaan sitten tämän syötteen $x(t)$ arvo sellaiseen alilistaan, joka liittyy siihen kartan soluun, joka on käytetyn etäisyysmetriikan perusteella kaikkein lähimpänä kyseisen syötteen kanssa [17]. Kun kaikki $x(t)$:n syötealkiot on jaoteltu kukin vastaavaan alilistaansa edellä kuvatulla tavalla, tarkastellaan seuraavaksi naapurustoa N_i mallin m_i ympäristössä. Tämä kartan solujen joukko N_i koostuu niistä soluista, jotka sijaitsevat korkeintaan määrätyn etäisyyden päässä solusta i . Muodostetaan seuraavaksi yhdiste kaikista joukkoon N_i kuuluvista alilistoista ja etsitään tästä yhdistejoukosta sen 'keskimmäinen' syötearvo \bar{x}_i . Tämä määritellään siksi syötealkioksi, jolla on pienin etäisyyksien summa alkuperäisiin

syötteisiin $x(t)$ nähden, silloin kun $t \in N_i$. Tätä syötettä \bar{x}_i kutsutaan alilistojen joukon yleistetyksi mediaaniksi [17].

Seuraava tehtävä on muodostaa jokainen \bar{x}_i kullekin kartan solulle edellä kuvatulla tavalla, huomioiden aina naapurujoukko N_i jokaisen solun i ympäristössä. Lisäksi kukin mallin m_i vanha arvo korvataan uudella \bar{x}_i arvolla. Tätä prosessia jatketaan eteenpäin iteratiivisesti. Toisin sanoen alkuperäiset syötteen $x(t)$ jaotellaan uudestaan alilistoihin, jotka nyt siis muuttuvat, koska myös m_i on muuttunut. Uudet \bar{x}_i arvot lasketaan, vanhat mallit korvataan niillä ja niin edespäin. Kysymyksessä on eräänlainen regressioprosessi [17].

Tämä prosessi tuottaa lopulta asymptoottisesti lähestyvät arvot malleille m_i . Tuloksena on kokoelma, joka approksimoi syötteiden $x(t)$ jakaumaa tietyllä, järjestäytyneellä tavalla. Itseorganisoivan kartan tietorakenteen solujen muoto voidaan myös valita vapaasti: esimerkiksi suorakulmaisia, kuusikulmaisia tai peräti epäsäännöllisiä topologioita voidaan käyttää [17]. Näistä kuusikulmainen muoto soveltuu hyvin visuaaliseen esitykseen, joten sitä käytetään mm. tässä työssä.

Yllä kuvattua prosessia, jossa kartan soluja muokataan syötteen perusteella, kutsutaan kartan opettamiseksi. Opetusprosessi voidaan saattaa kaavan muotoon seuraavanlaisella notaatiolla [17][18]:

$$m_i(t+1) = m_i(t) + f_{ai}(t)[x(t) - m_i(t)] \quad (36)$$

Tässä $t = 0, 1, 2, \dots$ on kokonaislukuarvoinen, diskreetti aika-koordinaatti. Funktiolla f_{ai} on keskeinen rooli: se toimii ns. naapurustofunktiona, jonka tulee olla määritelty koko kartan alueelle. Konvergenssin vuoksi on tarpeen, että $f_{ai}(t) \rightarrow 0$, kun $t \rightarrow \infty$. Useimmiten $f_{ai}(t) = f(\|r_a - r_i\|, t)$, jossa $r_a, r_i \in \mathbb{R}^2$ ovat sijaintivektorit kartan soluille a ja i . Kun $\|r_a - r_i\|$ kasvaa, niin $f_{ai} \rightarrow 0$. Naapurustofunktion keskimääräinen laajuus ja muoto määrittävät

eräänlaisen 'jäykkyyden' syötedatan pisteisiin sovitettavalle 'elastiselle pinnalle' [17].

Kirjallisuudessa esiintyy säännöllisesti ainakin kaksi erilaista vaihtoehtoa naapurustofunktioksi. Näistä yksinkertaisempi on myös edellä kuvatun tyyppinen niiden naapurisolujen joukko, jotka ympäröivät solua a . Merkitään näitä naapurisoluja N_a (voidaan määritellä $N_a = N_a(t)$, joista jälkimmäinen ajan funktiona). Nyt $f_{ai}(t) = \mathbf{m}(t)$, jos $i \in N_a$ ja $f_{ai}(t) = 0$ jos $i \notin N_a$. Tässä arvo funktiolle $\mu(t)$ määräytyy ns. oppimismuutoksen kertoimen perusteella ($0 < \mu(t) < 1$). Sekä $\mu(t)$ että naapuruston $N_a(t)$ säde yleensä laskee monotonisesti oppimisprosessin aikana ajan funktiona [17].

Toinen usein käytetty, edellistä 'pehmeämpi' naapurustofunktio voidaan ilmaista Gaussin funktion muodossa seuraavasti [17]:

$$f_{ai}(t) = \mathbf{m}(t) \cdot \exp\left(-\frac{\|r_a - r_i\|}{2\mathbf{n}^2(t)}\right) \quad (37)$$

Tässä $a(t)$ on toinen, skalaariarvoinen oppimismuutoksen kerroin. Parametri $\mu(t)$ vastaa $N_a(t)$:n sädettä edellä. Sekä $\mathbf{m}(t)$ että $\mathbf{n}(t)$ ovat ajan mukana monotonisesti laskevia funktioita.

3 Käytetyt menetelmät ja testiaineisto

On hyvä pitää mielessä, ettei yhtäkään edellisissä luvuissa esitetyistä menetelmistä ole varta vasten kehitetty yksittäisten kuvien laadullista analysointia silmällä pitäen. Tämän kaltaisia menetelmiä esiintyy kirjallisuudessa erittäin harvoin, jos lainkaan. Joitakin yksittäisten kuvien laatua mallintamaan suunniteltuja tekniikoita on aivan viime aikoina kyllä esitetty, mutta ne perustuvat yleensä subjektiivisiin arviointeihin [23][31]. Pääasiallinen syy tähän lienee se, että kuvan laatu on ihmisen kannalta aina subjektiivinen käsite. Niinpä sellaisen objektiivisen laatumittan kehittäminen, jossa vertailukohtana ei ole laadultaan 'täydelliseksi' oletettua mallikappaletta, on erittäin vaikeaa, jos edes mielekästäkään [28].

Tästä huolimatta – kuten myös aiemmin todettu – tämän tutkimuksen tarkoituksena oli kuitenkin pyrkiä löytämään sen kaltaisia laskennallisia laadullisia piirteitä, joiden avulla saataisiin esille sellaisia säännönmukaisia eroja kuvien välille, jotka mahdollisesti voitaisiin lisäksi yhdistää kuvissa käytettyihin painopapereihin. Luvussa 5 tullaan toteamaan, että kuvat eivät tässä yhteydessä ole pikselitasolla suoraan vertailukelpoisia keskenään. Niinpä ainoa vaihtoehto on pyrkiä soveltamaan niihin sellaisia piirteitä, jotka käyttävät syötteenään ainoastaan yksittäisiä kuvia. Näin ollen tehtäväksi jää vertailla kuvia näitä piirteitä hyödyntäen sillä tavoin, että mahdollinen painopaperin vaikutus saataisiin lopulta paljastettua.

3.1 MENETELMIEN VALINNASTA

Aikaisemman tutkimusmateriaalin puuttuessa jouduttiin turvautumaan suhteellisen suureen testimenetelmien kirjoon. Menetelmät pyrittiin valitsemaan siten, että kukin edellisissä luvuissa esitelty menetelmätyyppi olisi edustettuna. Kokeellisessa osassa keskitytään siis testaamaan kirjallisuudessa esiintyviä piirteitä kuvien laadun mittaamisessa ja niiden keskinäisessä vertailussa.

Suuri osa kohdassa 2.3 esitellyistä kuvahakujärjestelmistä käyttää hyväkseen erilaisia värihistogrammeja tai niiden johdannaisia. Niinpä tämän piirteen käyttökelpoisuutta kokeiltiin myös kuvien laadullisessa analysoinnissa. Erilaiset histogrammit ovatkin näkyvästi esillä työn kokeellisessa osassa. Aluksi syötekuvien värihistogrammeja vertaillaan keskenään käyttäen niiden pohjalta laskettuja tilastollisia tunnuslukuja. Tämän jälkeen testataan erilaisten väriavaruuksien ominaisuuksia käyttäen väriarvojen 2-ulotteisia histogrammeja. Lopuksi RGB-värihistogrammeja käytetään vielä sellaisenaan piirrevektoreina itseorganisoivalle kartalle.

Kohdassa 2.4 kuvattujen laatumittojen soveltuvuutta on myös syytä kokeilla tässä yhteydessä. Niiden edustajaksi valittiin laatumitta spektrikuvien tiivistykselle, jota näin ollen siis sovellettiin testiaineistona käytetyille syötekuville. Laatumittaan avulla muodostetut piirrevektorit annettiin itseorganisoivan kartan jäsennehtäviksi.

Myös kohdan 2.5 tekstuuripiirrettä kokeiltiin samaan tapaan kuin spektrikuvien laatumittaan. Tässä yhteydessä tyydyttiin tekstuuripiirteen hieman yksinkertaistettuun versioon, jolloin kontrastipiirre jätettiin huomioimatta ja käytettiin pelkkiä ikkunoiden LBP-arvoja.

Lopuksi itseorganisoiva kartta on laajalti käytössä tässä työssä. Sen avulla mm. muodostetaan uusi ns. 'vallitseva' väri -piirre, joka on kuvattu tarkemmin kohdassa 4.3. Kokeellisen osan lopussa moniulotteisia piirrevektoreita myös visualisoidaan käyttäen itseorganisoivaa karttaa.

3.2 TESTIMATERIAALI

Tutkimusta varten oli saatu yhteensä kahdeksan aikakauslehti-tyyppistä painotuotosta, jotka olivat sisältönsä puolesta identtisiä keskenään. Erot aikakauslehtien välille syntyivät niiden painamiseen käytetyistä paperilaaduista, sekä tämän lisäksi käytetystä painatustekniikasta. Testimateriaali käsitti kaikkiaan neljä erilaista painopaperilaatua ja kaksi erilaista painatustekniikkaa, jolloin kukin kahdeksasta aikakauslehdestä

edusti ainutlaatuista kombinaatiota näistä. Toisin sanoen kumpaakin painatustekniikkaa kohti oli neljä erityyppiselle paperille painettua versiota kyseisestä lehdestä. Alla olevasta taulukosta selviää kunkin lehden painatukseen käytetty tekniikka ja painopaperin laatu.

Lehti	Painopaperi	Painotekniikka
L1	kiiltävä	perinteinen
L2	kirkas	stokastinen
L3	himmeä	stokastinen
L4	ei-fluoresoiva	perinteinen
L5	ei-fluoresoiva	stokastinen
L6	himmeä	perinteinen
L7	kirkas	perinteinen
L8	kiiltävä	stokastinen

Taulukko 2: Aikakauslehdet, lehdistä käytetyt lyhenteet ja lehtien ominaisuudet.

Lehdissä esiintyvistä lukuisista valokuvista valittiin kolme erilaista ja mahdollisimman edustavaa otosta. Valitut kohteet olivat *olohuone*, *kakku* ja *parveke*. Näistä muodostettiin kolme kuvasarjaa, jotka sisälsivät neljä kuvaa kukin. Kuvasarjan kuvat siis esittivät samaa kohdetta, ainoastaan kukin yksilölliselle paperilaadulle painettuna. Kaikki valitut lehdet edustivat ns. perinteistä painatustekniikkaa, sillä tässä vaiheessa päätettiin jättää toinen – ns. stokastinen painatustekniikka – kokonaan vaille huomiota, mm. jotta parametrien määrä testiaineistossa voitiin tutkimuksen alkuvaiheessa pitää mahdollisimman pienenä.

Valitut testikuvat digitoitiin digitaalikameralla. Tuloksena saatiin mainitut kolme sarjaa (yhteensä 12 kappaletta) digitaalisia RGB-kolmikomponenttikuvia. Kukin näistä kuvista siis esitti tiettyä kohdetta (ts. aikakauslehden valokuvaa) sillä tavoin, kuin se esiintyy painettuna kullekin neljästä erilaisesta paperivaihtoehdosta perinteistä painotekniikkaa käyttäen.

Kaikki kuvat digitoitiin samankaltaisissa kuvausolosuhteissa. Kuvauksessa käytettiin 180 millisekunnin valotusaikaa. Mahdollisten kamerasta ja digitointitekniikasta johtuvien virheiden minimoimiseksi kuvausetaisyys pyrittiin valitsemaan siten, että yksi digitaalikuvaan pikseli sisältää informaatiota, joka on peräisin useammasta kuin yhdestä painokuvan rasteripisteestä. Toisaalta kuitenkin tällä tavoin muodostuvan kuvien paikkaresoluution tuli olla riittävän suuri tutkimuksen tarpeisiin. Digitoitujen kuvien tallentamiseen käytettiin häviötöntä *TIFF*-tiedostomuotoa. Kuvatun digitointiprosessin lopputuloksena saatuja kuvia käytettiin sitten tutkimuksen aikana syötteenä joukolle erilaisia analysointimenetelmiä. Näihin digitaalisiin kuviin viitataan jatkossa termillä *syötekuva*, tai vielä lyhyemmin ainoastaan termillä *kuva* silloin, kun sekaannuksen vaaraa ei ole.

Digitoinnin jälkeen syötekuvat vielä leikattiin kuvankäsittelyohjelmalla saman kokoisiksi. Kuvien paikkaresoluutioksi muodostui 416*416 pikseliä. Käytetty väriresoluutio oli puolestaan 24-bittinen, joka näin ollen siis mahdollisti kunkin pikselin värin valitsemisen noin 16 miljoonan eri värisävyn joukosta.

Lisäksi jokaisesta kuvasta muodostettiin neljänneskokoinen pienennös (kooltaan 208*208 pikseliä) käyttäen pienentämiseen keskiarvosuodinta. Näitä pienempiä kuvaversioita käytettiin isompien vastineidensa sijasta eräissä laskennoissa. Kuvat lisäksi jaoteltiin toisia menetelmiä varten tasakokoisiin osakuviin eli ikkunoihin. Kun yhden ikkunan kooksi valittiin 32*32 pikseliä, kukin syötekuva jakautuu tasaisesti 13*13 ikkunaan. Näin ikkunoita siis muodostuu yhteensä 169 kappaletta. Kuvassa 2 on esitetty *olohuone*-kuvasarjan ensimmäinen lehti ja sen ikkunarakenne. Kaikki syötekuvat noudattivat vastaavaa ikkunarakennetta. Kahden muun kuvasarjan (*kakku* ja *parveke*) ensimmäinen lehti on esitetty liitteessä 3a ja 3b.



Kuva 2: *Olohuone*-kuvasarjan ensimmäinen kuva ja sen jako ikkunoihin.

Kaikki seuraavassa luvussa esitetyt tulokset on saatu käyttämällä edellä esiteltyä materiaalia syötteenä analysointimenetelmille. Tulokset laskettiin Matlab-ohjelmistolla.

4 Kokeelliset tulokset

Tässä luvussa esitetään yhteenveto tutkimuksen aikana saavutetuista kokeellisista tuloksista. Tulokset on organisoitu alilukuihin sen perusteella, millaista tekniikka niiden aikaansaamiseksi on käytetty.

Luvun ensimmäinen kohta 4.1 ei kerro minkään varsinaisen menetelmän avulla saaduista lopputuloksista. Sen tarkoituksena on ennemminkin tutustuttaa lukija syötedataan ja sen perusominaisuuksiin esittämällä syötedatasta irrotettuja erilaisia yksinkertaisia tunnuslukuja. Kohdassa 4.2 on puolestaan esimerkkejä 2-ulotteisista histogrammeista jotka saadaan, kun syötekuvien väri-informaatio on muunnettu erilaisiin tunnettuihin väriavaruuksiin.

