

JPEG2000-mosaiikin tekeminen Maanmittauslaitoksen maastokarttarastereista

Per aspera ad astra eli kertomus siitä, kuinka näitä hommia todellisuudessa tehdään

Jukka Rahkonen 2012

Viimeksi muutettu 4. kesäkuuta 2012

Esimerkkiaineistona Maanmittauslaitoksen v. 2012 maastokarttarasterit 1:250000

Lähtöaineisto

Maanmittauspalvelun latauspalvelusta kerättyyn aineistoon kuuluu 65 tiff-tiedostoa, joiden yhteenlaskettu tiedostokoko on 491 565 399 tavua.

Gdalinfo-työkalulla saadaan muun muassa selville, että kuvat ovat 8-bittisiä ja ne käyttävät värikarttaa eli palettia, ja että kuvat on pakattu LZW-pakkausmenetelmällä.

```
G:\MML_250t>gdalinfo UL5L_RVK.tif
Driver: GTiff/GeoTIFF
Files: UL5L_RVK.tif
       UL5L_RVK.tfw
Size is 7680, 7680
Coordinate System is ` `
Origin = (500000.0000000000000000,6762000.0000000000000000)
Pixel Size = (12.500000000000000,-12.500000000000000)
Metadata:
  TIFFTAG_RESOLUTIONUNIT=2 (pixels/inch)
  TIFFTAG_XRESOLUTION=254
  TIFFTAG_YRESOLUTION=254
Image Structure Metadata:
  COMPRESSION=LZW
  INTERLEAVE=BAND
Corner Coordinates:
Upper Left  ( 500000.000, 6762000.000)
Lower Left  ( 500000.000, 6666000.000)
Upper Right ( 596000.000, 6762000.000)
Lower Right ( 596000.000, 6666000.000)
Center      ( 548000.000, 6714000.000)
Band 1 Block=7680x1 Type=Byte, ColorInterp=Palette
  Color Table (RGB with 256 entries)
    0: 0,0,0,255
    1: 38,38,38,255
    2: 51,51,51,255
    3: 102,102,102,255
```

1. GDAL-virtuaalirasterin teko alkuperäisistä kuvista

GDAL-virtuaalitiedoston avulla voidaan yksinkertaisesti koota kokonaisuudesta joukko kuvia virtuaaliseksi rasterikuvaksi, jota voidaan jatkossa käsitellä niin kuin se olisi yksi ainoa suuri

rasterikuva. Virtuaalirasteri voidaan tehdä gdalbuildvrt-työkalulla

```
G:\MML_250t>gdalbuildvrt 250t_mosaiikki_1.vrt *.tif

0.Warning 6: UK3L_RVK.tif has different values than the first raster for some entries in the color table.
The end result might produce weird colors.
You're advised to preprocess your rasters with other tools, such as pct2rgb.py or gdal_translate -expand RGB
to operate gdalbuildvrt on RGB rasters instead
Warning 6: UK3R_RVK.tif has different values than the first raster for some entries in the color table.
.Warning 6: UK4L_RVK.tif has different values than the first raster for some entries in the color table.
```

