

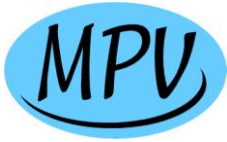
Matti Vuori

## Kamerasovellus luovaan kuvaamiseen

*Kamerakännyköistä tekee ainutlaatuisia valokuvausvälineitä se, että niissä on sisällään käyttöjärjestelmä, jonka päälle voidaan rakentaa kamerasovelluksia mm. C++- tai Python-kielillä. Nokia N900:n erityispiirre on sille saatava FCamera, joka on avoimen lähdekoodin alusta niin vakavien kuin vapaamielisempien kameroiden rakentamiseen. Tämä dokkari esittelee muutamalla sanalla sille laatimani luovan kuvauksen kamerasovelluksen MPVCamera.*



**"Does it produce perfect photos?" – mitä täydellisyys ja laatu oikeasti tarkoittaa luovassa toiminnassa...**



FCamera (<http://fcam.garage.maemo.org/fcamera.html>) on avoimen lähdekoodin kamerasovellus, joka ennenkaikkea demonstroi Stanfordin yliopiston tutkijoiden ja Nokian yhteistyötä kamerateknologiassa. Sovelluksesta tekee poikkeuksellisen se, että se on saatavana Nokia N900 -kamerapuhelimeen ja sen pohjalta on mahdollista laatia omia kamerasovelluksia ja myös julkaista niitä FCameran lisenssin puitteissa.

Olin aiemmin kokeillut kamerasovelluksen laadintaa kevyemmälle kamerapuhelinmallille, mutta sen muistirajoitukset (64 Mt) tekivät asiasta käytännössä mahdottoman. N900:n 256 Mt ja Maemo-käyttöjärjestelmän virtuaalimuisti mahdollistavat viiden megatavun kuvien käsittelyn ilman ongelmia (poislukien liukulukulaskentaa vaativa kuvankäsittely, jossa laitteen prosessori ei enää täysin riitä). FCamera on kirjoitettu C++-kielellä Qt-kirjastolle, mikä toisaalta tekee sovelluksesta kompleksimman kuin Python-kieli, mutta toisaalta antaa monia mahdollisuuksia ja rajattoman sovelluksen kasvunvaran. Jo nyt on löytynyt tärkeitä C++-kielisiä lähdekoodikirjastoja, joista on ollut korvaamaton apu kuvankäsittelyssä.

Seuraavat näyttökuvat esittelevät sovelluksen keskeiset piirteet. Huomaa, että laitteessa on kosketusnäyttö (800 x 480 pikseliä), minkä vuoksi perinteiset kameroiden näyttöjen layout-periaatteet eivät päde.

Ensin ohjelman perusnäyttö

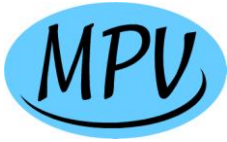


Vasemmassa ylänurkassa on systeemivalikko sovellustason toiminnolle, joita ei yleensä käytetä kuvaamisen aikana (sovelluksen sulkeminen, tietoja sovelluksesta, nettilinkkejä jne...) ja oikeassa reunassa – kuvaajan peukalon hallittavina – kameratoiminnot. Painikevalikon merkinnöistä näkee, miten monipuolinen FCamera on: valotus- ja tarkennusvaihtoehtoja on useita ja valkotasapaino on portaattomasti säädettävissä.

Keltatekstiset toiminnot Style ja Layout ovat MPVCameran uusia toimintoja. Niitä on kahdentasoisia Style ja layout. Katsotaanpa niitä seuraavaksi.

Style-valikon toiminnot ovat kuvan laadullisia ja ulkonäköön vaikuttavia valintoja, joista voidaan käyttää yhtä tai useampaa samaan aikaan.



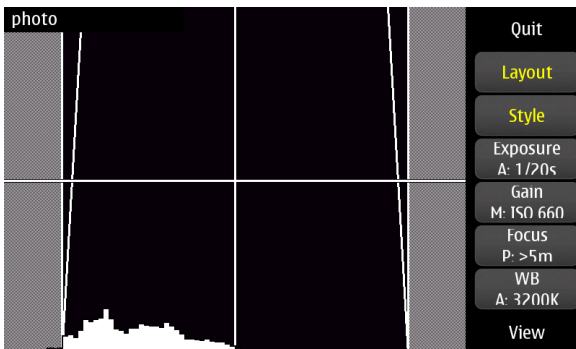


Tummanvihreä taustaväri on merkinä siitä, että kyseisille rukseille saadaan pitkällä painalluksella esille säädettävä parametri, esimerkiksi perspektiivikorjauksen suuruus. Pitkä painallus on hieman ikävä tekniikka, mutta paremman puutteessa sopiva tähän tilanteeseen.