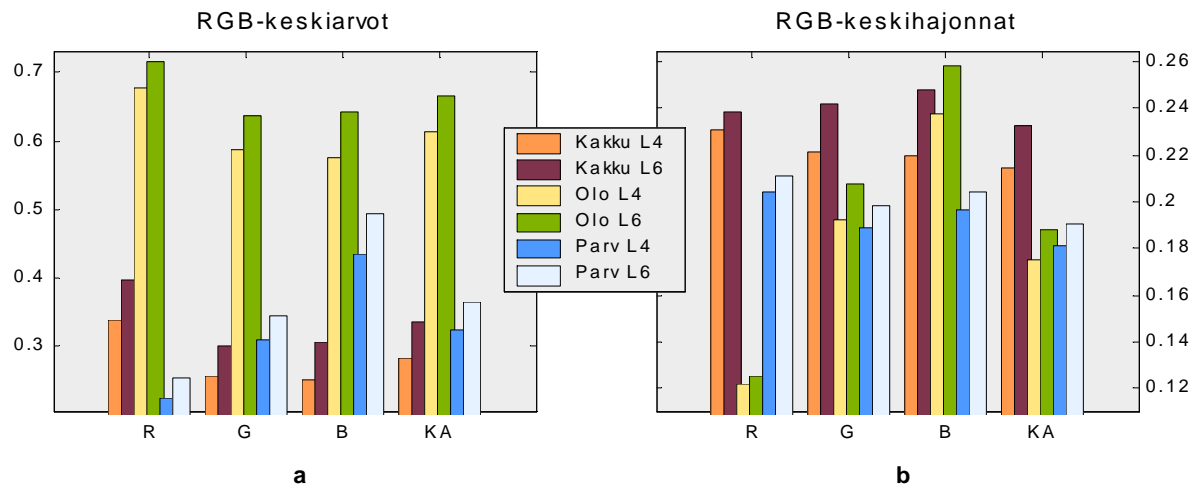
Kohdassa 4.3 on esitetty tuloksia, jotka on saatu käyttämällä itseorganisoivaa karttaa apuna eräänlaisen 'vallitsevan' värisävyn irrottamiseksi syötekuvista. Menetelmän avulla voidaan laskea sellaisia väriarvoja, jotka kuvaavat tietyllä tavalla kuvassa vallitsevaa yleisilmettä. Vertaamalla eri papereille painettujen, samaa kohdetta esittävien kuvien vastaavia arvoja, saatetaan saada viitteitä painopaperin vaikutuksesta kyseiselle kohteelle.

Lopuksi kohdassa 4.4 esitetyt tulokset on saatu käyttäen monipuolisinta menetelmien valikoimaa. Yhdistävä tekijä näille on itseorganisoivan kartan käyttö tulosten havainnollistamisessa. Kukin menetelmä antaa lopputuloksena piirrevektoreita, jotka kuvastavat syötekuvia kukin omalla tavallaan. Koska nämä piirrevektorit omaavat tyypillisesti jopa satoja ulottuvuuksia, niiden keskinäinen vertailu ilman apuvälineitä olisi hyvin hankalaa ja epähavainnollista. Tästä syystä itseorganisoivan kartan organisointivoimaa onkin käytetty hyväksi tulosten visualisoimisessa.

4.1 TILASTOLLINEN ANALYYSI

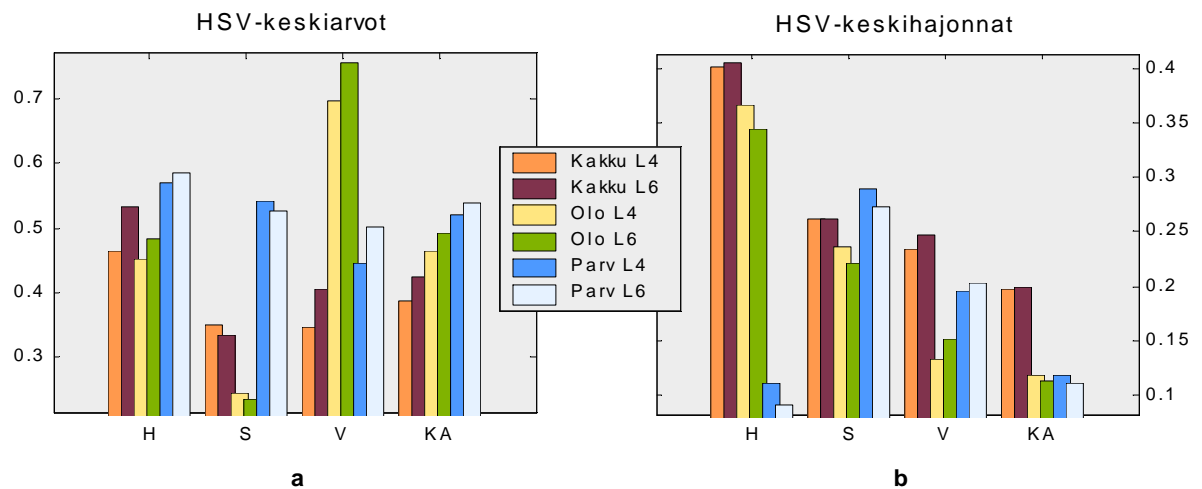
Tutkimuksen alkuvaiheessa kuvista pyrittiin irrottamaan ainoastaan yksinkertaisia piirteitä. Näiden perustana ovat pikselien intensiteetti-arvoista laskettavat erilaiset tilastolliset tunnusluvut. Esimerkki tämän kaltaisista yksinkertaisista mitoista ovat erilaiset histogrammit. Tunnuslukujen avulla pyritään tutustumaan syötedatan eri ominaisuuksiin ja tilastollisiin peruspiirteisiin. Tilan säästämiseksi seuraavassa esitetään tuloksia kustakin kolmesta kuvasarjasta ainoastaan lehdistä 4 ja 6 (kuvaajissa merkinnät *L4* ja *L6*), joiden välinen ero on silmämääräisesti kaikkein suurin.

Kuvassa 3 on näille kahdelle lehdelle painettujen kuvien pikselien RGB-arvoista sellaisenaan tuotettuja yksinkertaisia piirteitä. Kolme ensimmäistä pylvässarjaa esittää punaisen, vihreän ja sinisen kanavan pikselien intensiteettien keskiarvoja (kuva 3a) kussakin syötekuvassa. Neljäs pylvässarja esittää koko syötekuvan kaikkien kolmen kanavan intensiteettien keskiarvoa. Nämä arvot kertovat eri värikanavien kokonaisenergiasta. Alla olevista arvoista voidaan esimerkiksi päätellä, että *olohuone*-kuva on kaikkein 'valoisin'. Kuvassa *kakku* punaista väriä on suhteellisesti enemmän, kun taas kuvassa *parveke* sininen on vallitseva sävy. Lisäksi kuvassa 3b on esitetty intensiteetti-arvojen kanavakohtaiset keskihajonnat ja näiden keskiarvot. Nämä kertovat siitä, kuinka paljon hajontaa intensiteetti-arvojen kesken kuvissa esiintyy.



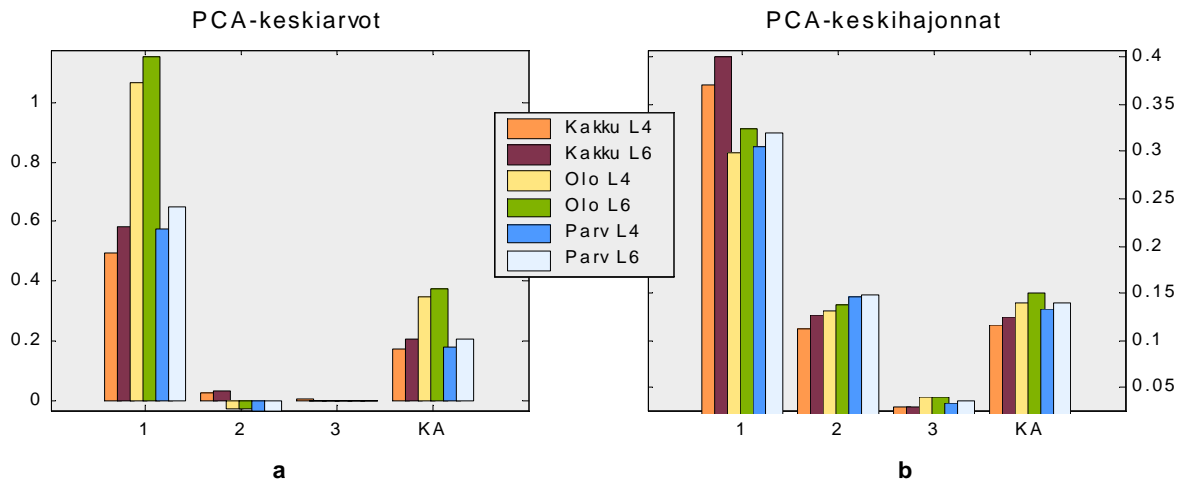
Kuva 3: RGB-intensiteettiarvojen keskiarvot (a) ja -hajonnat (b) kuudessa esimerkkikuvassa.

Kuvassa 4 on vastaavat arvot kuin edellä, mutta nyt sillä erotuksella, että syötekuvien intensiteetti-informaatio on muunnettu intuitiivisemmin tulkittavissa olevaan HSV-väriavaruuteen.



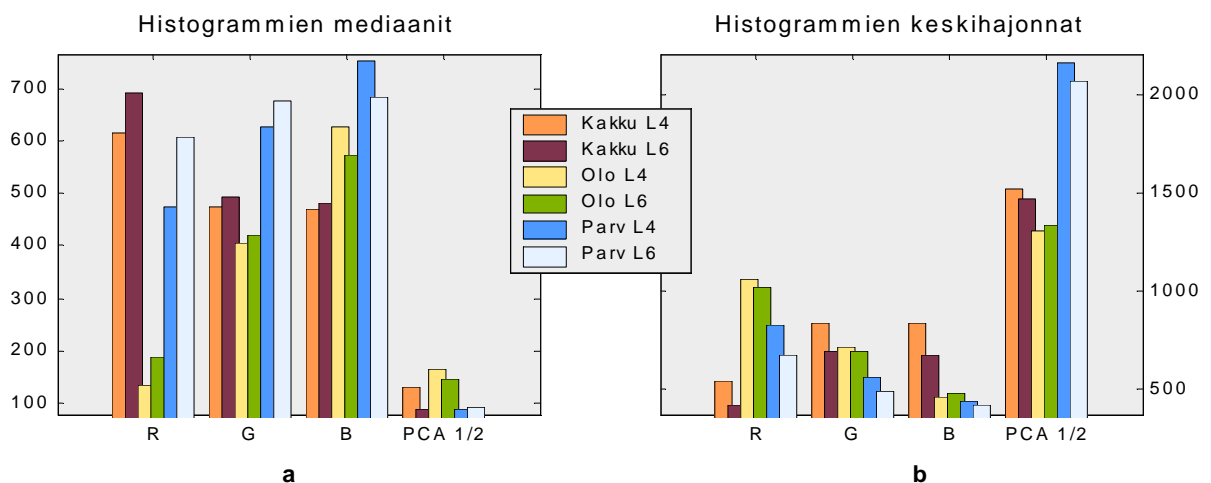
Kuva 4: Intensiteettiarvojen keskiarvot (a) ja -hajonnat (b) HSV-avaruudessa.

Kuvassa 5 on edelleenkin vastaavat tunnusluvut kuin edellä, mutta nyt kuva on muunnettu pääkomponenttianalyysin (PCA) avulla pääkomponenttiesitykseksi. Pääkomponenttianalyysissä RGB-kuutio ikään kuin pyöritetään sellaiseen sopivaan asentoon, jossa ensimmäinen komponenttikuva pitää sisällään eniten informaatiota, toinen seuraavaksi eniten ja vastaavasti viimeinen on merkitykseltään kaikkein vähäisin.



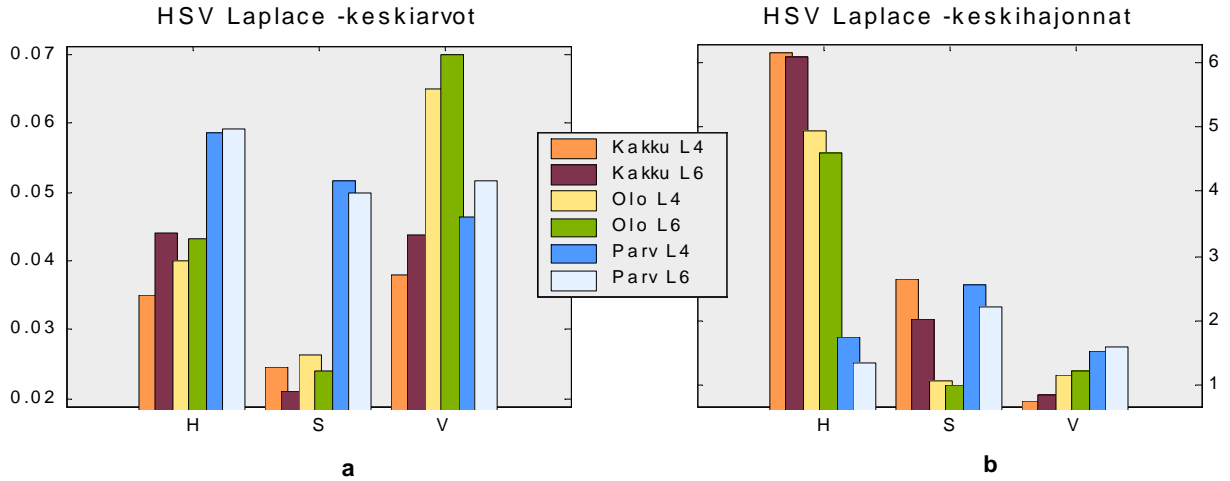
Kuva 5: Intensiteettiarvojen keskiarvot (a) ja -hajonnat (b) PCA-muunnoksen jälkeen.

Kuvassa 6 on erilaisten histogrammien pohjalta tuotettuja kuvaajia, jossa arvoina esitetään kunkin histogrammin mediaani (6a) ja keskijakonta (6b). Yleisesti histogrammit kuvaavat intensiteettiarvojen globaalia jakaumaa kuvissa. Alla kolme ensimmäistä pylvässarjaa edustaa syötekuvien RGB-histogrammien pohjalta tuotettuja arvoja. Neljäs pylvässarja puolestaan esittää erityisen PCA-suhdekuvan histogrammin vastaavat arvot. Tämä suhdekuva lasketaan jakamalla pikseleittäin kunkin syötekuvan toisen pääkomponenttikanan intensiteettien itseisarvot ensimmäisen kanavan vastaavilla arvoilla.



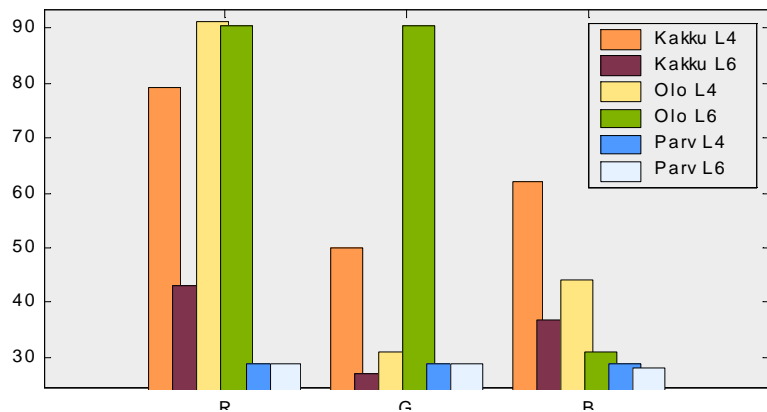
Kuva 6: Eri histogrammien pohjalta tuotettuja tunnuslukuja.

Kuvassa 7 esiintyvät arvot on poimittu syötekuvista, jotka on ensin käsitelty Laplace-reunasuotimella. Nämä arvot kertovat syötekuvien kontrastista ja reunaisuudesta.



Kuva 7: Reunasuotimella käsitellyistä HSV-kuvista irrotettuja tunnuslukuja.

Fourier-muunnoksen avulla alkuperäisen kuvan signaali-informaatio voidaan muuntaa taajuuskomponenttiesitykseksi. Tällä tavoin voidaan mitata eri taajuuskomponenttien läsnäoloa kuvassa. Tätä ominaisuutta käytetäänkin hyväksi monissa kuvankäsittelysovelluksissa. Viimeisen kuvan 8 arvot on laskettu syötekuvista komponenteittain suoritetun Fourier-muunnoksen jälkeen. Arvot vastaavat sellaisen suurimman energiapiikin Euklidista etäisyyttä kuvan keskipisteeseen, joka sijaitsee vähintään tietyn kynnyksarvon (10 pikseliä) päässä kuvan keskipisteestä.