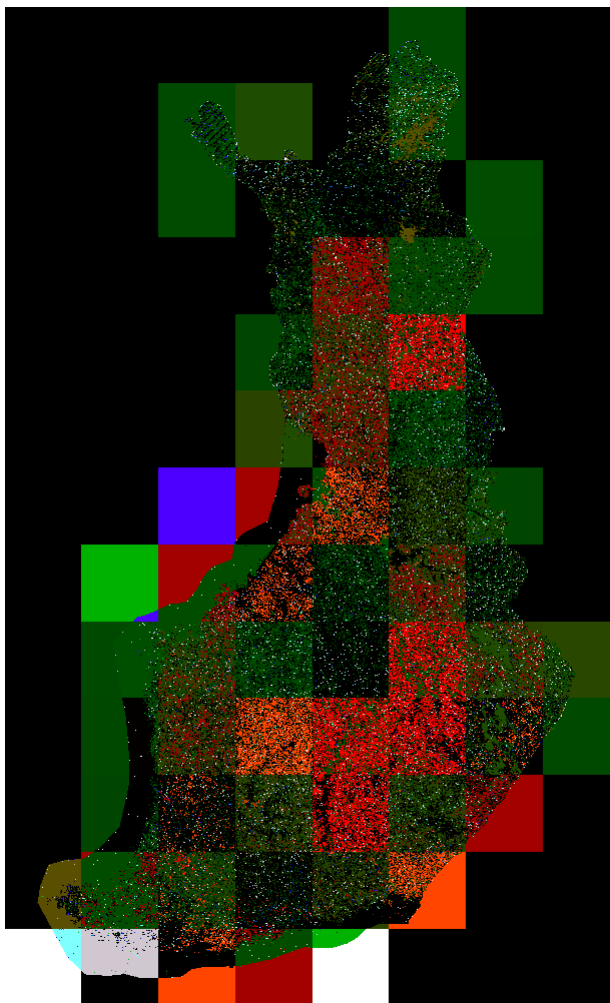
Painetaan mieleen oudot viestit ja varoitukset, mutta ei piitata niistä tässä vaiheessa enempää. Virtuaalirasteri on nyt joka tapauksessa luotu, ja gdalinfo-työkalu luettelee nyt mm. kaikki yksittäiset tiedostot, joista se on yhdistetty, sekä niiden yhteisesti kattaman alueen, jonka koko tässä tapauksessa on 61440 x 99840 pikseliä.

```
G:\MML_250t>gdalinfo 250t_mosaiikki_1.vrt
Driver: VRT/Virtual Raster
Files: 250t_mosaiikki_1.vrt
       UK2R_RVK.tif
       UK3L_RVK.tif
       UK3R_RVK.tif
...
Size is 61440, 99840
Coordinate System is ` '
Origin = (20000.000000000000000,7818000.000000000000000)
Pixel Size = (12.500000000000000,-12.500000000000000)
Corner Coordinates:
Upper Left  ( 20000.000, 7818000.000)
Lower Left  ( 20000.000, 6570000.000)
Upper Right ( 788000.000, 7818000.000)
Lower Right ( 788000.000, 6570000.000)
Center      ( 404000.000, 7194000.000)
Band 1 Block=128x128 Type=Byte, ColorInterp=Palette
  Color Table (RGB with 256 entries)
    0: 0,0,0,255
    1: 0,26,255,255
    2: 0,102,229,255
...
```

2. Virtuaalirasterista oikeaksi tiff-kuvaksi ja ensimmäinen vastoinkäyminen

Muunnetaan kokeeksi edellä luotu virtuaalirasteri GeoTIFF-kuvaksi. Testikuvan kooksi riittää mainiosti 1 prosentti alkuperäisestä sekä leveys- että korkeussuuntaan.

```
G:\MML_250t>gdal_translate 250t_mosaiikki_1.vrt 250t_testi_1.tif -outsize 1% 1%
Input file size is 61440, 99840
0...10...20...30...40...50...60...70...80...90...100 - done.
```



Tulos on outo, mutta mitä oikein tapahtui? No, GDAL-virtuaalimosaiikki tukee värikartan käyttöä, mutta se soveltaa jokaiseen virtuaalitiedostossa viitattuun kuvatiedostoon yhtä ja samaa värikarttaa. Tämän värikartan se ottaa ensimmäisestä vastaan tulleesta alkuperäisestä kuvatiedostosta. Ikävä kyllä Maanmittauslaitoksen alkuperäisaineistossa yksittäisillä kuvatiedostoilla on jokaisella oma värikarttansa. Tulokuvasta nähdään, että virtuaalimosaiikin ensimmäisen kuvatiedosto osuu kuvan vasempaan alakulmaan, ja sen värit ovat oikeat. Yksikään muu alkuperäisistä kuvatiedostoista ei käytä samaa värikarttaa, ja siitä syystä lopputulos on kylläkin hauska, mutta käyttökeltoton.

