



TAMPEREEN TEKNILLINEN YLIOPISTO
Tietoliikenne-elektroniikan koulutusohjelma

KYÖSTI HERRALA
KASVOJEN PAIKANNUS
Kandidaatintyö

Tarkastaja: lehtori Heikki Huttunen
Jätetty tarkastettavaksi
1. helmikuuta 2009

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO
Tietoliikenne-elektroniikan koulutusohjelma

HERRALA, KYÖSTI: Kasvojen paikannus

Kandidaatintyö, 25 sivua

Helmikuu 2009

Pääaine: Signaalinkäsittely ja multimedia

Tarkastaja: lehtori Heikki Huttunen

Avainsanat: Kasvojen paikannus, konenäkö, koneoppiminen, hahmontunnistus, kaskadi, boosting, adaboost

Kasvojen paikannuksen tavoitteena on etsiä kuvasta ihmiskasvojen sijainti ja koko. Kasvohavaintojen tekeminen luotettavasti ja tehokkaasti on välttämätön esivaihe monissa käytännön sovelluksissa kuten ihmisen identiteetin, sukupuolen tai mielialan tunnistuksessa. Suurimpana ongelmana on havaintoluokan suuri sisäinen vaihtelu ja kuvaolosuhteiden vaikutus. Ongelmaan on sovellettu erilaisia lähestymistapoja, joista koneoppimiseen perustuvat menetelmät ovat viimeaikoina saavuttaneet lupaavia tuloksia ja eniten huomiota. Tämä työ esittelee yhden tämän lajin menetelmän yksityiskohdat ja taustalla olevan teorian. Lisäksi esitellään laajennokset kääntyneiden kasvojen tunnistamiseen ja väri-informaation hyväksikäyttämiseen. Tunnistimen hyvä tarkkuus yhdistettynä tehokkuuteen perustuu laskennan keskittämiseen potentiaalisiin kuva-alueisiin kaskadiksi hajoitettun luokittimen avulla. Luokitin käyttää hyödykseen integraalikuvan avulla nopeasti laskettavia yksinkertaisia Haar-piirteitä, jotka valitaan opetusaineston pohjalta AdaBoost-algoritmilla. Työn soveltavassa osuudessa testataan tunnistimen sietokykyä piirteiden peittymiselle ja tasorotaatiolle. Tuloksista havaitaan, että tunnistaminen häiriintyy vasta kun silmien kaltaisia kriittisiä alueita peitetään merkittävästi. Myös peittävän alueen tyyppillä on merkitystä. Tasorotaation sietokyky on minimissään $\pm 15^\circ$ ja yli puolessa tapauksista yli $\pm 25^\circ$.

ALKUSANAT

Tämä työ on laadittu osana Tampereen teknillisen yliopiston signaalinkäsittelynlaitoksen kandidaatti-seminaaria. Työn aihepiirin valinnassa innoittajana on toiminut muun muassa laitoksen oivaltavat hahmontunnistuskurssit sekä tietenkin työn konkreettisen aiheen ehdottanut ohjaaja Heikki Huttunen. Taustatukea ja kommentteja olen saanut myös perheeltäni varsinkin työn oikolukuvaiheessa. Siitä kiitokset heille.

1/2/2009

SISÄLLYS

Termit ja niiden määritelmät	1
1. Johdanto	2
2. Teoreettinen tausta	3
2.1. Katsaus erilaisiin menetelmiin	3
2.2. Uudelleenotanta hahmontunnistuksessa	4
2.3. Boosting-algoritmit	5
2.4. Adaptive boosting (AdaBoost).....	6
2.5. Luokittimen spesifisyys ja sensitiivisyys	7
3. Tunnistimen rakenne	9
3.1. Piirteet ja niiden laskeminen	9
3.2. Opetusnäyteaineisto	10
3.3. Luokitinalgoritmi	11
3.3.1. Rakenne	11
3.3.2. Komponenttiluokittimet.....	12
3.3.3. Kaskadin muodostaminen.....	13
3.4. Esi- ja jälkikäsitteily.....	14
3.4.1. Tasorotaatio ja profiilikasvot.....	14
3.4.2. Väri-informaation hyödyntäminen	15
3.4.3. Pällekkäiset havainnot.....	16
4. Tunnistus käytännössä	17
4.1. Virheanalyysi ja tarkkuuden arviointi	17
4.2. Piirteiden peittyminen	17
4.3. Tasorotaatio.....	18
4.4. Johtopäätökset.....	19
Lähteet.....	21