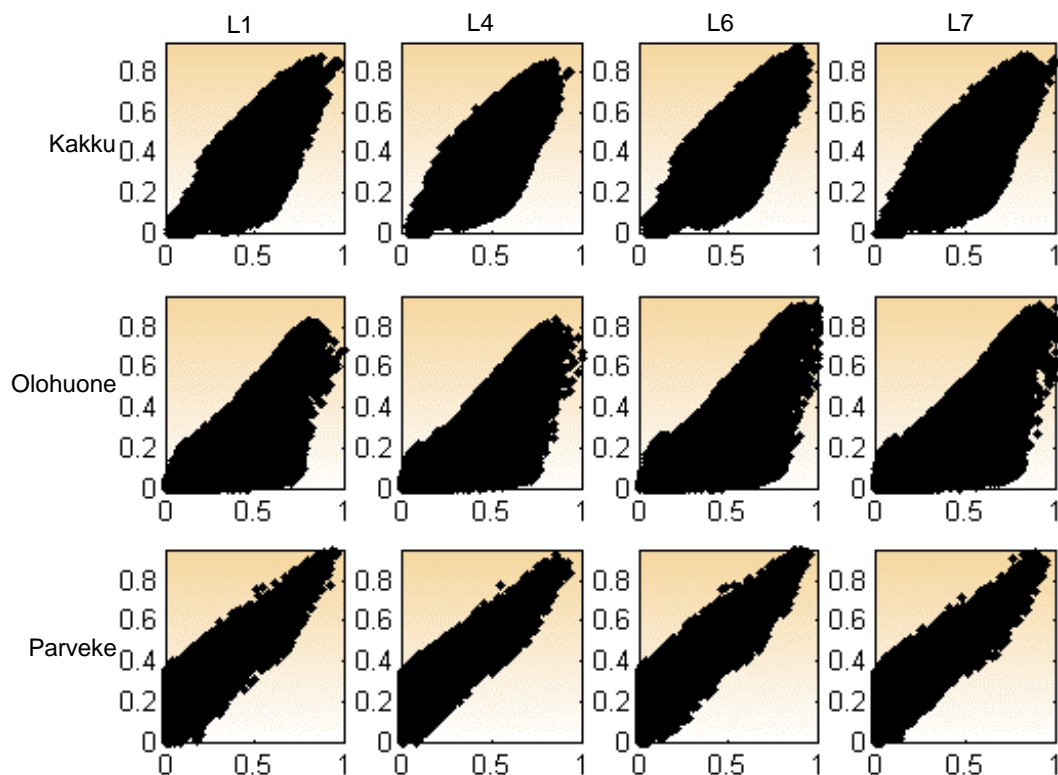


Kuva 8: Fourier-huippuarvojen etäisyyksiä keskipisteestä syötekuvissa.

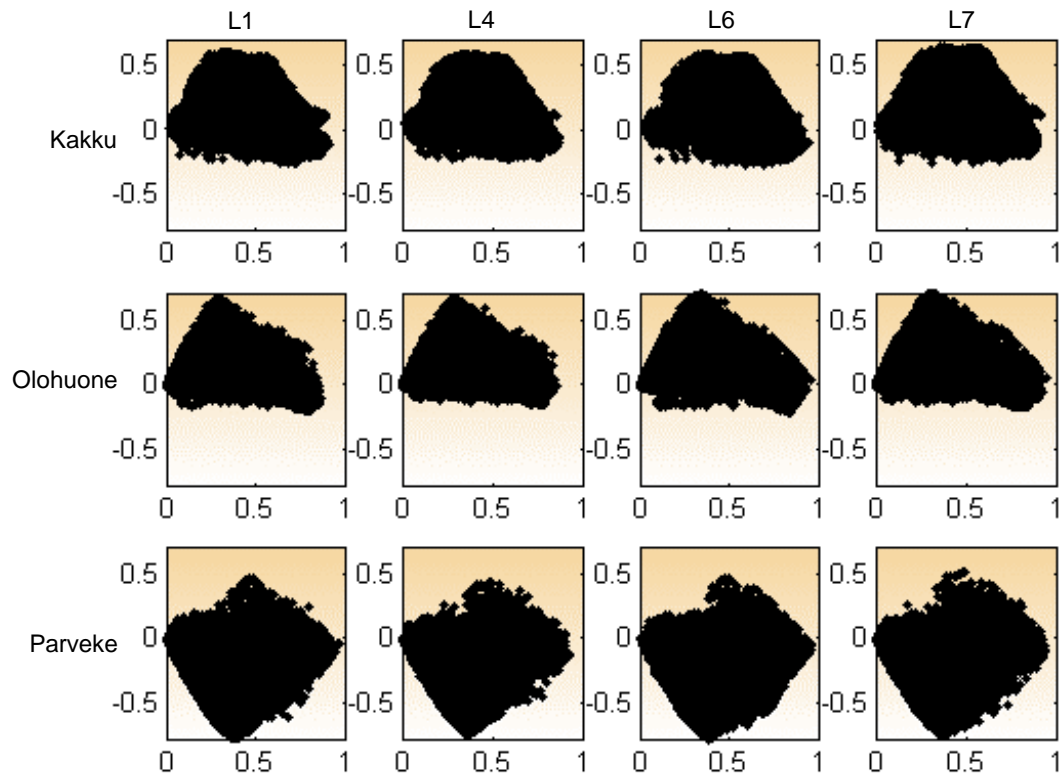
4.2 VÄRIAVARUUSPROJEKTIOT

Tutkimuksessa haluttiin testata myös erilaisten väriavaruuksien käyttökelpoisuutta syötekuvien ominaisuuksien tulkitsemisessa. Tätä varten kuvien värikanavien sisältämästä informaatiosta muodostettiin eräänlaisia 2-ulotteisia histogrammeja käyttäen värien määrittelyyn lukuisia erilaisia tunnettuja väriavaruuksia. Käytetyt väriavaruudet olivat *RGB*, *Ohta*, *YCbCr*, *NCC*, *Log*, *PIP2*, *HSV*, *AB* ja *TSL*. Useimmat näistä väriavaruuksista määritellään 3-ulotteisen koordinaatiston avulla, niinpä seuraavassa esitetyt kuvat ovat käytännössä näiden avaruuksien tasoprojektioita.

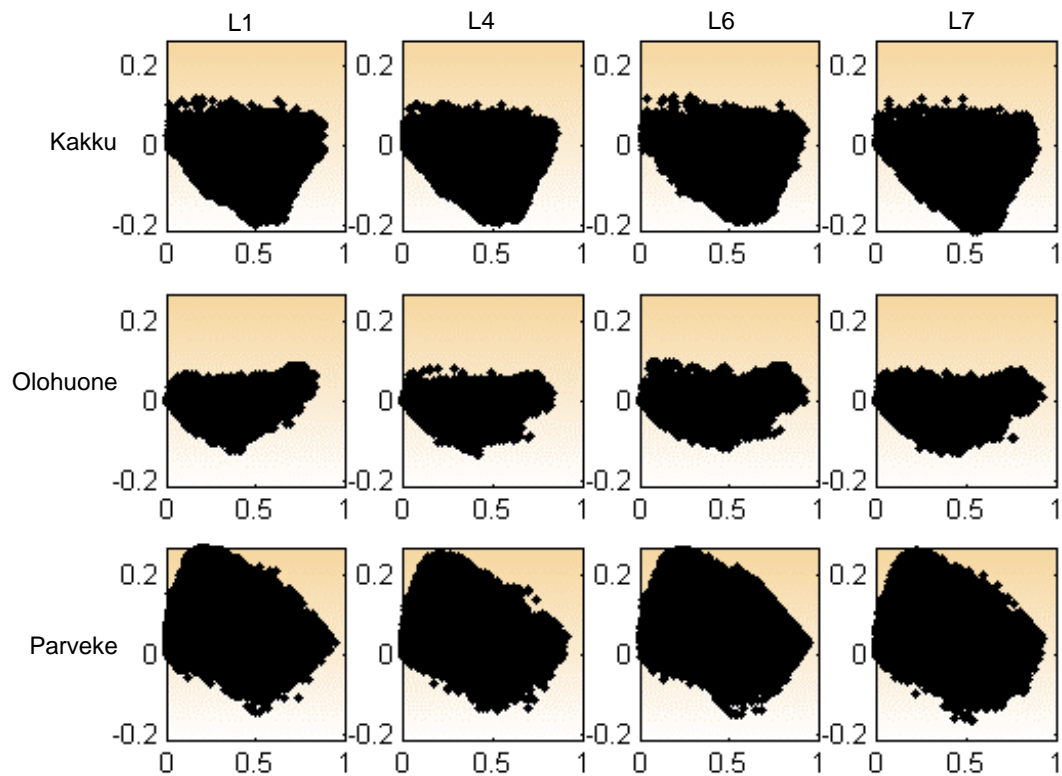
Ensimmäiseksi syötekuvien väri-informaatio on kuvassa 9 esitetty RGB-avaruuden poikkileikkauksena R/G-tasossa. Kuvassa 10 on vastaava esitys, mutta tässä syötekuvien väri-informaatio on ensin muunneltu *Ohta*-väriavaruuteen. Viimeinen kuva 11 on edelleen vastaava, mutta tällä kertaa *YCbCr*-väriavaruudessa.



Kuva 9: RGB-väriavaruusprojektiot R/G-tasossa kaikissa 12 syötekuvassa.



Kuva 10: *Ohta*-väriavaruusprojektiot kahden ensimmäisen värikomponentin suhteen.



Kuva 11: *YCbCr*-väriavaruusprojektiot Y/Cb -tasossa.

Loput väriavaruusprojektiokuvat (mukaan lukien edellä esitettyjen ortogonaaliprojektiot) löytyvät liitteestä 6.

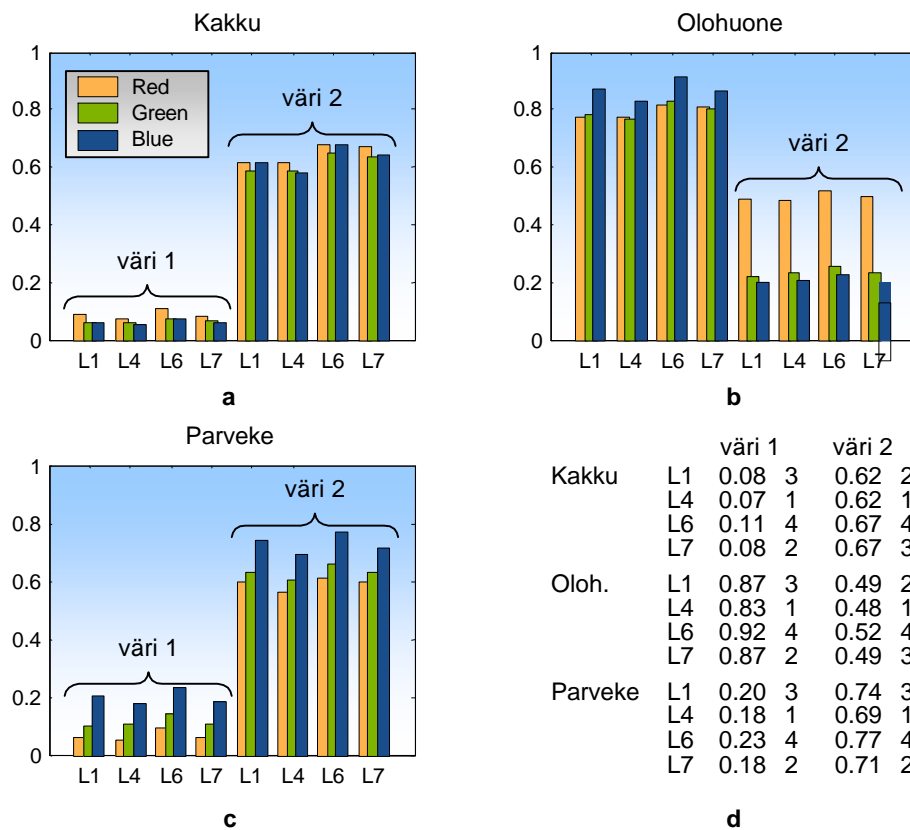
4.3 VALLITSEVA VÄRI

Tässä kohdassa esitellään tuloksia, jotka on saatu sellaisen menetelmän avulla, jota ei sellaisenaan löydy kirjallisuudesta (sen sijaan toisenlainen menetelmä visuaalisesti dominoivan värin määrittämiseksi löytyy esimerkiksi lähteestä [24]). Tässä menetelmässä syötekuvista irrotetaan piirre, joka on intuitiiviselta merkitykseltään eräänlainen 'vallitseva' väri. Toisin sanoen sellainen väri, joka on kullekin kuvalle tietyllä tavalla ominainen. Tuloksena saadut väriarvot kuvaavat syötekuvassa vallitsevaa yleisilmettä. Näiden väriarvojen keskinäinen vertailu saattaa myös ilmentää painopaperin vaikutusta syötekuvien välillä.

Piirteen laskentaprosessi yhdelle syötekuvalle suoritetaan seuraavalla tavalla. Ensin kuva jaetaan tasaisesti $a \times a$ pikselin kokoisiin ikkunoihin (a on pienekö kokonaisluku, tässä arvoksi valittiin $a=4$). Seuraavaksi lasketaan ikkunoiden sisältämien pikselien intensiteettien keskiarvot, jotta päästään eroon mahdollisista pienistä esimerkiksi digitointimenetelmästä johtuvista virheistä. Nämä keskiarvot edustavat kyseisiä ikkunoita jatkossa. Tuloksena saadaan m kappaletta 3-ulotteisia vektoreita, jotka sen jälkeen annetaan syötteenä itseorganisoivalle kartalle. Kun kartta on opetettu näillä arvoilla, siitä valitaan k kappaletta eniten osumia saanutta solua. Kuten muistetaan luvusta 2.6, syötedatan vektorit eivät kuvautu itseorganisoivan kartan soluille tasaisesti, vaan samankaltaiset syötteen kuvautuvat lähelle toisiaan. Tämän jälkeen käydään läpi tämä solujen joukko yksi kerrallaan. Oletetaan, että tarkastelun kohteena on i :nneksi eniten osumia saanut solu r_i ($i=1,2,\dots,k$). Tarkastellaan nyt niitä syötevektoreita, jotka kuvautuivat kartan opettamisen jälkeen tähän kyseiseen soluun ja lasketaan lopuksi näiden alkuperäisten syötteen keskiarvovektori. Näin tuloksena saadaan sellainen väriarvo, joka on approksimaatio niistä toisiaan muistuttavista väreistä, joita alkuperäisessä kuvassa esiintyi i :nneksi eniten. On huomattava, että kyseistä värisävyä ei välttämättä sellaisenaan esiinny alkuperäisessä

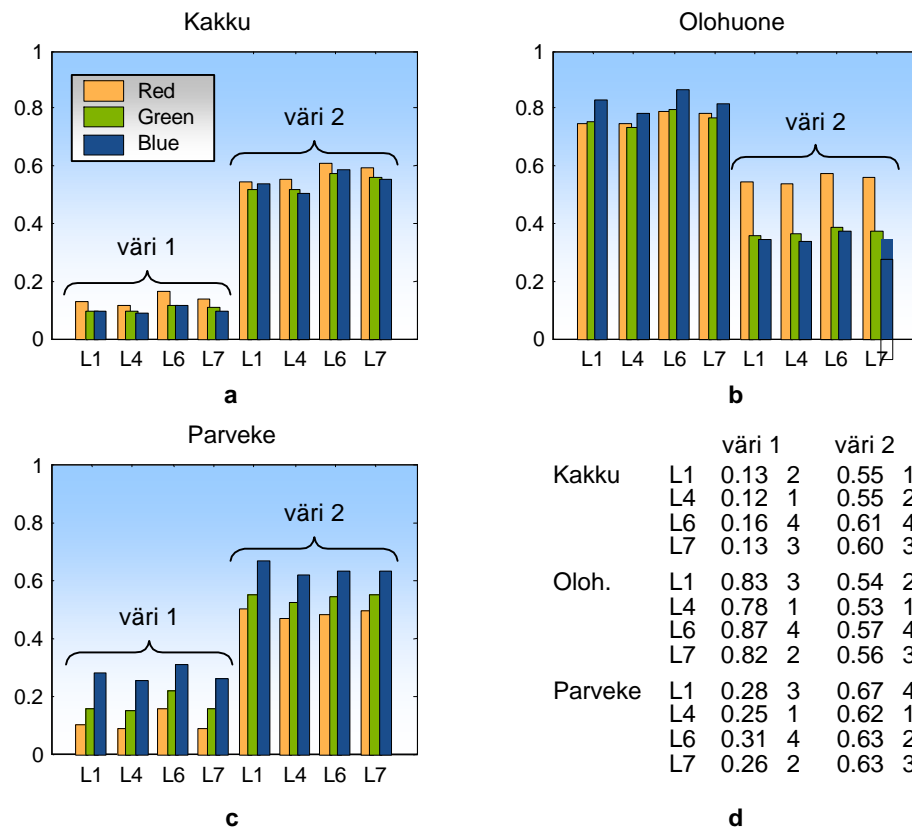
kuvassa, vaan kyseessä on ennemminkin yhdistetty kuvaus toisiaan muistuttavista, usein esiintyvistä värisävyistä.