Gdalbuildvrt-työkalu antoi tästä varoituksen ”*The end result might produce weird colors*”. Samoin se antoi ehdotuksia tilanteen korjaamiseksi: ”*You're advised to preprocess your rasters with other tools, such as pct2rgb.py or gdal_translate -expand RGB to operate gdalbuildvrt on RGB rasters instead*”

3. Värikarttaa käyttävien rastereiden purkaminen kolmikanavaisiksi värikuviksi

Puretaan värikarttaa käyttävät tiff-kuvat kolmikanavaisiksi RGB-kuviksi, pakataan ne tilan säästämiseksi deflate-menetelmällä, ja tehdään niistä samalla GeoTIFF-kuvia. Tarvittavat gdal_translate-komennot ovat tällaisia:

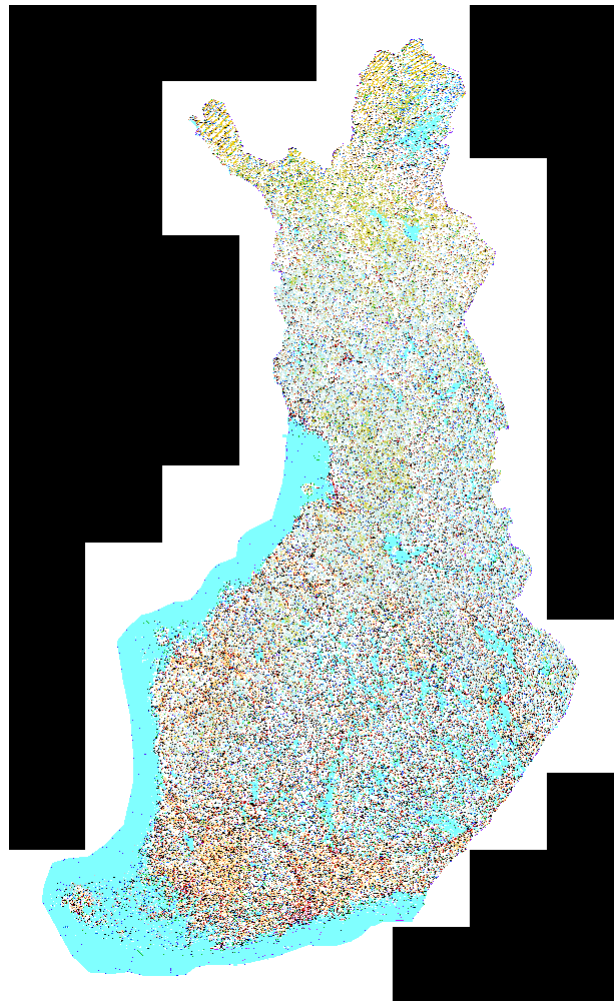
```
gdal_translate -of Gtiff -a_srs epsg:3067 -co compress=deflate lähtökuva.tif  
tulokuva.tif
```

4. Toinen yritys RGB-virtuaalirasterilla ja pienempi vastoinkäyminen

Toistetaan 1. ja 2. vaiheen toimenpiteet eli luodaan ensin uusista RGB-kartoista uusi virtuaalirasteri ja muunnetaan tämä tiff-kuvaksi

```
G:\MML_250t\RGB>gdalbuildvrt 250t_mosaiikki_2.vrt *.tif
```

```
G:\MML_250t\RGB>gdal_translate 250t_mosaiikki_2.vrt 250t_testi_2.tif -outsized 1% 1%
```



Kartan värisävyt ovat nyt kunnossa, mutta kuva kokonaisuutena on ruma, koska ne alueet, joille ei ole lähtöaineistossa mitään sisältöä, maalataan täysin mustaksi.

