

Aalto-yliopisto
Perustieteiden korkeakoulu
Tietotekniikan koulutusohjelma

Videokoodauksen rinnakkaistaminen

Kandidaatintyö

17. helmikuuta 2013

Miro Nurmela

| | |
|---|--|
| Tekijä: | Miro Nurmela |
| Työn nimi: | Videokoodauksen rinnakkaistaminen |
| Päiväys: | 17. helmikuuta 2013 |
| Sivumäärä: | 25 |
| Pääaine: | Ohjelmistotekniikka |
| Koodi: | T3001 |
| Vastuopettaja: | Ma professori Tomi Janhunen |
| Työn ohjaaja(t): | TkT Vesa Hirvisalo (Tietotekniikan laitos) |
| <p>Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.</p> <p>Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.</p> | |
| Avainsanat: | videokoodaus, enkoodaus, dekodeaus, transkoodaus, rinnakkaisohjelmointi, rinnakkainen videokoodaus, rinnakkaislaskenta |
| Kieli: | Suomi |

Sisältö

| | |
|--|-----------|
| 1 Johdanto | 4 |
| 2 Aiempi tutkimus ja taustaa | 5 |
| 3 Videokoodaus | 6 |
| 3.1 Videokoodauksen peruskäsitteet | 6 |
| 3.1.1 Videodatan matka lähteestä näyttölaitteelle | 6 |
| 3.1.2 Videouvan esittäminen ja tallentaminen, diskreetti kosinimuunnos | 7 |
| 3.2 Videokoodauksen tarpeellisuus | 8 |
| 4 Rinnakkaislaskenta | 9 |
| 4.1 Rinnakkaisuuden peruskäsitteet | 9 |
| 4.2 Rinnakkaislaskenta ja nykypäivän tietokonejärjestelmät | 9 |
| 5 Videokoodaus ja rinnakkaislaskenta | 10 |
| 5.1 Rinnekkaisuuden hyödyt videokoodaukselle | 11 |
| 5.2 Erilaiset kiihdytysalustat ja ohjelmistoratkaisut | 12 |
| 6 Yhteenveto | 13 |
| Lähteet | 14 |

1 Johdanto

DISCLAIMER: Huonoa, muistiinpanonomaista tekstiä. Tullaan parantamaan.

2 Aiempi tutkimus ja taustaa

3 Videokoodaus

Viimeisen viidentoista vuoden aikana suurin osa videodatasta on muuttunut analogisesta digitaaliseksi. VHS-kaseteista ja analogisista TV-lähetyksistä on siirrytty Blu-Ray-tekniikoihin, kännykkäkameroihin teräväpiirtotelevisioihin. Tekniikan kehitys ei näy ainoastaan tallennusmedioissa, sillä niin ikään tiedonsiirto on kehittynyt ja muuttunut monin paikoin langattomaksi. (Richardson (2010)) Videodatan määrä on myös valtavassa kasvussa (Cisco (2013), YouTube (2013)). Raaka videodata suuret määrät tallennustilaa eikä sen siirtäminen langattomasti ole mahdollista - tarvitaan siis teknologia, jolla videodataa pakataan ja puretaan (enkoodataan ja dekodataan). Tätä pakkaamisen ja purkamisen prosessia kutsutaan videokoodaukseksi. Pakkaamisen ja purkamisen lisäksi videokoodaus kattaa myös signaalin reaaliaikaisesta kääntämisestä (transkoodaus). Transkoodaus tarkoittaa esimerkiksi enkoodatun datan kääntämistä toiseen koodausstandardiin (Angelides (2011)). Seuraavissa alaluvuissa käsitellään videokoodausmenelmien perusteita. Tästä lähin näihin viitataan termein enkoodaus, dekodaus ja transkoodaus.

CODEC (COder DECOder pair) viittaa ohjelmistoon tai laitteeseen, joka suorittaa videokoodausta (Richardson (2010)).

3.1 Videokoodauksen peruskäsitteet

3.1.1 Videodatan matka lähteestä näyttölaitteelle

Videodata saa alkunsa lähteestä, joka on tyypillisesti kamera, joka tallentaa raakadatan. Tämän jälkeen suoritetaan enkoodaus, jonka jälkeen tiivistetty data voidaan tallentaa tai siirtää. Tosimaailma on ajallisesta ja paikallisesti jatkuvaa, mutta digitaalinen maailma ei. Jotta videodataa voidaan tallentaa, täytyy maailmasta ottaa näytteitä eri ajanhetkillä. Tyypillisesti otokset eri ajanhetkiltä ovat nelikulmaisia ruudukkoja - tällainen diskreetti esitys voidaan enkoodata ja esimerkiksi kameran muistikortille tallentaa. Näin peräkkäisistä tallennetuista ruudukoista voidaan koota myöhemmin videokuvaa. (Richardson (2010))

Jos enkoodaus on näyttölaitteelle väärässä formaatissa, täytyy suorittaa transkoodaus oikeen formaattiin.

Prosessin toisessa päässä tiivistetty data dekodataan ja järjestetään niin, että se voidaan halutulla kohdelaitteella näyttää käyttäjälle.