Kuvassa 12 on edellä kuvatun menetelmän avulla tuotettuja intensiteetti-arvoja syötekuville (a, b ja c). Kustakin syötekuvasta määritettiin kaksi 'vallitsevinta' väriä. Laskennassa apuna käytetty itseorganisoiva kartta koostui kuusikulmio-topologiaa noudattavista 4*4 solusta. Tulokset on esitetty RGB-väriarvoina. Kuvan oikeassa alakulmassa sijaitsevassa taulukossa (kuva 12d) on lisäksi esitetty jokaisen syötekuvan kummankin vallitsevimman värin maksimi-intensiteetti ($I_{maks} = \max(R,G,B)$), sekä kuvasarjan sisäinen keskinäinen järjestys kyseisten arvojen osalta. Huomion arvoista on säännönmukaisuus, joka on selvästi erotettavissa eri kuvasarjojen kesken.



Kuva 12: Syötekuvista irrotetut kaksi 'vallitsevinta' väriarvoa RGB-avaruudessa kullekin kuvasarjalle (a, b ja c) sekä kyseisten väriarvojen kuvakohtaiset maksimi-intensiteetit ja näiden keskinäinen järjestys (d).

Myös itseorganisoiva kartta sellaisenaan muodostaa abstraktin kuvauksen syötedatasta. Kuva 13 on saatu aikaan samaan tapaan kuin edellä, mutta tällä kertaa sillä erotuksella, että kuvaajassa esiintyvät arvot on poimittu suoraan opetetun kartan malleista. Toisin sanoen, tiettyyn kartan soluun kuvautuneiden syötearvojen keskiarvoa ei tässä tarvita, vaan eniten osumia saaneiden solujen sisältämät mallit on poimittu sellaisenaan edustamaan kyseisiä syötteitä. Tulokset ovat samankaltaisia kuin edellä.



Kuva 13: Syötekuvien väriarvojen perusteella muodostetut kaksi yleisintä itseorganisoivan kartan mallia kullekin kuvalle (a, b ja c), sekä näiden arvojen maksimi-intensiteetit ja keskinäinen järjestys (d).

4.4 SOM-ANALYYSI

Tässä kohdassa erilaisia kuvista irrotettuja piirteitä on pyritty jäsentämään itseorganisoivaa karttaa käyttäen. Syötekuvista irrotettavat piirteet ovat tyypillisesti hyvinkin moniulotteisen avaruuden vektoreita, jolloin niiden välisiä suhteita voi olla vaikea hahmottaa. Niinpä tähän ongelmaan on pyritty vastaamaan antamalla piirrevektorit itseorganisoivan kartan

järjesteltäviksi. Sen jäsen- ja visualisointivoima pääsee tämän kaltaisessa tilanteessa oikeuksiinsa.

Seuraavassa käytetyt piirteet ovat syötekuvien RGB-histogrammit, RGB-väriarvot sellaisenaan, kohdan 2.5 tekstuuripiirre (LBP) sekä kohdassa 2.4.4 esitelty spektrikuvien virhemitta.

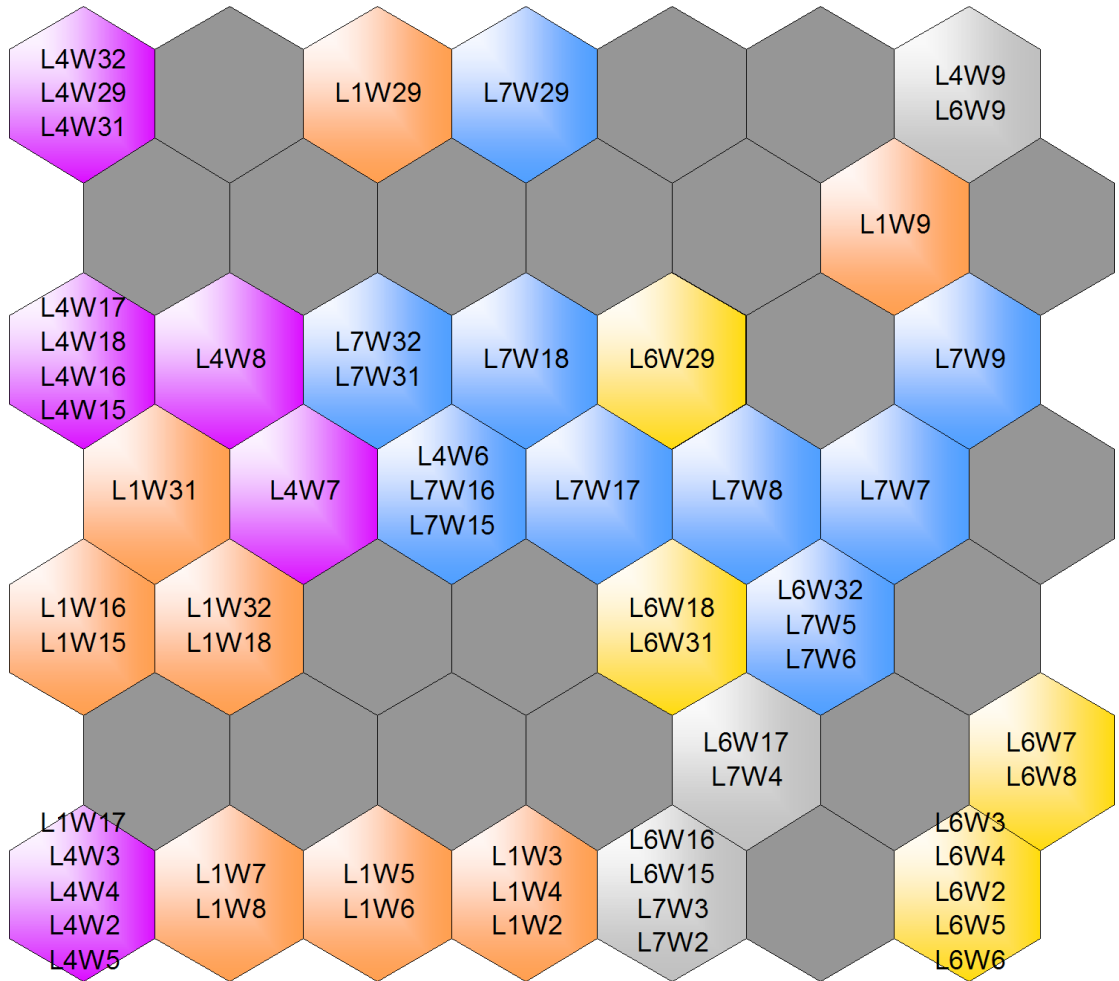
Kuten kohdassa 3.2 todettiin, kukin kuvasarja sisälsi neljä kuvaa. Itseorganisoivan kartan opettaminen ainoastaan neljää syötevektoria käyttäen ei kuitenkaan juurikaan ole mielekästä, niinpä laskentaa varten syötekuvat jaettiin tasaisesti 32×32 pikselin kokoiisiin ikkunoihin. Ikkunoita muodostuu tällä tavalla kaikkiaan 169 kappaletta (kuvissa indeksit 0..168). Näistä jatkotarkastelun kohteeksi valikoitiin kuitenkin vain 15 kappaletta kaikkein kirkkaimpia. Kirkkaus määritettiin ikkunan alueella esiintyvien pikselien intensiteettien kokonaisenergian perusteella, käyttäen kunkin kuvasarjan ensimmäistä lehteä indeksien määräämiseen. Käsittelyn rajaamista koskemaan vain kirkkaimpia kuvan osia voidaan perustella sillä, että näissä kohdissa painomusteen vaikutus on suhteellisesti vähäisempi ja siten painopaperin osuus on paremmin havaittavissa.

Ideaalitapauksessa syötteenä käytetty kuvista irrotettu piirre olisi sellainen, että eri lehdiltä – toisin sanoen eri painopaperilaaduilta – peräisin olevat piirrevektorit kuvautuisivat kartassa mahdollisimman etäälle toisistaan. Näin ollen mikäli kartassa esiintyy valtaosa sellaisia soluja, jotka sisältävät useilta eri papereilta peräisin olevia piirrevektoreita, on tulos tulkittava huonosti onnistuneeksi. Tämän ominaisuuden hahmottamista helpottamaan itseorganisoivan kartan solut on seuraavissa visualisoinneissa värjätty eri värein silloin, kun solussa esiintyy enemmistö samalta lehdeltä peräisin olevia syötteitä.

Edellä kuvattujen alkuehtojen vallitessa kukin itseorganisoiva kartta opetettiin siis 15×4 piirrevektorin perusteella. Kartan koko oli kaikille käytetyille piirteille sama, 7×7 solua. Myös tämän luvun itseorganisoivien

karttojen laskennassa käytettiin kuusikulmio-topologiaa. Seuraavassa esitettävät kuvat visualisoivat kartan tilaa opettamisprosessin päätyttyä.

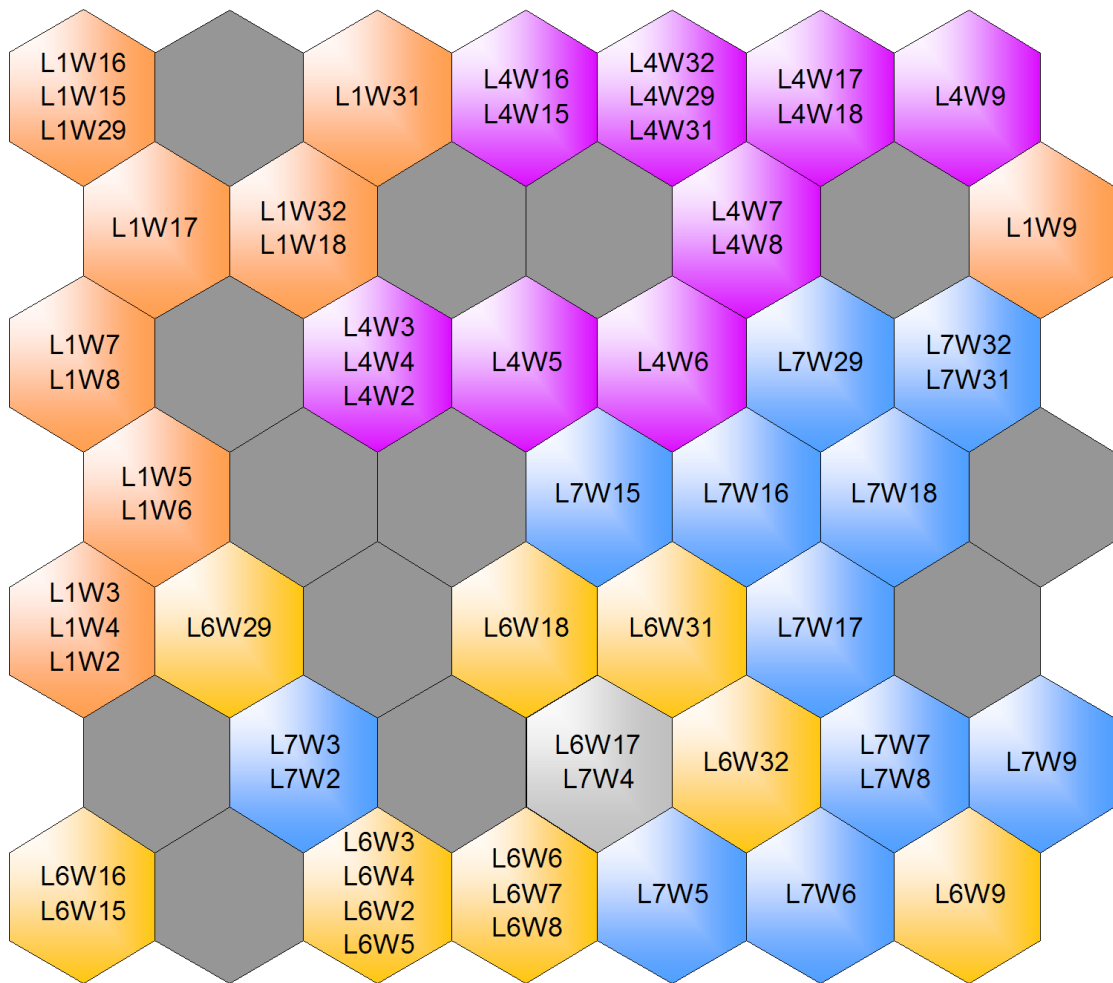
Kuvassa 14 on olohuone-kuvan ikkunoiden histogrammien perusteella muodostettu itseorganisoiva kartta. Kartan soluilla esiintyvät symbolit ovat muotoa $LxWy$, jossa x on lehden numero ($x \in \{1,4,6,7\}$) ja y on ikkunan indeksi ($y \in [0,168]$).



Kuva 14: Itseorganisoivan kartan järjestelmät piirrektorit olohuone-kuvasta, kun piirteenä on käytetty ikkunoiden RGB-histogrammeja. Symbolit ovat muotoa $LxWy$, jossa x on lehden numero ja y on ikkunan indeksi.

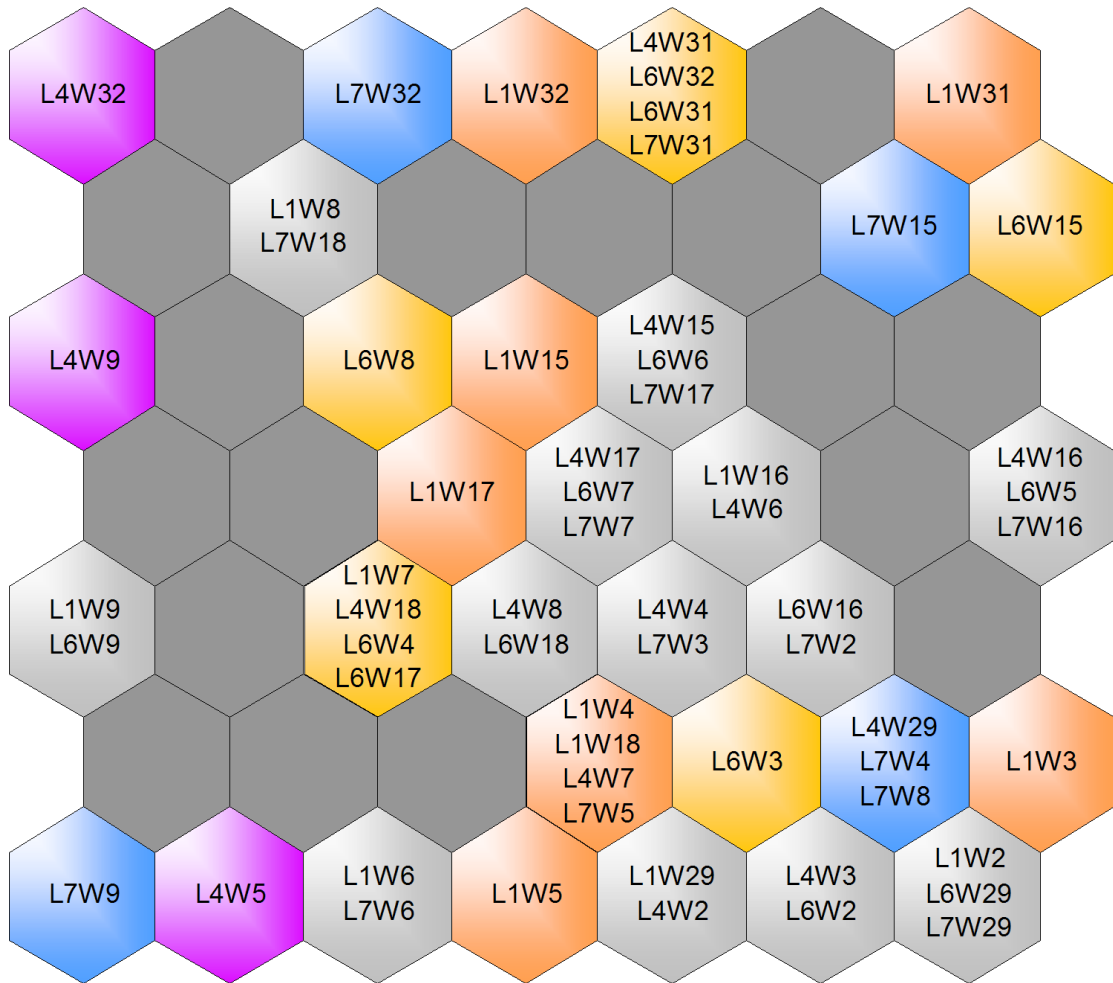
Kuvassa 15 on vastaava kuvaaja kuin edellä. Tällä kertaa piirrektoreina käytettiin pikselien intensiteettiarvoja sellaisenaan. Tässä yhteydessä

käytössä oli syötekuvien neljännespienennökset, jolloin ikkunan koko oli myös samassa suhteessa pienempi.