TERMIT JA NIIDEN MÄÄRITELMÄT

AdaBoost	Koneoppimisalgoritmi, joka muodostaa joukosta heikkoja luokittimia vahvan luokittimen käyttäen adaptiivisesti painotettuja opetusnäytteitä.
Bootstrap-otos	Uudelleenotannan avulla keinotekoisesti muodostettu joukko otoksia, joiden ominaisuuksilla voidaan tutkia ja muodostaa parempia estimaattoreita.
Haar-piirre	Kuvan suorakulmaisten alueiden pikselisummista laskettava tunnusluku, jota luokittelija voi hyödyntää.
HSV	Väriavaruus, jota on onnistuneesti käytetty muun muassa ihonvärin mallinnukseen. Koostuu värin sävystä (H=hue), kyllästyksestä (S=saturation) ja valoisuusarvosta (V=value)
Integraalikuva	Vakioajassa laskettava kuvan muunnos, jonka avulla voidaan tehokkaasti laskea Haar-tyyppisiä piirteitä eri skaaloissa ja sijainneissa.
Kaskadiluokitin	Päätöspuumainen rakenne peräkkäisiä binääri-luokittimia.
ROC-käyrä	Käyrä, jonka avulla luokitalgoritmin tehokkuutta voidaan arvioida tehtäessä erilainen kompromissi väärin ja oikeiden positiivisten luokitusten kesken.

1. JOHDANTO

Kasvojen paikannuksella pyritään etsimään syötteenä annetusta kuvasta siinä esiintyvien ihmiskasvojen sijainti ja koko. Viimeaikoina kuvainformaation tuottamista ja käsittelyä varten tarvittavat resurssit ovat kehittyneet tasolle, jolla vaativampi analysointi on tullut mahdolliseksi. Laitteiston ja kuvankäsittelyn hinta on romahtanut. Samalla aihepiirin tutkiminen ja algoritmien kehittäminen on merkittävästi lisääntynyt 1990-luvun loppupuolelta lähtien.

Kasvojen paikannus luo uusia mahdollisuuksia parantaa ihmisen ja koneen vuorovaikutusta monenlaisissa tietoteknisissä keksinnöissä. Erilaiset sulautetut järjestelmät tarjoavat houkuttelevia sovelluskohteita reaaliaikaiselle kuvankäsittelylle. Esimerkiksi nykyaikainen digitaalikamera saattaa optimoida kuvan tarkennuksen siten, että kuvattavan henkilön kasvat tarkentuvat oikein. Tarkennettuihin kohteisiin voidaan myös investoida suurempi bittivirta pakkauksen yhteydessä, jolloin niiden sisältämä oleellinen informaatio säilytetään. Tämän kaltaiset sovellukset korostavat kasvojen paikannuksen roolia kuvainformaation alkukäsittelyssä osana suurempaa kokonaisuutta.

Paikannusongelmasta tekee haastellisen erityisesti kasvojen monimuotoisuus tunnistettavana kohteena. Jo yksittäisen kasvoprototyypin tunnistus on hankalaa, koska kohde muuttuu merkittävästi ilmeen muuttuessa ja riippuen siitä onko kyseessä esimerkiksi profiili tai suoraan edestäpäin kuvattu kohde. Lisäksi kasvojen piirteisiin vaikuttavat ainakin valaistus ja kohdetta peittävät yksityiskohdat. Tunnistimen täytyy myös toimia erittäin nopeasti monissa reaaliaikaisissa sovelluksissa. Ongelman ratkaisuun vaaditaan kehittynyt hahmontunnistusjärjestelmä, jonka suunnittelu vaatii näiden ongelmien erityistä huomiointia.