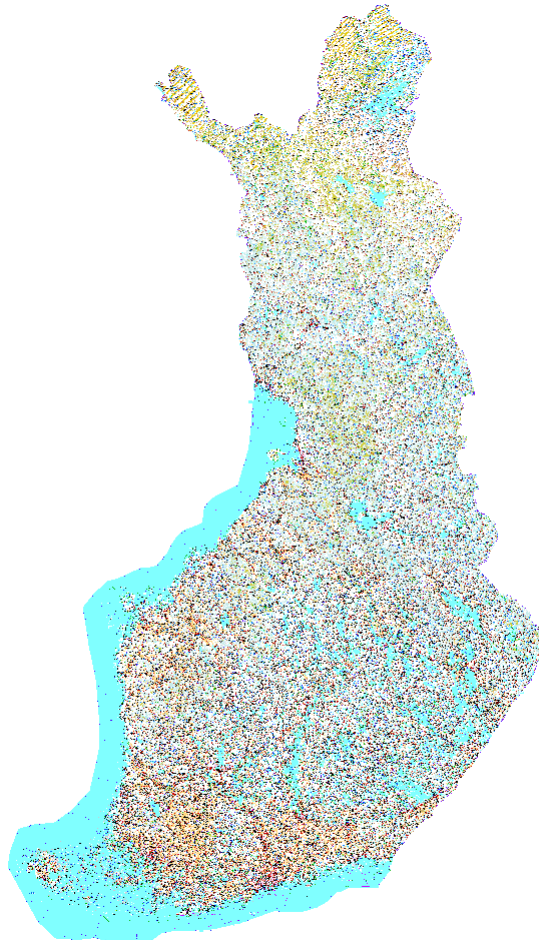
5. Taustavärin määrittäminen virtuaalirasterille ja kolmas yritys

Gdalbuildvrt-ohjelmasta löytyy kuin löytyykään parametrit, joilla voidaan asettaa virtuaalirasterin taustaväri. Toistetaan koe käyttämällä seuraavia komentoja.

```
G:\MML_250t\RGB>gdalbuildvrt -hidenodata -vrtnodata "255 255 255"  
250t_mosaiikki_3.vrt *.tif
```

```
G:\MML_250t\RGB>gdal_translate 250t_mosaiikki_3.vrt 250t_testi_3.tif -outsized 1% 1%
```

Huom! Jos kokeilet tätä itse, niin gdalbuildvrt -komentorivin vieminen komentoikkunaan tästä asiakirjasta leikepöydän kautta ei ehkä onnistu. Jos saat virheilmoituksen tai tulos on outo, kirjoita komento näppäimistöltä.



Tulos näyttää nyt hyvältä ja voidaan siirtyä JPEG2000-pakkaamiseen.

6. Pakollinen välivaihe – TIFF-pakkauksen purkaminen (se on vain levytilaa)

JPEG2000-pakkaus tehdään tässä esimerkissä Kakadu-demo-ohjelmalla "Kdu_compress". Ohjelma on valittu ensiksikin siitä syystä, että se suoriutuu tehtävästä, ja toisaalta siitä syystä, että kuka tahansa saa sitä tähän tarkoitukseen käyttää. Kakadu on varsinaisesti kaupallinen JPEG2000-ohjelmakirjasto, mutta sen demo-ohjelmien käyttö on sallittua ei-kaupallisiin tarkoituksiin, eikä ohjelmien käyttöä ole estetty millään keinotekoisilla teknisillä rajoituksilla.