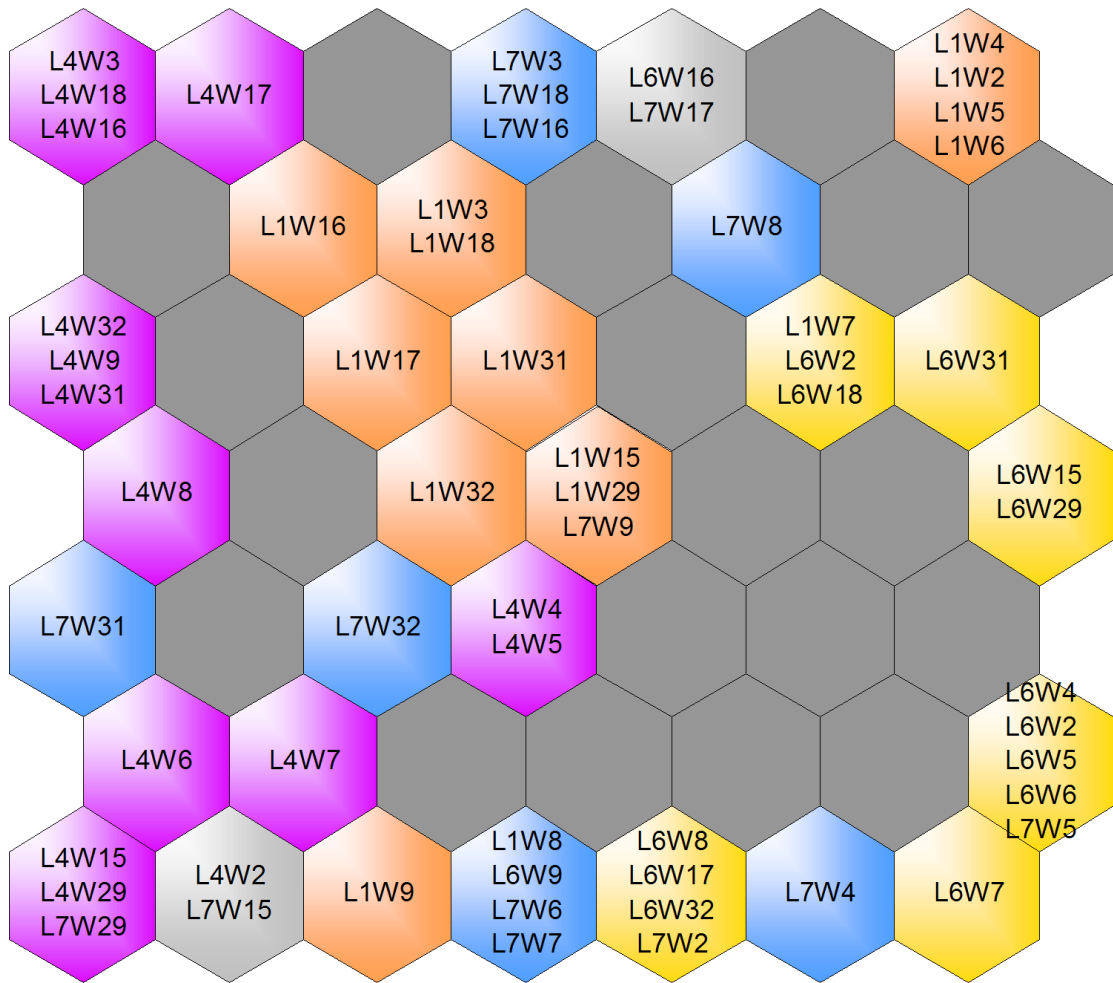
Kuva 15: Itseorganisoivan kartan järjestelmät piirrektorit *olohuone*-kuvasta, kun piirteenä on käytetty ikkunoiden pikselien RGB-intensiteettiarvoja. Symbolit ovat muotoa $LxWy$, jossa x on lehden numero ja y on ikkunan indeksi.

Histogrammit tai intensiteettiarvot sellaisenaan eivät huomioi syötekuvien spatiaalisia rakenteita. Niinpä seuraavassa kuvassa 16 piirrektoreina on käytetty LBP-tekstuuripiirteiden perusteella muodostettuja histogrammeja. Tekstuuripiirre muodostetaan pikselien naapurustojen perusteella, joten se tallettaa tietoa myös pikselien spatiaalisesta rakenteesta.



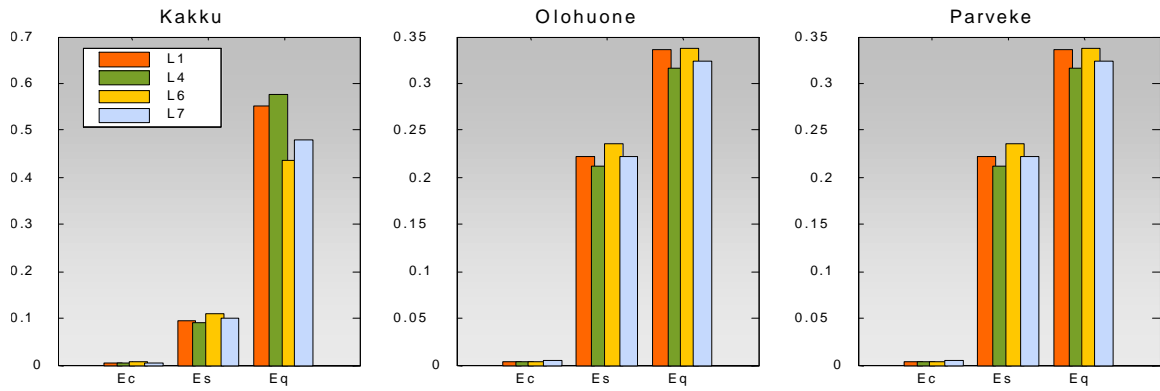
Kuva 16: Itseorganisoivan kartan järjestelmät piirvektorit *olohuone*-kuvasta, kun piirteenä on käytetty ikkunoiden LBP-tekstuuripiirteiden histogrammeja. Symbolit ovat muotoa $LxWy$, jossa x on lehden numero ja y on ikkunan indeksi.

Luvun lopuksi kuvassa 17 esitetään edellisten kaltainen kuvaaja, jossa tällä kertaa piirteenä on käytetty spektrikuvien laatumittaa. Syötteenä käytetyt RGB-kuvat on tässä yhteydessä ajateltu spektrikuvien erikoistapaukseksi, osin siitä syystä, että tutkimuksessa on tarkoitus jatkossa keskittyä juuri spektrikuvien analysointiin. Lisäksi mitan laskennassa normaalisti edellytetty referenssikuva on jätetty piirteen laskennassa huomiotta. Tätä referenssikuvaa ei nimittäin tässä tapauksessa voida määrittää järkevästi, sillä kuvasarjan sisällä esiintyy kuvien kesken mm. erilaisia siirtymä- ja rotaatiovirheitä, jotka vaikuttavat oleellisesti niiden keskinäiseen vertailuun. Siten referenssikuvaksi ei voida kiinnittää yhtäkään kuvasarjan kuvista ilman ilmeistä vaaraa tulokseen vaikuttavista, oleellisen suurista virheistä.



Kuva 17: Itseorganisoivan kartan järjestelmät piirvektorit olohuone-kuvasta, kun piirteenä on käytetty ikkunoiden virhemitta-arvoja (E_C , E_S ja E_K). Symbolit ovat muotoa $LxWy$, jossa x on lehden numero ja y on ikkunan indeksi.

Kuvassa 18 on vielä esitetty vertailun vuoksi saman spektrikuvien virhemitan tuottamat arvot kullekin syötekuvalle silloin, kun syötteenä on käytetty koko kuva-alaa edellä käytetyn kirkkaimpiin ikkunoihin jaottelun sijasta.



Kuva 18: Spektrikuvien virhemitan tuottamat arvot (E_C , E_S ja E_K) syötekuville.

5 Pohdinta

On syytä korostaa, että ne tutkimustulokset, joita edellä on esitetty, eivät tarjoa lopullista ratkaisua alkuperäiseen tutkimusongelmaan. Tätä tutkielmaa voidaan ajatella sellaisena perustana, joka voi toimia pohjana lisätutkimukselle. Tässä pohdintaluvussa nimetään joitakin tutkimuksen aikana kohdattuja ongelmia sekä sellaisia haasteita, joihin ei toistaiseksi olla täysin kyetty vastaamaan. Lisäksi luodaan katsaus saavutettuihin tuloksiin sekä siihen, mihin seikkoihin tutkimuksen jatkon kannalta olisi syytä kiinnittää erityistä huomiota.

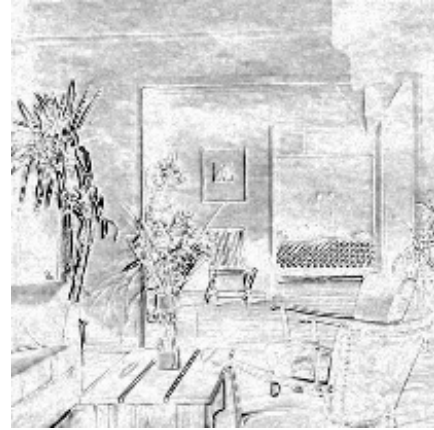
Ensimmäinen oleellinen kysymys ratkaistavaksi tutkimuksen lähtökohtia ajatellen on syötemateriaalin hankinta. Etukäteen ei voida tietää esimerkiksi sitä, syntyvätkö erot eri painopaperien välille korkean- vai matalan tason piirteissä. Kun kuvat digitoidaan digitaalikameralla, niistä muodostuu eräänlainen abstraktio, jonka tuloksia voidaan kyllä analysoida, mutta digitointivaiheessa tietoa kuitenkin väistämättä häviää. Jotain tärkeää voi tällöin yhtä hyvin jäädä myös kokonaan huomiotta. Käytännössä esimerkiksi ilmeni, että suorittamalla kuvaus liian läheltä aikakauslehden pintaa, saattaa lopputuloksessa esiintyä mm. niin sanottujen Moirè-kuvioiden kaltaisia, rasteroinnista aiheutuvia vääristymiä.

Olosuhteiden pakosta digitoinnissa jouduttiin tyytymään varsin karkean tason kuvaukseen alkuperäisistä aikakauslehden kuvista. Tällä seikalla – yhdistettynä siihen tosiasiaan, että lehtiin painetut kuvat ovat alun perin olleet täysin identtisiä keskenään – on merkittäviä seurauksia. Muun muassa digitoinnin tuloksena saatujen syötekuvien spatiaalinen rakenne on hyvin samankaltainen; pois lukien itse digitointiprosessista aiheutuneet virheet, jotka ovat tässä yhteydessä itse asiassa painopaperista ja painotekniikasta johtuvia eroja merkittävämpiä. Tästä välittömänä seurauksena onkin odotettavissa, että erilaiset pikselien spatiaalista rakennetta mittaavat piirteet (kuten tutkimuksessa käytetty tekstuuripiirre) tuottavat erittäin vaatimattomia tuloksia.

Edelliseen liittyen oleellinen seikka ymmärtää valittuja menetelmiä ja niiden avulla saatuja tuloksia tarkasteltaessa onkin se, että kuvasarjojen sisällä syötekuvat eivät suinkaan ole pikselitasolla suoraan vertailukelpoisia keskenään. Vaikka kuvat digitoitiin yhtäläisissä olosuhteissa, ne sisältävät silti väistämättä virheitä, joihin lukeutuvat erityisesti erilaiset rotaatio-venymä- ja siirrosvirheet. Tätä havainnollistetaan kuvassa 19. Sitä varten on laskettu kolme pikseleittäin otettua erotuskuvaa *olohuone*-kuvasarjan sisällä siten, että ensimmäisestä lehdestä digitoitua kuvaa on käytetty referenssikuvana johon muita on sen jälkeen verrattu. Kuten voidaan esimerkiksi suurin piirtein kuvan keskellä sijaitsevaa taulua tarkastelemalla havaita, erotuskuvat ovat keskenään hyvinkin erilaisia huolimatta siitä, että kuvasarjan alkuperäiset kuvat muistuttavat spatiaalisen rakenteensa puolesta toisiaan lähes identtisesti. Liitteessä 2 on asian varmentamiseksi esitetty kaikki neljä *olohuone*-kuvasarjan kuvaa sellaisenaan.



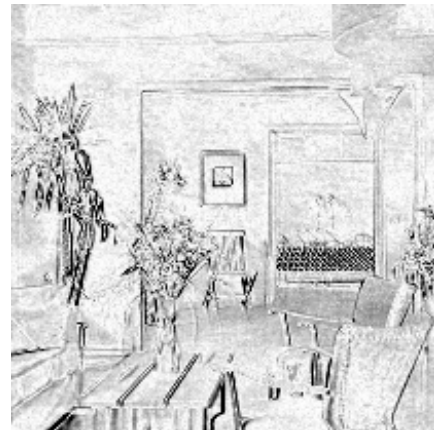
Lehti 1



Lehtien 1 ja 4 erotus



Lehtien 1 ja 6 erotus



Lehtien 1 ja 7 erotus

Kuva 19: *Olohuone*-kuvasarjan lehtien erotuskuvien negaatiot. Erotuskuvat on lisäksi skaalattu vakiolla, jotta eroavaisuudet saadaan selvemmin näkyviin.

Koska kuvasarjojen kuvat eivät näin ollen ollen koostu joukosta sellaisia vastinpikseleitä, joiden keskinäinen vertailu olisi mielekäästä, valtaosaa olemassa olevista kuvien laatua mittaavista tekniikoista ei voida sellaisenaan soveltaa tässä yhteydessä. Käytännössä tämän kaltaista laatumittaa haluttiin tässä tutkimuksessa kuitenkin testata, niinpä laatupiirteen laskennassa normaalisti tarvittava referenssikuva yksinkertaisesti jätettiin huomiotta. Toisin sanoen laskenta suoritettiin käsittelemällä kutakin syötekuvaa muista erillisenä kokonaisuutena. Käsitteellisesti on kuitenkin vaikeaa sanoa, mitä tämän kaltainen laatumitta ilman vertailusyötettä itse asiassa mittaa.

Eräs ongelma tämän kaltaisessa tutkimuksessa on myös piirteiden jäsentely. Tutkitut menetelmät tuottavat syötedatan perusteella piirrevektorin, jossa voi olla vaihteleva ja hyvinkin suuri määrä ulottuvuuksia. Piirteitä voidaan muodostaa joko globaalisti tai paikallisesti ikkunatasolla. Globaali lähestymistapa tuottaa yhden piirrevektorin kuvaa kohti, kun taas paikallinen lähestymistapa tuottaa suuremman piirrevektorien joukon. Paikalliset piirteet voidaan lopuksi joko yhdistää yhdeksi globaaliksi piirteeksi tai ne voidaan vaihtoehtoisesti myös käsitellä erikseen. Tässä työssä käytetyt syötekuvasarjat koostuvat neljästä kuvasta. Näin ollen globaalin piirteiden käyttö kuvasarjan kuville tuottaa tässä tapauksessa ainoastaan neljä piirrevektoria. Vähäisestä lukumäärästä huolimatta kyseisten piirrevektorien keskinäinen vertailu saattaa olla hyvinkin hankalaa, erityisesti silloin, kun nämä sisältävät lukuisia toisistaan riippumattomia ulottuvuuksia. Numeeriset arvot sellaisenaan ovat vaikeasti tulkittavissa, niinpä eri muuttujien välisten suhteiden hahmottaminen ja visualisointi on tässä yhteydessä oleellinen kysymys ratkaistavaksi. Toivottavaa olisikin, että lopullinen laatua mittaava piirre tuottaisikin kuvalle hyvin rajallisen määrän tunnusarvoja, mieluiten tietysti ainoastaan yhden.