Tämä työ esittelee erään kasvojentunnistusjärjestelmän rakenteen ja teoreettisen taustan sekä tutkii joitakin sen ominaisuuksia käytännössä. Pohjana tarkastelulle toimii jopa mullistavana pidetty Violan ja Jonesin menetelmä (2004), jonka ominaisuuksia punnitaan hahmontunnistusjärjestelmän suunnittelun eri osa-alueilla. Työn rakenne koostuu alussa kuvattavasta järjestelmän teoreettisesta taustasta ja sen osakomponenttien esittelystä. Työn lopuksi tutkitaan menetelmän toimintaa kahdessa yleisessä ja ongelmallisessa tilanteessa.

2. TEOREETTINEN TAUSTA

2.1. Katsaus erilaisiin menetelmiin

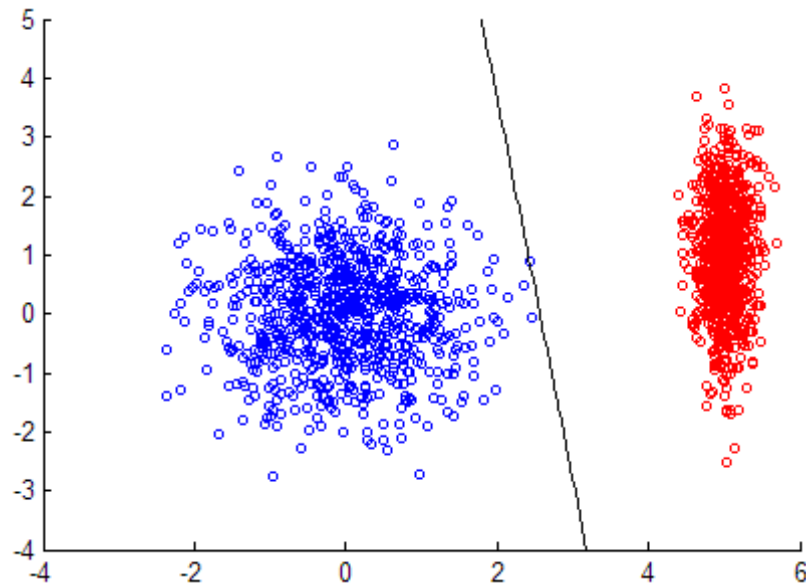
Kasvojen paikannukseen on sovellettu monia erilaisia lähestymistapoja. Uusimpien joukossa suosituimpia ja parhaimpia ovat olleet koneoppimista ja tilastollista hahmontunnistusteoriaa hyväksikäyttävät menetelmät. Suuri osa menetelmistä perustuu joko kuvasta laskettavien paikallisten piirteiden tai pikselitason harmaasävyjakauman holistiseen mallinnukseen. Erottavana tekijänä on, kuinka vahvasti mallin rakenne ja parametrit riippuvat suoraan raakadatasta tai järjestelmää suunnittelevan asiantuntijan tietämyksestä. Eroja on myös siinä, aloitetaanko kuvaa tarkastella pienistä yksityiskohdista vai suuremmista kokonaisuuksista. (Yang et al. 2002.)

Kasvojen paikannusmenetelmät voidaan luokitella neljään luokkaan (Yang et al. 2002). Tietämyspohjaisissa menetelmissä tunnistin rakennetaan esimerkiksi sumean logiikan sääntöjen avulla ilmaisemaan ihmisslähtöinen kuvaus kasvoista. Säännöillä asetetaan vaatimuksia tyypillisille kasvojen piirteille ja piirteiden suhteelle toisiinsa. Tällainen asiantuntijalähtöinen lähestyminen etenee yleisistä kohti yksityiskohtaisempia sääntöjä.