Kakadun demo-ohjelmat voi ladata ilmaiseksi ja kirjautumatta sivustolta <http://www.kakadusoftware.com/> Lataussivu löytyy valikon "kakadu software – downloads" kautta. Valikko ei tule näkyviin tai ei ainakaan toimi kaikilla Internet Explorer -selaimen versioilla. Lataussivun suora linkki, joka toimii nyt (toukokuussa 2012).

http://www.kakadusoftware.com/index.php?option=com_content&task=view&id=26&Itemid=22

Kdu_compress on tehty demo-ohjelmaksi esittelemään ja mainostamaan Kakadu-JPEG2000 -kirjaston mahdollisuuksia. Siitä syystä ohjelmassa on tiettyjä rajoituksia, joiden kanssa täytyy vain tulla toimeen. Tämän esimerkin pakkaustehtävässä kohdataan nämä rajoitukset:

- Kdu_compress-ohjelman ilmainen demo lukee vain pakkaamattomia tiff-tiedostoja.
- Kdu_compress kopioi automaattisesti lähtökuvana käytettävän GeoTIFF-kuvan georeferoinnin tulokseksi syntyvään JPEG2000-kuvaan. Mitään muuta keinoa georeferoinnin tekemiseksi ei demo-ohjelmassa sitten olekaan.

Näistä demo-ohjelman rajoituksista johtuen lähtöaineistona on käytettävä pakkaamatonta GeoTIFF-muodossa olevaa kuvaa, ja siitä syystä tässä vaiheessa virtuaalirasterina olevasta maastokartta-aineistosta on ensin tehtävä tällainen kuva gdal_translate -ohjelmalla. Tarvittava komento on

```
gdal_translate -of GTiff -a_srs epsg:3067 250t_mosaiikki_3.vrt 250t_bigtiff.tif
```

Muunnoksen tuloksena syntyy 61440 x 99840 pikselin kokoinen 3-kanavainen kuva, jonka tiedostokoko on noin 18,4 Gt.

6. JPEG2000-pakkaus

JPEG2000 on monimutkainen tiedostomuoto eikä sen erilaisten parametrien merkityksen selostaminen kuulu tämän ohjeen aiheeseen. Alla on kuitenkin ensin komento, jolla 1:250000 maastokarttarasteri on hyvällä menestyksellä pakattu, ja komennon jäljessä lyhyet kommentit kunkin parametrin merkityksestä. Komentorivillä tämä komento on annettava ilman rivinvaihtoja.

```
C:"Program Files (x86)"\Kakadu\kdu_compress
-i 250t_bigtiff.tif
-o 250t_rpcl.jp2
-rate 0.8
Clayers=28
Clevels=9
Corder=RPCL
ORGgen_plt=yes
Cprecincts={256,256},{256,256},{128,128},{64,64}
Stiles={20000,20000}
-flush_period 20000
-com "Processed by http://latuviitta.org"
-com "Terms of use:
http://www.maanmittauslaitos.fi/en/NLS_open_data_licence_version1_20120501"
-progress 2000
-mem
-cpu 0
```

C:\Program Files (x86)\Kakadu\kdu_compress

- suoritettavan ohjelman nimi; huomaa lainausmerkit ” ”välilyöntimerkkejä sisältävän tiedostopolun osan ympärillä

-i 250t_bigtiff.tif

- -i = input, lähtötiedoston nimi

-o 250t_rpcl.jp2

- -o = output, tulostiedoston nimi

-rate 0.8

- Tulostiedoston bittimäärä, yksikkönä bittiä/lähtökuvan pikseli. Tässä tapauksessa lähtökuva on 24-bittinen (3 kanavaa, 8 bittiä kussakin). Kun 24 puristetaan 0,8:ksi, niin pakkaussuhteeksi tulee 1:30. Samalla tiedetään jo etukäteen, että 18 Gt alkuperäisestä kuvasta tulee syntymää 600 Mt JPEG2000-tiedosto.

Clayers=28

- Luodaan JPEG2000-bittivirtaan 28 eri laatutasoa. JPEG2000-bittivirtaa, voidaan lukea alusta alkaen ja muodostaa siitä ensin vähillä biteillä heikompileatuinen kuva, jonka laatu paranee sitä mukaa kun bittivirtaa luetaan eteenpäin. Clayers järjestää bittivirran niin, että tätä mahdollisuutta voidaan hyödyntää. Harvat ohjelmat ihan oikeasti kuitenkaan käyttävät tätä ominaisuutta.