Kokeellisia tuloksia tarkastelemalla nähdään, että eri painopaperit aiheuttavat selvästikin säännönmukaisia eroja niiltä digitoitujen kuvien kokonaisenergioihin. Toisin sanoen syötekuvien kokonaiskirkkauudessa on eroja, jotka voidaan yhdistää käytettyyn painopaperiin. Tämä käy ilmeiseksi esimerkiksi tutkimalla tilastollisen analyysin yhteydessä esitettyjä RGB- ja HSV-keskiarvoja (kuvat 3 ja 4). Sama ilmiö lienee myös 'vallitseva väri' -piirteiden erottelukyvyn takana. Tämä piirre luokittelee kuvassa esiintyvät värit ikään kuin kahdella tasolla: toisaalta sen avulla löydetään toisiaan läheisesti muistuttavat värisävyt, mutta lisäksi myös kyseisten värisävyjen esiintymistiheys kuvan alueella. Kun tarkastellaan kussakin kuvassa suhteellisesti eniten esiintyneitä sävyjä syötekuvasarjan sisällä, voidaan olettaa, että vallitsevimmat väriarvot ovat peräisin likimain vastaavilta

alueilta syötekuvissa. Tämä piirre onkin varsin vakaa esimerkiksi siinä mielessä, että se on invariantti mm. rotaatiovirheille ja skaalan vaihdoksille.

Kuvaa 12 tarkastelemalla voidaan todeta, että syötekuvien kahden yleisimmän värisävyn intensiteetit noudattavatkin selvää säännönmukaisuutta. Tietyltä painopaperilta peräisin olevat sävyt ovat toisia kirkkaampia. Kääntäen myös tietyltä paperilta peräisin olevat sävyt ovat muita tummempia. Sama ilmiö on selvästi havaittavissa kummankin yleisimmän värisävyn kohdalla.

Näin voitaisiin siis saada viitettä siitä, että painopapereita tehokkaasti toisistaan erotteleva piirre saattaisi liittyä pikselien globaaleihin intensiteettiarvoihin kuvissa. Muiden tilastollisten piirteiden, kuten kohdassa 4.1 esitettyjen hajontojen, PCA- tai Fourier-muunnoksen pohjalta laskettujen tunnuslukujen soveltuvuus voidaan puolestaan kyseenalaistaa. Erityisesti kuvassa 8 esitetyt merkittävimmän taajuuskomponentin esiintymistä mittaavat etäisyysarvot ovat lähes satunnaisen oloisia, eikä niiden perusteella syötekuvista voida sanoa juuri mitään. Tämä saattoi toisaalta olla odotettavissakin, sillä kuten todettu, syötekuvat eri kuvasarjojen sisällä muistuttavat spatiaalisen rakenteen puolesta voimakkaasti toisiaan. Tämä syötekuvasarjojen ominaisuus aiheuttaakin ongelmia erityisesti kuvahakupiirteiden ja tekstuuripiirteiden soveltamisessa. Nämä piirteet on suunniteltu sellaiseen tilanteeseen, jossa vertailtavat kuvat ovat keskenään varsin erilaisia. Näin ollen niiden suoriutuminen tässä tutkimuksessa vallitsevissa erityisolosuhteissa ei välttämättä tuota optimaalisia tuloksia.

Myös erilaiset väriavaruudet ja kohdassa 4.2 kuvatut 2-ulotteiset histogrammit osoittivat vain pieniä merkkejä käyttökelpoisuudesta. Niinpä ne jätettiin jatkossa varsin vähäiselle huomiolle.

Tarkastellaan seuraavaksi oheista taulukkoa 3. Siihen on koottu luvussa 4.4 ja liitteissä 4 ja 5 esitettyjen SOM-visualisointien pohjalta muodostettuja lukuarvoja. Arvot ovat prosenttilukuja, jotka kuvaavat tietyltä painopaperilta

peräisin olevan syötevektorin todennäköisyyttä kuvautua SOM-visualisoinnissa sellaiselle itseorganisoivan kartan solulle, joka sisältää suurimman osan samalta paperilta peräisin olevia syötteitä. Työssä käytetyt kolme kuvasarjaa ja neljä tutkittua piirrettä on esitetty. Taulukon arvoja ei tule ottaa absoluuttisena totuutena eri piirteiden suoriutumisen mittarina, mutta niiden avulla voi saada jonkinlaisen intuition asiasta. Optimaalinen piirre olisi sellainen, joka ryhmittelisi eri painopapereilta peräisin olevat syötteet mahdollisimman kauas toisistaan, jolloin itseorganisoivan kartan soluissa esiintyisi pääosin ainoastaan samalta painopaperilta peräisin olevia syötteitä. Tällöin siis taulukossa esiintyvät todennäköisyydet olisivat suuria, pääsääntöisesti lähellä sataa prosenttia.

Olohuone	L1	L4	L6	L7	KA
RGB-histogrammit	93	87	67	80	81
RGB-väriarvot	100	100	93	93	97
LBP-tekstuuripiirre	53	20	47	20	35
Virhemitta	20	93	87	60	65

a

Kakku	L1	L4	L6	L7	KA
RGB-histogrammit	47	33	47	27	38
RGB-väriarvot	27	20	33	27	25
LBP-tekstuuripiirre	47	0	47	33	32
Virhemitta	40	67	67	47	55

b

Parveke	L1	L4	L6	L7	KA
RGB-histogrammit	0	60	20	27	27
RGB-väriarvot	7	7	0	0	3
LBP-tekstuuripiirre	33	47	27	27	33
Virhemitta	53	27	47	27	38

c

Taulukko 3: SOM-visualisointien vertailua eri piirteiden ja kuvasarjojen kesken. Arvot ovat syötevektorien prosentuaalisia osumia sellaisiin itseorganisoivan kartan soluihin, jotka sisältävät suurimman osan samalta painopaperilta peräisin olevia syötteitä.

Tutkimalla *olohuone*-kuvasarjasta saatuja tuloksia (taulukko 3a) tässä valossa paljastaa, että RGB-arvoihin perustuvat piirteet toimivat siihen parhaiten. Tämän kuvan osalta tekstuuripiirre tuottaa odotetusti heikoimman tuloksen ja spektrikuvien virhemitta puolestaan

keskinkertaisen tuloksen. Tutkimalla *parveke*-kuvaa (taulukko 3c) kuitenkin huomataan, että esimerkiksi RGB-väriarvot piirrevektoreina ryhmittäytyivätkin lähes päinvastaisella tavalla. Sama ilmiö on havaittavissa myös *kakku*-kuvasarjan sisällä.

Vertailemalla piirteitä kaikkien kolmen kuvasarjan kesken havaitaan, että sekä tekstuuripiirteiden että virhemitan suoriutumisen oli kaikista huolimatta vakainta, vaikka kummankaan näistä – erityisesti tekstuuripiirteiden – avulla saadut tulokset eivät ole missään yhteydessä erityisen hyviä.

Yhteenvetona voidaankin sanoa, että erityisesti testimateriaalin ja myös testattujen menetelmien joukkoa olisi laajennettava, jotta voitaisiin tehdä pitemmälle meneviä johtopäätöksiä. Tähän asti suoritettujen tutkimusten perusteella yksikään menetelmä tai käytetyistä piirteistä ei tuota ylivoimaisia tuloksia muihin nähden. Toisaalta yhtään menetelmätyyppiä ei edellisen perusteella voida myöskään täysin rajata ulos tarkastelun piiristä. Jatkotutkimuksen kannalta menetelmiä olisikin selvästi kehitettävä edelleen. Näyttää hyvin todennäköiseltä, että parhaat tulokset on saavutettavissa vain erityisellä, painopaperianalyysiä varten varta vasten suunnitellulla menetelmällä.

Viiteluettelo

- [1] Adams, R.: *Calculus: a complete course*. Don Mills: Addison-Wesley, 1999.
- [2] Androutsos, D., Plataniotis, K. N. ja Venetsanopoulos, A. N.: *Distance Measures for Color Image Retrieval*. International Conference on Image Processing, Vol. 2, ICIP '98, 1998, s. 770–774.
- [3] Aslandogan, Y. A. ja Yu, C. T.: *Techniques and systems for image and video retrieval*. IEEE Trans. on Knowledge and Data Engineering, Vol. 11, No. 1, 1999.
- [4] Brunelli, R. ja Mich, O.: *On the Use of Histograms for Image Retrieval*. Proc. IEEE Multimedia Systems '99 International Conference on Multimedia Computing and Systems [IEEE ICMCS99], 1999, s. 143–147.
- [5] Cinque, L., Levialdi, S., Olsen, K. A. ja Pellicanó, A.: *Color-Based Image Retrieval Using Spatial-Chromatic Histograms*. IEEE International Conference on Multimedia Computing and Systems, Vol. 2, 1999.
- [6] Eskicioglu, A. M.: *Quality Measurement for Monochrome Compressed Images in the Past 25 Years*. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Vol. 4, 2000.
- [7] Eude, T. ja Mayache, A.: *An Evaluation of quality metrics for compressed images Based on human Visual Sensitivity*. Fourth International Conference on Signal Processing (ICSP'98), China, 1998, s. 779–782.
- [8] Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Qian Huang, Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D. ja Yanker, P.: *Query by image and video content: the QBIC system*. Computer, Volume: 28 Issue: 9, 1995, s. 23–32.
- [9] Fränti, P.: *Blockwise Distortion Measure for Statistical and Structural Errors in Digital Images*. University of Joensuu, Department of Computer Science Report Series A-1996-8, 1996.
- [10] Gonzalez, R. C. ja Woods, R. E.: *Digital Image Processing*. Prentice-Hall, New Jersey 2002.

- [11] Huang, J., Kumar, S. R., Mitra, M., Zhu, W. J. ja Zabih, R.: *Image indexing using color correlograms*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997, s. 762–768.
- [12] Kaarna, A. ja Parkkinen, J.: *Quality Metric for Multispectral Image Compression*. Journal of the Imaging Society of Japan, Vol. 41, no. 4, 2002, s. 379–391.
- [13] Kasutani, E. ja Yamada, A.: *The MPEG-7 Color Layout Descriptor: a Compact Image Feature Description for High-speed Image/Video Segment Retrieval*. Proc. International Conference on Image Processing, Vol. 1, 2001, s. 674–677.
- [14] Kohonen, T.: *The self-organizing map*. Proceedings of the IEEE, Vol. 78 Issue: 9, 1990, s. 1464–1480.
- [15] Kohonen, T.: *Things you haven't heard about the self-organizing map*. IEEE International Conference on Neural Networks, Vol.3, 1993, s. 1147–1156.
- [16] Kohonen, T.: *New developments and applications of self-organizing maps*. Proc. International Workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing, 1996, s. 164–172.
- [17] Kohonen, T.: *Self-Organizing Maps*. Springer Series in Information Science 30, Third Edition. Springer-Verlag, Berlin 2001.
- [18] Helsinki University of Technology. *The Self-Organizing Map (SOM)*. Internet WWW-sivu, URL: <http://www.cis.hut.fi/research/som-research/som.shtml> (3.2.2003).
- [19] Kosheleva, O. ja Kreinovich, V.: *On the Optimal Choice of Quality Metric in Image Compression*. Fifth IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI'02), 2002.
- [20] Laaksonen, J. T., Koskela, J. M., Laakso, S. P. ja Oja, E.: *PicSOM - content-based image retrieval with self-organizing maps*. Pattern Recognition Letters, 21 (13-14): 1199-1207, 2000.
- [21] Ma, W-Y., Zhang, H. J.: *Benchmarking of Image Features for Content-based Retrieval*. Record of the 32nd Asilomar Conf. on Signals, Systems & Computers, Vol: 1., 1998.

- [22] Manjunath, B. S., Ohm, J-R., Vasudevan, V. V. ja Yamada, A.: *Color and Texture Descriptors*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, 2001, s. 703–715.
- [23] Martens, J-B.: *Multidimensional Modeling of Image Quality*. Proceedings of the IEEE, Vol. 90, No. 1, 2002, s. 133–153.
- [24] Mojsilovic, A. ja Hu, J.: *A Method for Color Content Matching of Images*. IEEE International Conference on Multimedia and Expo, ICME2000, New York, 2000.
- [25] Mäenpää, T., Ojala, T., Pietikäinen, M. ja Soriano, M.: *Robust texture classification by subsets of Local Binary Patterns*. Proc. 15th International Conference on Pattern Recognition, 2000.
- [26] Ojala, T., Rautiainen, M., Matinmikko, E. ja Aittola, M.: *Semantic Image Retrieval with HSV Correlograms*. Proc. 12th Scandinavian Conference on Image Analysis, Bergen, Norway, s. 621–627.
- [27] Ojala, T. ja Pietikäinen, M.: *Unsupervised texture segmentation using feature distributions*. Pattern Recognition 32, 1999, s. 477–486.
- [28] Pappas, T. N. ja Safranek, R. J.: *Perceptual criteria for image quality evaluation*. Handbook of Image and Video Processing (A. C. Bovik, ed.), s. 669–684, Academic Press, 2000.
- [29] Pass, G., Zabih, R. ja Miller, J.: *Comparing images using color coherence vectors*. Proceedings of the fourth ACM international conference on Multimedia, Massachusetts US, 1996.
- [30] de Pátria, R. P. ja Niterói, R. J.: *Comparing the Influence of Color Spaces and Metrics in Content-Based Image Retrieval*. Int. Symposium on Computer Graphics, Image Processing and Vision, SIBGRAPI '98, 1998.
- [31] de Ridder, H.: *Image processing and the problem of quantifying image quality*. Proc. International Conference on Image Processing, Vol. 2, 2001. s. 3–6.

- [32] Rui, Y. ja Huang, T. S.: *Image Retrieval: Current Techniques, Promising Directions, and Open Issues*. Journal of Visual Communication and Image Representation 10, 1999, s. 39–62.
- [33] Smith, J. R. ja Chang, S-F.: *Local color and texture extraction and spatial query*. Proc. International Conference on Image Processing, Vol. 3, 1996, s. 1011–1014.
- [34] Stricker, M. ja Swain, M.: *The Capacity of Color Histogram Indexing*. Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR '94, 1994, s. 704–708.
- [35] Tompa, D., Morton, J. ja Jernigan, E.: *Perceptually Based Image Comparison*. Proceedings of ICIP2000 conference, 2000.
- [36] University of South Dakota, Internet Sensation & Perception Laboratory. *Weber's Law of Just Noticeable Differences*. Internet WWW-sivu, URL: <http://www.usd.edu/psyc301/WebersLaw.htm> (3.2.2003).
- [37] Wang, Z. ja Bovik, A. C.: *A universal image quality index*. IEEE Signal Processing Letters, Vol. 9, no. 3, 2002, s. 81–84.
- [38] Watt, A. ja Policarpo, F.: *The Computer Image*. Addison-Wesley, New York 1999.
- [39] White, D. A. ja Jain, R.: *Algorithms and Strategies for Similarity Retrieval*. Tech. Rep. VCL-96-101, Visual Comp. Lab., University of California, 1996.

Liite 1: Spektrikuvien virhemitan laskennassa käytetyt Laplace-suodinmatriisit kerroksittain.