3.1.2 Videouvan esittäminen ja tallentaminen, diskreetti kosinimuunnos

Kuten mainittu, digitaalinen maailma ei todellisen maailman tavoin ole jatkuva. Videokuva tallentaessa täytyy maailmasta tellennettu kuva sovittaa haluttuun formaattiin - suorakaiteen muotoiseen ruudukkoon näytteitä maailmasta.

Digitaaliset kuvat esitetään siis näytteinä tosimaailmasta - näytteet tunnetaan paremmin kuvapisteinä eli pikseleinä. Näistä kuvapisteistä kootaan peräkkäisiä ruutuja, joita peräkkäin toistamalla saadaan aikaan vaikutelma elävästä kuvasta. Värikuvaa kuvatesa jokaisella kuvapisteellä täytyy olla tieto siinä esiintyvien värien määrästä. Suositussa RGB-väriavaruudessa jokaisella kuvapisteellä on kolme parametriä - punaisen, vihreän ja sinisen värin määrä. Tiheys, jolla ruutuja tallennetaan määrää ruutunopeuden (frame rate). Erään mittarin videokuvan laadusta antavat kuvapisteiden määrä ja ruutunopeus. (Richardson (2010))

Videokoodausta varten on olemassa häviöllisiä ja häviöttömiä metodeita. Yhteistä niille on videodatan vahvan spatiaalisen ja ajallisen redundanssin hyödyntäminen. Käytännössä peräkkäisissä ruuduissa on suurella todennäköisyydellä lähes samat kuvapisteet ja yhdessä ruudussa lähekkäiset kuvapisteet muistuttavat toisiaan suurella todennäköisyydellä. (Richardson (2010))

Raakoja näytteitä ei ole tapana säilyttää, vaan ne muunnetaan paremmin enkoodaukseen sopivaan muotoon. Esitellään tässä yksi yleisimmistä muunnoksista, diskreetti kosinimuunnos (DCT).

DCT on Fourier-muunnoksen kaltainen muunnos, joka muuntaa videokoodauksen tapauksessa ruutujen näyteblokkien pikseliarvoja eri taajuuksilla värähteleviksi kosinifunkitoiksi. Saavuettu etu on se, että häviöllisissä koodausmenetelmissä voidaan jättää koodamatta pienet korkeataajuiset funktiot - niitä ihminen on huono huomaamaan. (Richardson (2010))

3.2 Videokoodauksen tarpeellisuus

4 Rinnakkaislaskenta

Rinnakkaislaskennan (parallel computing) tavoite on parantaa laskennan suorituskykyä. Käsitteellisesti rinnakkaislaskentaa ei kannata sekoittaa samanaikaiseen laskentaan (concurrent computing). Akateemisessa mielessä jälkimmäinen tarkoittaa rinnakkaisen laskennan oikeellisuuteen, kun taas ensimmäinen keskittyy saamaan hajautetusta ja rinnakkaisesta laskennasta näkyviä hyötyjä. Tämä työ keskittyy rinnakkaisesta laskennasta saatuihin suorituskyvyn parannuksiin. (citation needed)

4.1 Rinnakkaisuuden peruskäsitteet

Rinnakkaisuutta voidaan jakaa kahteen osaan sen perusteella, mihin rinnakkaisuus kohdistuu. Tehtävärinnakkaisessa laskennassa rinnakkain suoritetaan erilaisia tehtäviä - esimerkiksi saman ohjelman eri säikeitä. Datarinnakkaisessa ohjelminnissa taas rinnakkaisuus syntyy siitä, että operoidaan samaan aikaan eri datan osa-alueilla. Datarinnakkainen lähestymistapa sopii varsin hyvin videokoodauksen rinnakkaistamiseen - jos esimerkiksi tehdään DCT, voisaan kukin blokki käsitellä erikseen (blokit riippumattomia toisistaan), joten saavutetut hyödyt laskenta-ajassa ovat huomattavat. (citation needed)

- asiaa race conditioneista, mutexeista, synkronoinnista?
- amdahl?

4.2 Rinnakkaislaskenta ja nykypäivän tietokonejärjestelmät

Nykypäivänä suurin osa laitteista, jotka pystyvät toistamaan videota ovat moniytimisiä tietokoneita - jopa joissakin televisioissa on nykykän kaksoisydinprosessoireita. Samalla prosessorit ovat kehittyneet suorittamaan useampia käskyjä kerrallaan (pipelining), joten rinnakkaisuus on ehdottomasti nykypäivää. (citation needed)

5 Videokoodaus ja rinnakkaislaskenta

5.1 Rinnekkaisuuden hyödyt videokoodaukselle

5.2 Erilaiset kiihdytysalustat ja ohjelmistoratkaisut

6 Yhteenveto

Lähteet

Harry Angelides, Marios C. & Agius. *Handbook of MPEG Applications : Standards in Practice*. Wiley, Chichester, West Sussex, United Kingdom, 2011. ISBN 978-0-470-75507-0 (nid.), 978-0-470-97458-2 (elektr.). 551 s.

Cisco. Cison ennustus mobiilidatan kehityksestä. http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html, tammikuu 2013.

Iain Richardson. *H.264 Advanced Video Compression Standard*. Wiley, Chichester, West Sussex, United Kingdom, toinen painos, 2010. ISBN 978-0-470-51692-8 (nid.), 978-0-470-98928-9 (elektr.). 348 s.

YouTube. Youtube-statistiikkaa. http://www.youtube.com/t/press_statistics, tammikuu 2013.