Clevels=9

- Luodaan 9 yleistystasoa, eli versioita, jossa alkuperäinen pikselikoko aina kaksinkertaistuu sarjassa 2, 4, 8, 16 ... JPEG2000 on myös tässä suhteessa edistynyt, eli yleistettyjen tasojen luomiseksi ei tarvita lisää bittejä. Yleistystasojen luominen merkitsee vain bittivirran järjestämistä niin, että pääsy heikomman resoluution versioihin on mahdollista.

Corder=RPCL

- JPEG2000-bittivirran edistymisjärjestys voidaan järjestää neljän eri komponentin mukaan. R=resoluutio, P=paikka, C=komponentti (esim. värikanava) ja L=Laatutaso. Tyypillisen paikkatietokuvat ovat suuria, ja niiden todellisessa käytössä kuvasta on saatava esille pieni alue jostain satunnaisesta kohdasta kuvaa. Tällaiseen käyttöön parhaiten sopivia edistymisjärjestyksiä ovat RPCL tai PCRL. Tähän on valittu RPCL, koska se samalla antaa nopean pääsyn koko Suomen mittakaavaan.

ORGen_plt=yes

- Mikä lie parametri, ilmeisesti on joskus luettu netistä, että se olisi hyvä olla käytössä

Cprecincts={256,256},{256,256},{128,128},{64,64}

- Nämä tarvitaan, jotta nopea pääsy satunnaiselle alueelle on mahdollista.

Stiles={20000,20000}

- Tulokuva jaetaan sisäisesti 20000 x 20000 pikselin suuruisiin paloihin, niin ettei ohjelman tarvitse käsitellä koko tietomäärää muistissa yhdellä kertaa. Huom. lähtökuva 18 Gt.

-flush_period 20000

- Annetaan pakkausohjelmalle pyyntö yrittää kirjoittaa levyille siihen mennessä valmistunut bittivirta aina 20000 käsitellyn rivin jälkeen, jos se vain sattuu olemaan mahdollista. Tässä tapauksessa se on mahdollista, koska aikaisemmin määrättiin 20000 rivin korkuisten sisäisten laattojen käyttö, ja niiden saumakohdassa levyille hyljättäminen on mahdollista.

-com "Processed by <http://latuviitta.org>"

- Kirjoitetaan JP2-tiedoston otsikkoon maininta pakkauksen tekijästä

-com "Terms of use:

http://www.maanmittauslaitos.fi/en/NLS_open_data_licence_version1_20120501"

- Kirjoitetaan JP2-tiedoston otsikkoon maininta MML:n käyttöehdoista

-progress 2000

- Pyydetään ohjelmaa tulostamaan näytölle tietoja pakkaamisen edistymisestä aina 2000 käsitellyn rivin jälkeen

-mem

- Pyydetään ohjelmaa tulostamaan tietoja muistin käytöstä pakkaamisen valmistuttua