	<i>A</i>	<i>B</i>	<i>C</i>																											
$g_x:$	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>2</td><td>-1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	-1	2	-1	0	0	0	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>-1</td><td>-2</td><td>-1</td></tr> <tr><td>4</td><td>8</td><td>4</td></tr> <tr><td>-1</td><td>-2</td><td>-1</td></tr> </table>	-1	-2	-1	4	8	4	-1	-2	-1	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-2</td><td>-1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	-1	-2	-1	0	0	0
0	0	0																												
-1	2	-1																												
0	0	0																												
-1	-2	-1																												
4	8	4																												
-1	-2	-1																												
0	0	0																												
-1	-2	-1																												
0	0	0																												
	<i>A</i>	<i>B</i>	<i>C</i>																											
$g_y:$	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>-2</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> </table>	0	-1	0	0	-2	0	0	-1	0	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>-1</td><td>4</td><td>-1</td></tr> <tr><td>-2</td><td>8</td><td>-2</td></tr> <tr><td>-1</td><td>4</td><td>-1</td></tr> </table>	-1	4	-1	-2	8	-2	-1	4	-1	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>-2</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> </table>	0	-1	0	0	-2	0	0	-1	0
0	-1	0																												
0	-2	0																												
0	-1	0																												
-1	4	-1																												
-2	8	-2																												
-1	4	-1																												
0	-1	0																												
0	-2	0																												
0	-1	0																												
	<i>A</i>	<i>B</i>	<i>C</i>																											
$g_z:$	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>-1</td><td>4</td><td>-1</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> </table>	0	-1	0	-1	4	-1	0	-1	0	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>0</td><td>-2</td><td>0</td></tr> <tr><td>-2</td><td>8</td><td>-2</td></tr> <tr><td>0</td><td>-2</td><td>0</td></tr> </table>	0	-2	0	-2	8	-2	0	-2	0	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>-1</td><td>4</td><td>-1</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> </table>	0	-1	0	-1	4	-1	0	-1	0
0	-1	0																												
-1	4	-1																												
0	-1	0																												
0	-2	0																												
-2	8	-2																												
0	-2	0																												
0	-1	0																												
-1	4	-1																												
0	-1	0																												

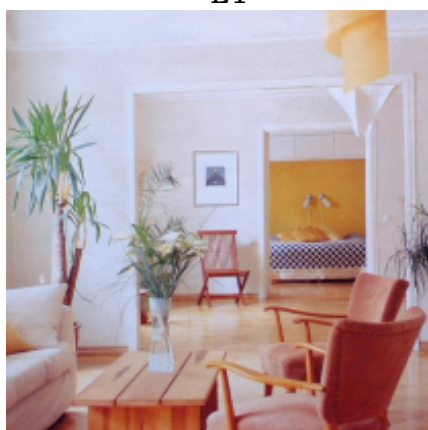
Liite 2: *Olohuone*-kuvasarjan kaikki neljä lehteä.



L1



L4



L6



L7

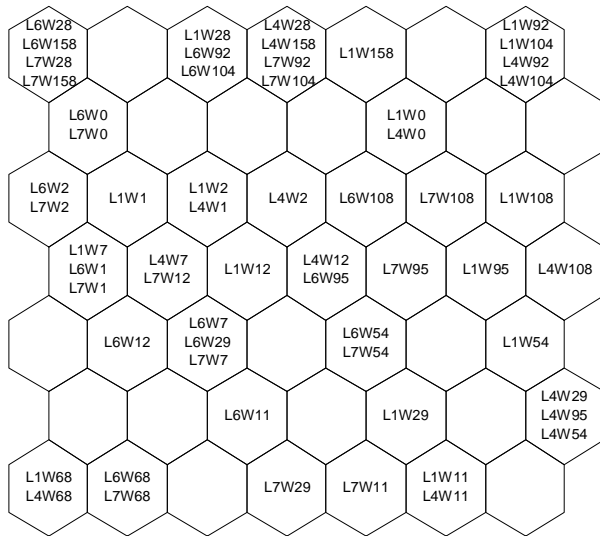
Liite 3a: Syötekuva *kakku*.



Liite 3b: Syötekuva *parveke*.

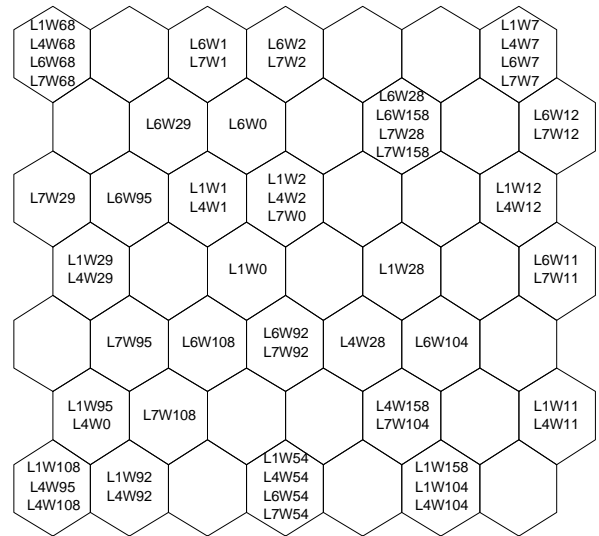


Liite 4: SOM-visualisoinnit syötekuvasta *kakku*.



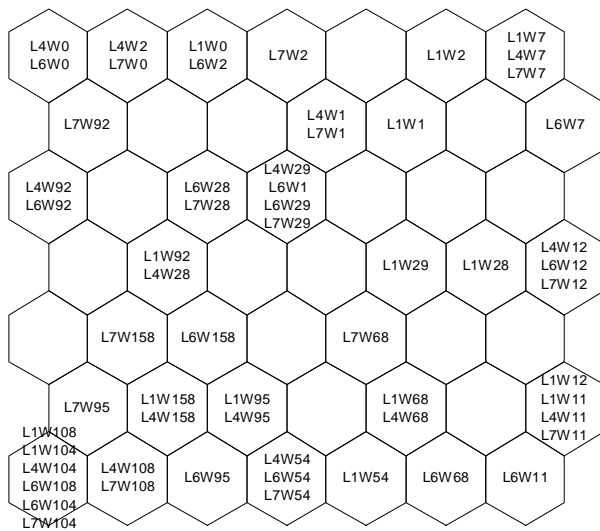
RGB-histogrammi

a



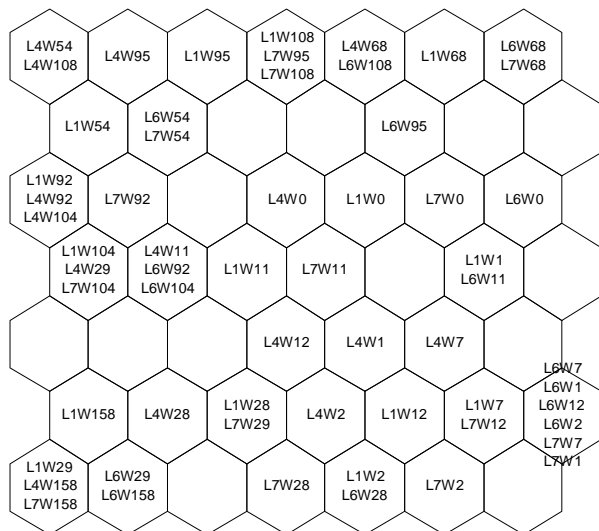
RGB-väriarvot

b



LBP-tekstuuripiirre

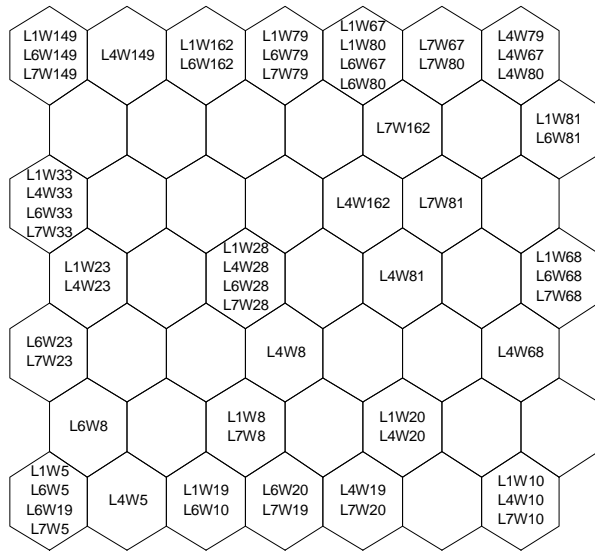
c



Virhemitta

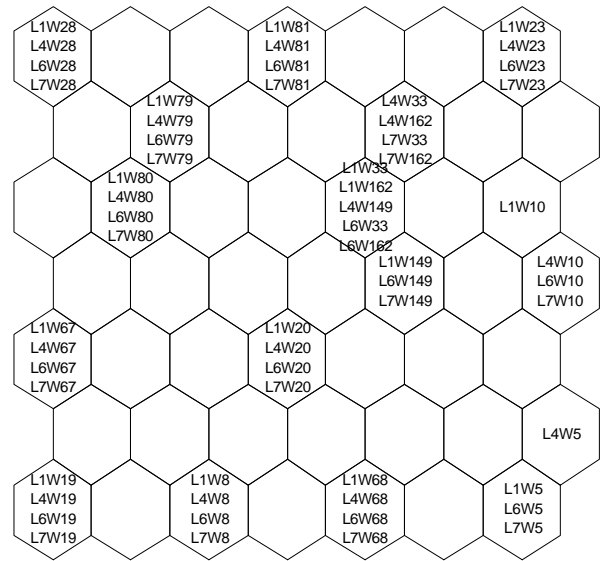
d

Liite 5: SOM-visualisoinnit syötekuvasta *parveke*.