Paikallisten piirteiden etsimiseen perustuvat menetelmät suorittavat aluksi potentiaalisten kasvoalueiden rajauksen. Rajaus tapahtuu esimerkiksi perustuen kasvojen tyypillisiin ominaisuuksiin kuten tunnusomaiseen värisävyyn, tekstuuriin, symmetriaan tai muotoihin. Rajauksen jälkeen tunnistaminen tapahtuu ryhmiteltyjen paikallisten piirteiden avulla tutkimalla niiden keskinäistä asemaa.

Malliesimerkkeihin perustuvat menetelmät hakevat kasvojen sijainnin tutkimalla yhtäläisyyksiä vertailua varten tallennettuihin esimerkkeihin. Mallit voivat olla kiinteitä tai datan pohjalta parametrisoitavia. Malli voi rakentua myös osamalleista, jotka tunnistavat paikallisia piirteitä kuten silmät ja suu.

Viimeisen ryhmän menetelmiä muodostavat ne, joissa mallin rakenne ja parametrit perustuvat opetusnäyteaineistoon eli kasvokuviiin. Tässä työssä esiteltävä menetelmä kuuluu tähän ryhmään. Tunnistukseen käytettävät kuvan ominaisuudet löydetään koneoppimisen keinoilla. Ominaisuuksista muodostuu piirrevektoreita $\mathbf{x} \in \mathbb{R}^n$, joiden avulla luokitinalgoritmi muodostaa luokitteluhypoteesin erottelemaan kasvot ja vastaesimerkit. Opetusnäytteet $(\mathbf{x}_i, y_i) \in D_{opetus}$ koostuvat piirrevektorista \mathbf{x}_i ja tunnetusta luokasta $y_i \in \{-1 = ei\ kasvo, 1 = kasvo\}$. Kuva 1 on esitetty perceptron-algoritmilla opittuun luokitteluhypoteesiin liittyvä päätöspinta ja käytetyt opetusnäytteet. Perceptron-luokittimessa tunnistaminen perustuu etäisyyden laskemiseen hypertasosta, joka opitaan päivittämällä tason suuntavektoria opetusnäytteiden perusteella.



Kuva 1 Perceptron-tyyppisen binääriluokittimen opetusnäytteet ja päätöspinta

Tilastollisessa hahmontunnistuksessa taustalla on periaatteellinen idea kahden tai useamman luokkatiheysfunktion määrittämisestä esimerkiksi kasvojen ja vastaesimerkkien piirrevektoreille. Luokkatiheysfunktioista johdetaan vastaavat päätösalueet luokittimelle. Luokkatiheysfunktioiden estimoinnin ongelmana on kuvainformaation ulottuvuuksien suuri määrä, jos piirrevektorit muodostettaisiin suoraan kuvan yksittäisten pikselien arvoista. Sen vuoksi luokitus vaatii ulottuvuuksien määrän karsimista esimerkiksi projisoimalla data pienempään aliavaruuteen tai valikoimalla monimutkaisempia piirteitä (Yang et al 2002). Piirrevaruuden suuren koon vuoksi täytyy kiinnittää myös erityistä huomiota opetusnäytteiden keräämiseen.

2.2. Uudelleenotanta hahmontunnistuksessa

Tilastollista uudelleenotantaa käytetään tavallisesti esimerkiksi silloin, kun halutaan tutkia jakauman tilastollisen tunnusluvun estimaattorin ominaisuuksia. Odotusarvon estimaattina käytetty otoksen keskiarvo on tyypillisesti melko herkkä yksittäisille ääriarvopisteille (outliers). Sen sijaan esimerkiksi mediaanin arvo ei muutu yksittäisten otoksen pisteiden vuoksi herkästi.