-cpu 0

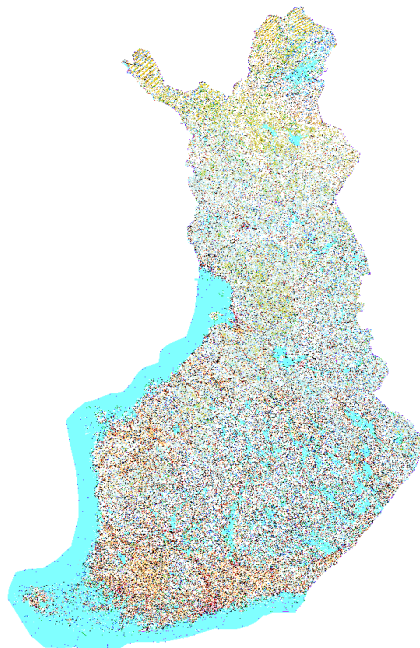
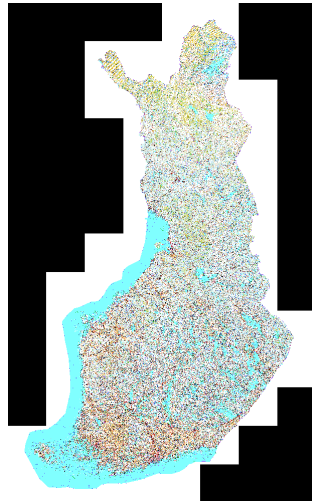
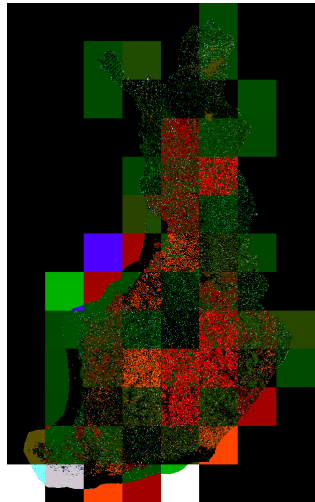
- Pyydetään ohjelmaa tulostamaan tietoja prosessoriajan käytöstä pakkaamisen valmistuttua.

7. Nautitaan lopputuloksesta

Oikeastaan se oli siis aika helppoa, sitten kun tietää mitä on tekemässä. Lähtöaineistot ja tarvittavat ohjelmistot ovat vapaasti saatavilla, jos kiinnostaa tehdä sama perässä. Valmis kartta on myös saatavilla BitTorrentin kautta. Linkki tiedoston kuvaukseen seurantal palvelimen sivustolla

<http://laillisettorrentit.net/index.php?page=torrent-details&id=d1d3a55c9f1a5ab3897fee1f6f8d778cb3f220b0>

Karttaa voidaan katsoa esimerkiksi Kakadu-demo-ohjelmalatauksen mukana tulleella Kdu_show-ohjelmalla. Avoimen lähdekoodin paikkatieto-ohjelmista kuvan voi avata suoraan OpenJUMP Plus-versiolla, Kosmo GIS:llä ja gvSIG-ohjelmalla. Quantum GIS avaa kuvan myös, kunhan sen mukana tulevassa GDAL-versiossa on tuki jonkin hyvän JPEG2000-ajurin käytölle. OSGeo4W-asennusohjelman avulla tuollaisen tuen saa asentamalla MrSID-GDAL-ajurin.



8. Jälkikirjoitus JPEG2000:sta ja väripalettia käyttävistä kuvista

Aallokkeisiin perustuva pakkausmenetelmä, jota JPEG2000 käyttää, ei sovi erityisen hyvin väripalettia käyttäville kuville. Kakadu, toisin kuin monet muut JPEG2000-kirjastot, kyllä tukee paletin käyttöä, mutta sekin antaa aiheesta tällaisen varoituksen:

```
Kakadu Warning:  
Optimizing palette ...  
Use '-no_palette' to avoid nasty palettization effects when reconstruction  
is anything but lossless. '-no_palette' also saves the time consuming  
palette reordering step.
```

Seuraavat kuvat näyttävät, miten tämä kelju ilmiö näkyy käytännössä. Ensimmäisessä kuvassa on tehty häviötön pakkaus ja kuva vastaa luonnollisesti täysin alkuperäistä karttaa. Toisen kuvan esittämässä häviöllisessä pakkauksessa pakkausohjelma on poiminut väripaletista värejä melko sattumanvaraisessa järjestyksessä. Häviöllistä pakkausta tehtäessä täytyy siis väripalettia käyttävien kuvien värikartta joka tapauksessa aina leventaa ensin RGB-muotoon. Tämä voidaan tehdä etukäteen esimerkiksi `gdal_translate`-ohjelmalla. `Kdu_compress`-ohjelmassa on tätä varten myös erityinen parametri `"-no_palette"`. Se ei kuitenkaan tunnu aina toimivan täysin luotettavasti.