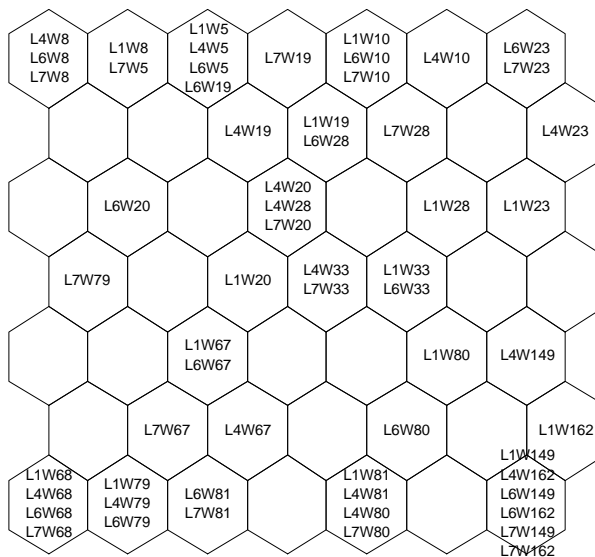
RGB-histogrammi

a



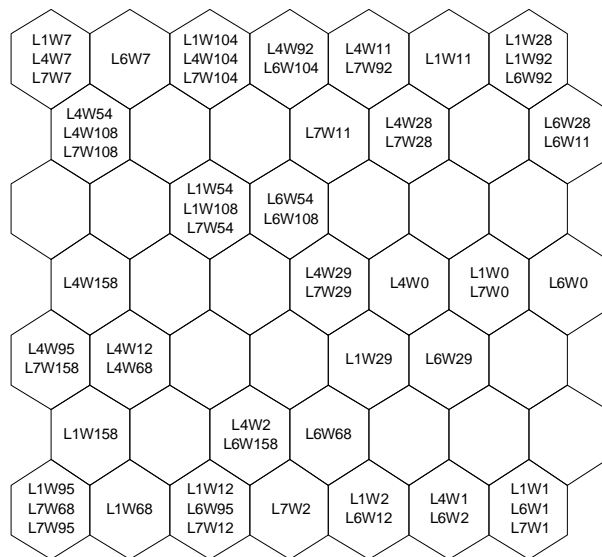
RGB-väriarvot

b



LBP-tekstuuripiirre

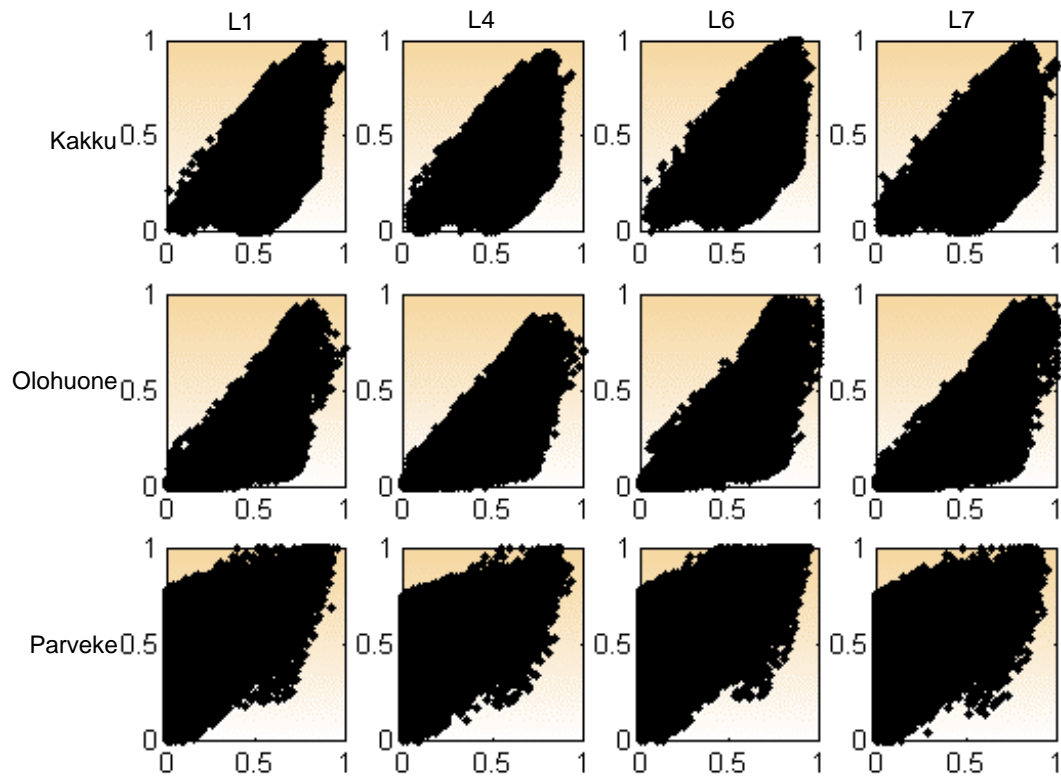
c



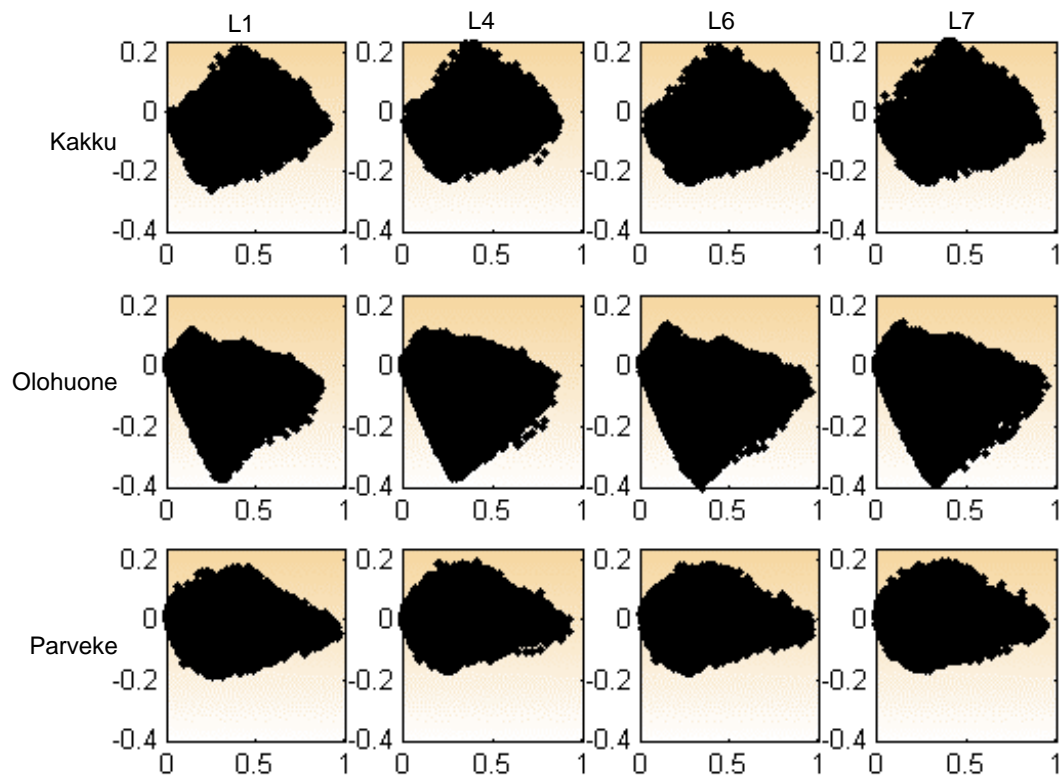
Virhemitta

d

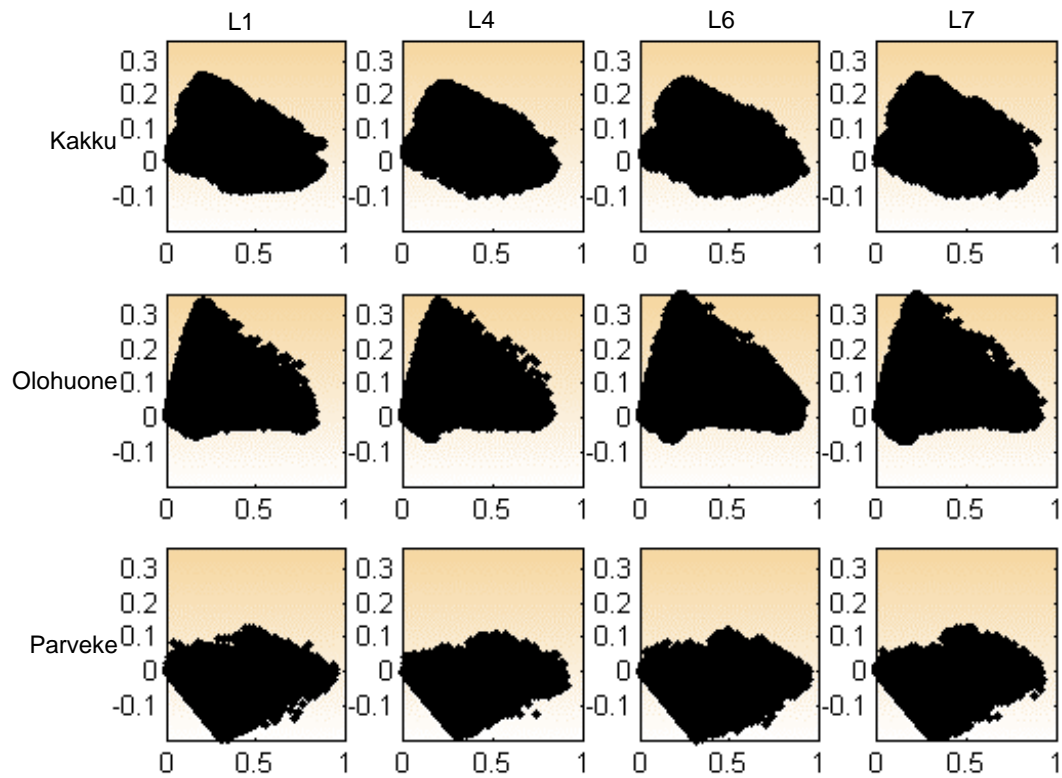
Liite 6a: RGB-väriavaruusprojektiot syötekuvista tasossa R/B.



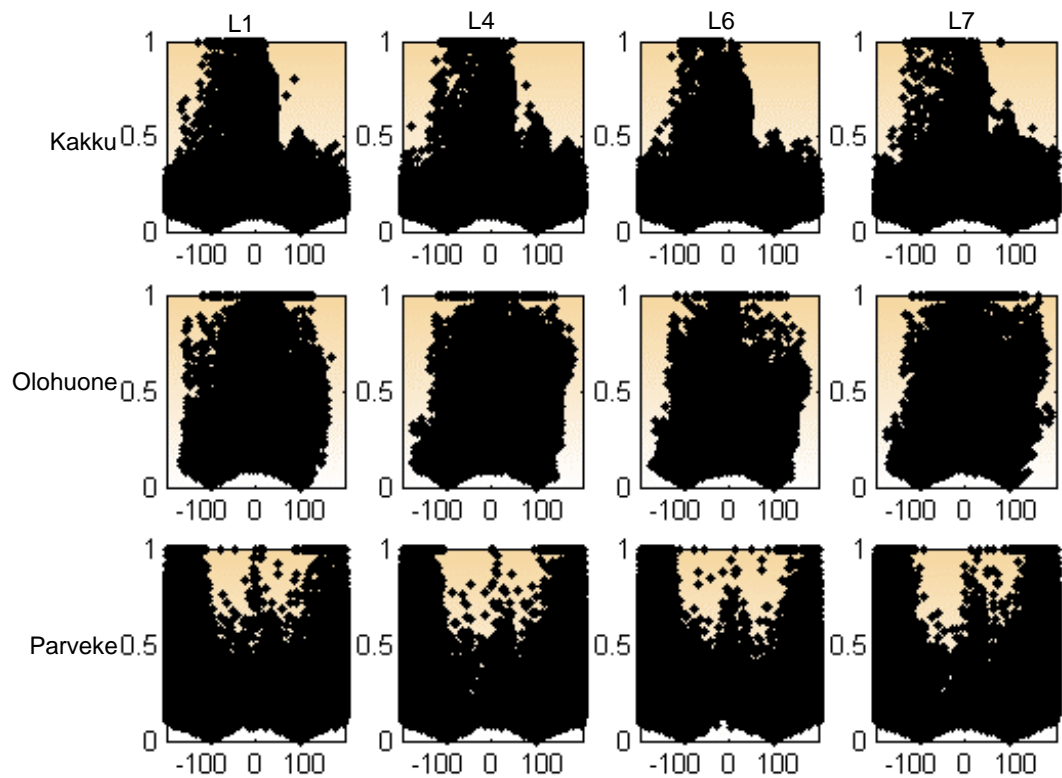
Liite 6b: *Ohta*-väriavaruusprojektiot komponenttien 1 ja 3 suhteen.



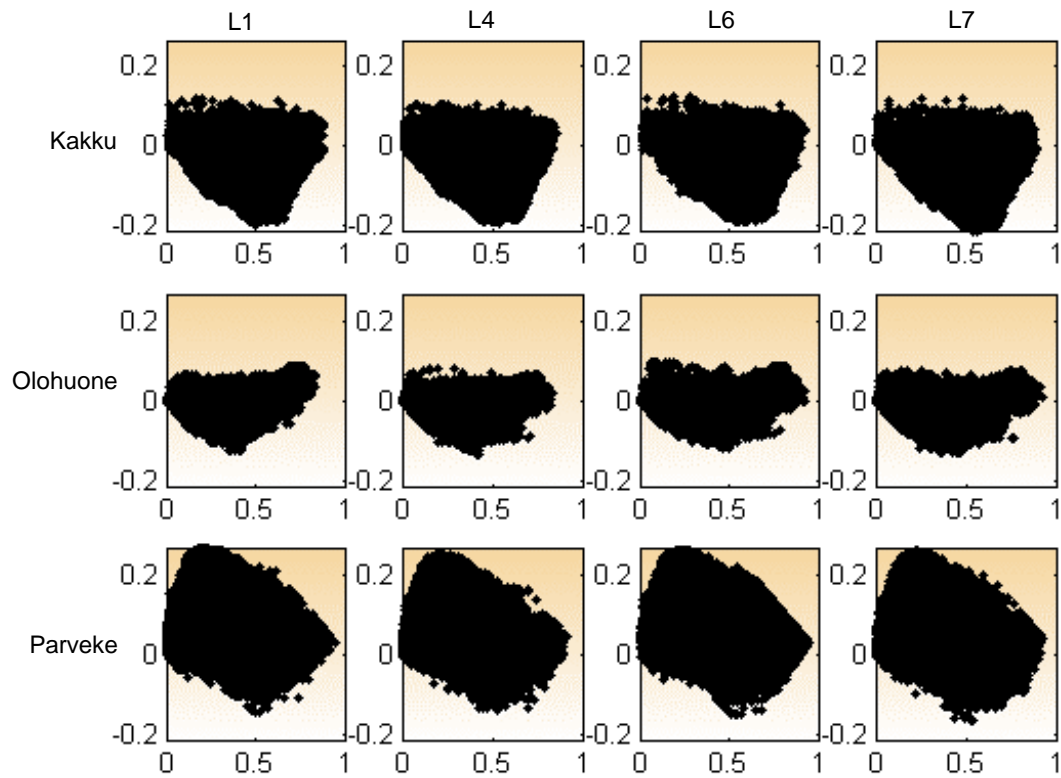
Liite 6c: *YCbCr*-väriavaruusprojektiot tasossa *Y/Cr*.



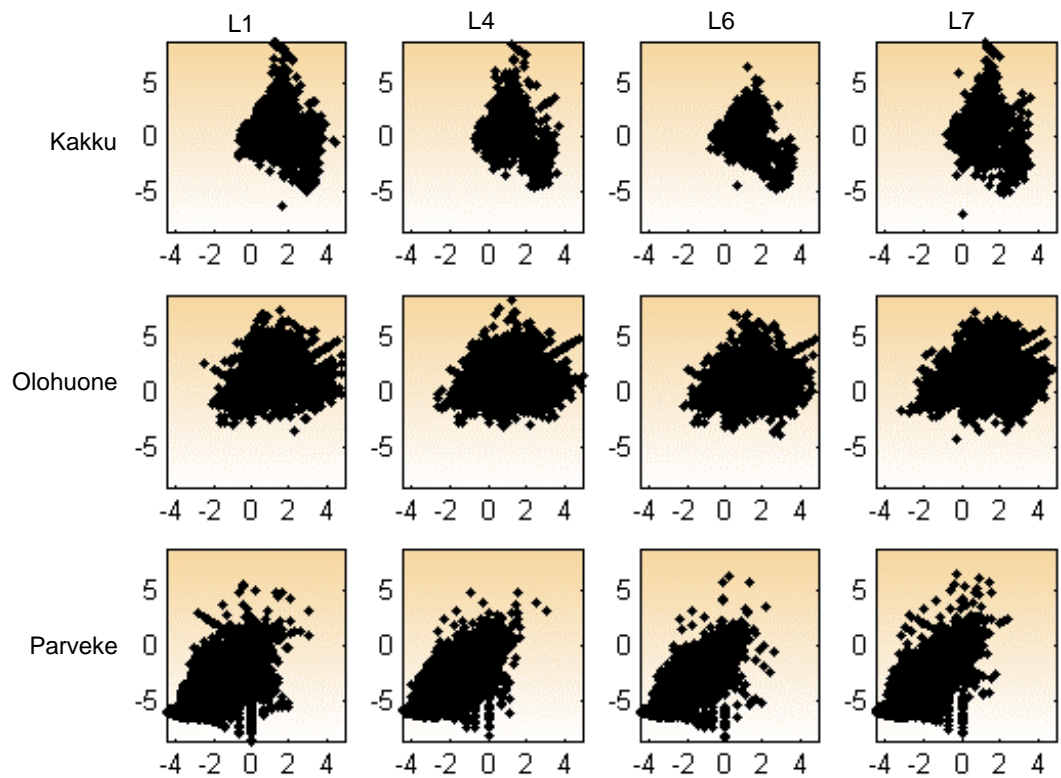
Liite 6d: *HSV*-väriavaruusprojektiot tasossa *H/S*.



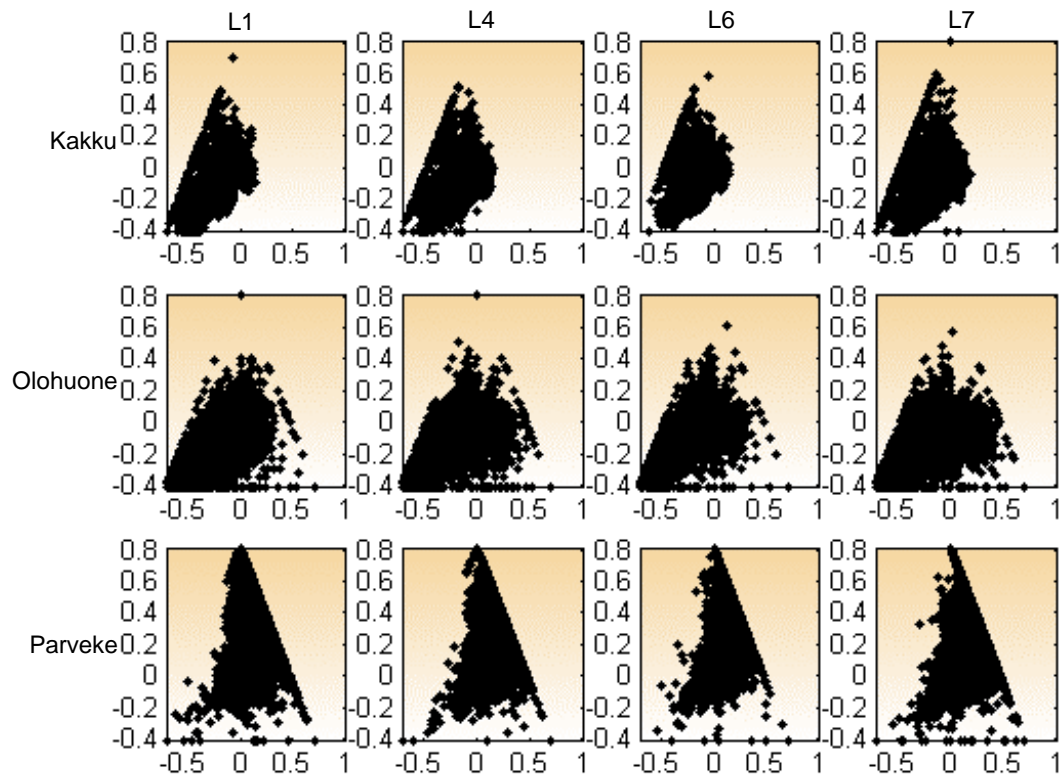
Liite 6e: *NCC*-väriavaruusprojektiot 2:n ensimmäisen komponentin suhteen.



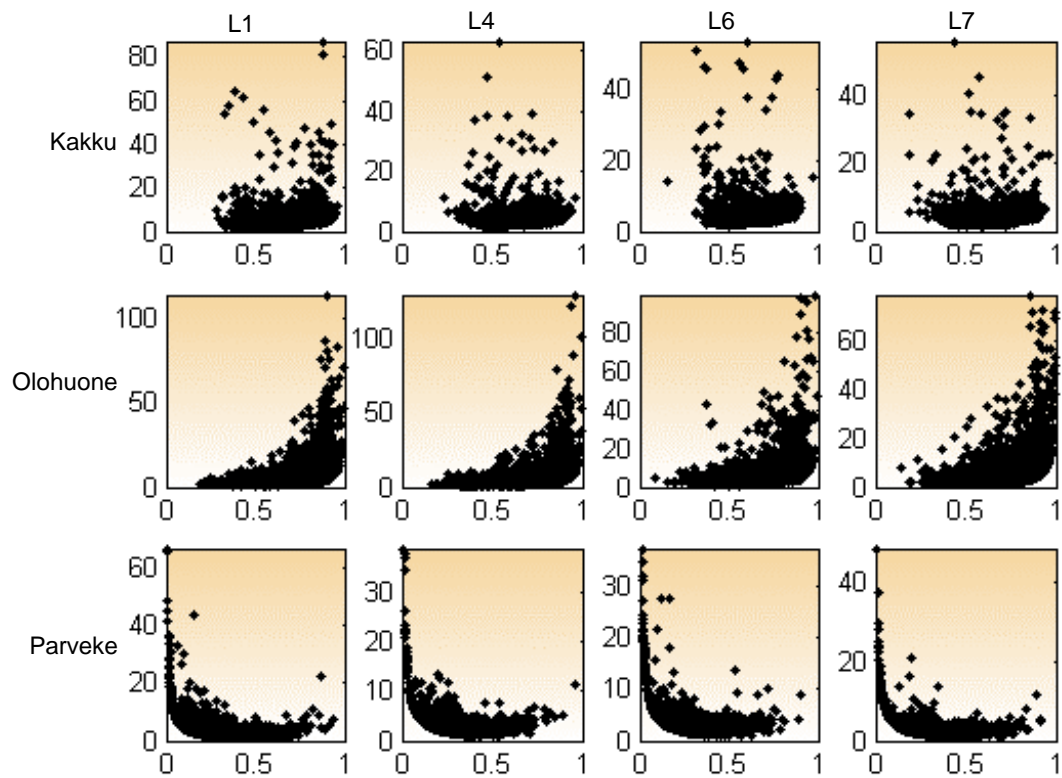
Liite 6f: RGB-logaritmiväriavaruusprojektiot kanavien R/G suhteen.



Liite 6g: *P1P2*-väriavaruusprojektiot.



Liite 6h: *AB*-väriavaruusprojektiot.



Liite 6i: TSL-väriavaruusprojektiot kahden ensimmäisen komponentin suhteen