Uudelleenotannan keinoilla voidaan myös pienentää otoksesta laskettavan estimaatin virheherkkyyttä, jolloin estimaattorin varianssi laskee. Yksinkertainen tapa laskea virhesietoisempi estimaatti odotusarvolle otoksen keskiarvon avulla on suorittaa ensin useampi uudelleenotanta. Aluksi alkuperäisen otoksen näytteistä (n kpl) muodostetaan uusia keskenään samankokoisia otoksia (m kpl) palauttavalla otannalla. Näitä kutsutaan bootstrap-otoksiksi. Niissä on mukana keskimäärin 63,2% alkuperäisistä näytteistä ja loput ovat kahdentuneita. Bootstrap-estimaatti jakauman parametrille saadaan keskiarvona uusille joukoille lasketuista normaaleista estimaateista. (Duda et al. 2001.)

Uudelleenotannalla on samankaltainen merkitys hahmontunnistuksessa kuin tilastollisessa estimoinnissa. Kahden luokan ongelmassa on usein saatavilla toisesta tunnistettavien näytteiden luokasta kattavasti näytteitä, joita kutsutaan positiivisiksi opetusnäytteiksi (kasvot). Sen sijaan toiseen luokkaan kuuluvista negatiivisista opetusnäytteistä (ei-kasvot) on vaikeampi muodostaa kattavaa otosta. Negatiivisten opetusnäytteiden valikoima on oletettavasti paljon laajempi ja niiden muodostaman päätösalueen tilavuus piirreavaruudessa epämääräisempi (Duda et al. 2001).

Täydellisen ja kattavan opetusaineiston muodostaminen ei kuitenkaan ole välttämätöntä edeltäkäsien. Varsinkin negatiivisia opetusnäytteitä voidaan täydentää siten, että niiden sisältämä informaatio on merkityksellistä. Luokitinta rakennettaessa eniten informaatiota on opetusnäytteillä, jotka sijaitsevat lopputuloksessa lähellä päätöspintaa (Duda et al. 2001). Käytännössä tämän tyyppisiä näytteitä löydetään tehokkaasti, jos negatiivisia opetusnäytteitä voidaan generoida helposti ja tehokkaasti. Alustavaa luokitinta testataan näillä näytteillä etsien väärin luokittuvia tapauksia. Nämä näytteet tulee huomioida kerätessä uutta edustavampaa opetusnäytteiden joukkoa, eli bootstrap-otosta. Uudella parannetulla aineistolla opetettu luokitin toimii luultavasti paremmin.

2.3. Boosting-algoritmit

Tietyllä algoritmilla voi olla huono taipumus sopeutua luokitusongelmaan riippuen käytetystä opetusnäytejoukosta. Yhteensopivuusongelman ydin ei välttämättä ole liian yksinkertainen luokitintyyppi vaan epästabiili algoritmi. Algoritmin sanotaan käyttäytyvän epästabiilisti, jos yksittäisten opetusnäytteiden pieni muutos aiheuttaa suuren muutoksen luokittimen toiminnassa tai rakenteessa. Tällaisesta käyttäytymisestä hyvänä esimerkkinä ovat lukuisat päätöspuualgoritmit. (Duda et al. 2001.)

Luokittimen herkkyyttä reagoida yksittäisiin näytteisiin epätoivotusti pyritään ehkäisemään yhdistelemällä useampia luokittimia, joiden opetusaineistot muodostetaan uudelleenotannalla. Tarkoituksena on vähentää epästabiiliusongelman ilmenemistä uudelleenotannan keinoilla. Komponenttiluokittimien äänestämällä yhdistetyn tuloksen on todettu empiirisesti olevan usein yksittäisiä virhesietoisempi.

Boosting on englanninkielinen termi, jolla tarkoitetaan hahmontunnistus-kirjallisuudessa yleispätevää tapaa parantaa yksittäisen, epästabiilin ja epätarkan luokitin-algoritmin toimintaa. (Duda et al. 2001.) Prosessin tarkoituksena on rakentaa iteratiivisesti sarja samantyyppisiä komponenttiluokittimia, joiden vahvuudet yhdistämällä saadaan yksittäistä luokitinta parempi tulos. Tavoitteeseen päästään esimerkiksi muokkaamalla opetusnäytteiden virheiden painokertoimia iteraatioiden välissä, jolloin eri iteraatioissa käytetään opetusnäytteitä uudelleen eri tavalla. Iteraatioissa otetaan mukaan aina kokonaisvirheen kannalta kaikista informatiivisimmat opetusnäytteet.