Muutamia poimintoja toiminnoista:

- Devignette poistaa kuvaan tulevaa reunojen tummenemista – kamera-alustan ominaisuus. Mutta sitä voidaan myös lisätä Vignette-toiminnolla!
- InsertObj lisää kuvaan leikekuvan – vaikka kuun taivaalle. Tällainen toiminto on tietävästi joissakin kamerapuhelimeissa vakiovarusteena.
- Kuvalle on erilaisia rajausvaihtoehtoja (Teleada rajaa kuvan magneettikiinnitteiselle teleisäkkeelle, jonka piirtoympyrä jää hieman vajaaksi). Rajausvaihtoehdoille on ohjelmoitu omat etsinkuvansa.
- Tämä lienee ainoa kamera, jossa pystyy kuvalle tekemään perspektiivikorjauksen – hyödyllistä rakennuksia kuvattaessa.
- Erilaisia taiteellisia efektejä on mukana muutamia. Niissä on jouduttu tekemään kompromisseja, jotta laskenta-aika ei kasva kohtuuttoman pitkäksi. Pidempää prosessointia voidaan tehdä myös kuvan ottamisen jälkeen kuvaselaimessa. Siellä voidaan käynnistää mm. kohinanpoistotoiminto.
- Ugly ja Perfect -valinnat ovat vain huumorin ja hivin vuoksi! Tällä kameralla saa täydellisiä kuvia vain laittamalla ruksin ruutuun...

Mutta yleisesti ottaen kaikki efektit ovat luonteeltaan hienovaraisia ja tukevat harkittua kuvan piirteiden muuntamista.



*Etsinkuvaesimerkki: Kun otetaan kuvia 1:1-kuvasuhteella, reunat on rajattu läpinäkyvällä peitolla; kun otetaan peilauskuvia, käytetään hiusristikköä ja kun kuvalle tehdään perspektiivikorjaus, näytetään sen apuviivat.*

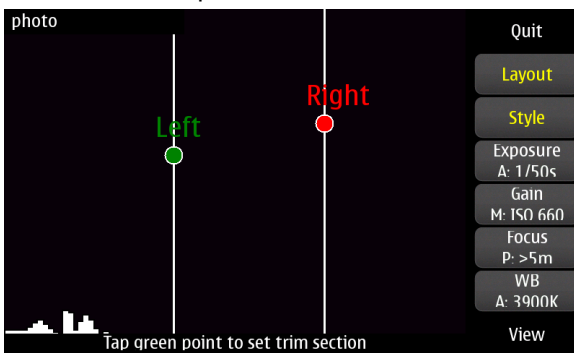
Toinen toimintojen luokka käsittelee kuvien taittoa ja esillepanoa.



Layout-luokan toiminnoilla voidaan tehdä esimerkiksi seuraavia usein tarpeellisia tehtäviä, jotka aiemmin piti aina tehdä kuvankäsittelyohjelmalla tai erikoistuneella PC-sovelluksella:

- Neljän kuvan ruudukko halutun levyisellä kehyksellä
- Peilata kuva sen keskiakselin suhteen
- Tehdä kahdesta kuvasta päällekkäisvalotus.
- Poistaa kuvasta jokin vaakasuora alue!

Viimeksimainitun toiminnon käyttöliittymä demonstroi tapaa täpätä näyttöä ja valita siten aina vuorollaan toinen asemointipiste.



Kamera on muuten tarkoitustaan vastaava, mutta tietenkään pikkuruisella kennolla ja objektiivilla varustettu laite ei pärjää edes hyvien pokkarien kanssa – kuvan kohina on melkoista, dynamiikka ei ole kehuttava ja tarkkuudessa on kehittämisen varaa. Mutta se ei kaikilla kuvaustyyliillä olekaan olennaista – teknistä laatua haetaan sitten paremmilla kameroilla ja tietenkin kamerasovellus siirretään jossain vaiheessa johonkin tulevaan ohjelmitavaan kameralaitteeseen, jossa on kehittyneempi kamera – sellaisia olisi jo saatavilla, ja saman käyttöliittymäkirjaston eli Qt:n ansiosta siirtäminen olisi luultavasti helppoa – tuntematta muiden mallien kamera-ajurien rajapintoja.

Tähän kaikkeen on tarvittu noin kaksi ja puolittuhatta riviä omaa C++/Qt -koodia ja monta talvi-iltaa... Ja tällaiset projektit eivät välttämättä pääty ihan läpivuosiin.

FCameran ja FCam:n lisäksi ohjelmassa on käytetty KDE-projektin qlmageblitz-luokkaa, josta löytyy Qt:n vakiotarjontaa nopeammat ja laadukkaammat funktiot muutamiin kuvankäsittelyoperaatioihin (sekä öljymaalauksen ja hiilipiirrossimulaatioon!) ja kohinanpoistorutiinina Digikam 1.1.0:n (vuodelta 2005) wavelet-pohjaista koodia. Parempaa avointa koodia varmasti olisi, mutta juuri kohinanpoistorutiinit on niin tiukasti sidottu vaikkapa kuvankäsittelyohjelman plugin-arkkitehtuuriin, että niiden uudelleenkäyttö on erittäin työlästä.