Boosting-menetelmän vaiheet ovat yksinkertaistettuna seuraavat:

- Oletetaan, että käytössä on joukko opetusnäytteitä $(\mathbf{x}_i, y_i) \in D_{\text{opetus}}, i \in (1, n)$, missä y_i on piirvektoriin \mathbf{x}_i kuuluva luokka.
- Toistetaan kierroksilla $t = 1, \dots, T$:
 - o Opetusnäytteistä valitaan uudelleenotannalla $D_t \subset D_{\text{opetus}}$.
 - o Heikko luokitusalgoritmi muodostaa hypoteesin h_t , joka minimoi luokitusvirheen joukolle D_t .
- Lopuksi heikot hypoteesit yhdistetään vahvaksi luokittelu-säännöksi.

Hahmontunnistustermein ilmaistuna tarkoituksena on rakentaa yksi vahva luokitin hyödyntäen heikkoja luokittimia. Heikko luokitusalgoritmi on sellainen, jonka antama tulos on ainakin yhtä tarkka kuin sattumanvaraisesti valittava luokitus tulos. Kahden luokan tapauksessa heikko luokitin oppii tunnistamaan opetusnäytteistä siis vähintään yli 50% oikein. Heikoksi luokittimeksi valitaan yleensä yksinkertainen menetelmä kuten perceptron-luokitin. Oletettavasti paraskaan perceptron-luokitin ei pysty ratkaisemaan alkuperäistä luokitusongelmaa kiitettävästi, koska näytteet eivät useinkaan separoidu lineaarisesti. Optimaalinen heikko luokitin on silti vaivattomasti muodostettavissa.

Vahva luokitusalgoritmi muodostuu yhdistelemällä heikkojen komponenttiluokittimien tulos. Tämä voi tapahtua esimerkiksi äänestämällä tai asettamalla kynnsarvo heikkojen luokittimien ulostulojen painotetulle summalle. Komponenttiluokittimet muodostetaan ratkaisemalla sarja toinen toistaan vaativampia tehtäviä heikolla luokittimella. Uusia komponenttiluokittimia lisätään parantamaan lopputulosta näytteillä, joiden valinnassa on painotettu edeltävien luokittimien tekemiä virheitä.

2.4. Adaptive boosting (AdaBoost)

AdaBoost on Freundin ja Schapiren (1996) esittelemä käytännöllinen boosting-algoritmi. Adaptiivisuudella algoritmin nimessä viitataan sen keskeiseen ideaan ylläpitää opetusnäytteiden painokertoimia, joita säätelemällä heikon luokittimen huomio ohjataan väärin luokitteviin hankaliin näytteisiin. Algoritmi tuottaa myös painokertoimet muodostuville komponenttiluokittimille, joiden avulla niiden suhteellinen tarkkuus otetaan huomioon hypoteesien yhdistelyvaiheessa.

Kahden luokan tapauksessa algoritmi on seuraavanlainen:

- Oletetaan, että käytössä on joukko opetusnäytteitä $(\mathbf{x}_i, y_i) \in D_{\text{opetus}}, i = 1 \dots n$, missä luokka $y_i \in \{-1, +1\}$.
- Alustetaan painokertoimet: $D_1(i) = 1/n$
- Toistetaan kierroksittain $t = 1, \dots, T$:
 - o Muodostetaan painotetun opetusvirheen ε_j minimoiva heikko luokitin:

$$h_t = \arg \min_{h_j \in H} \varepsilon_j$$

- Saadulla hypoteesilla $h_t : X \rightarrow \{-1, +1\}$ on opetusvirhe

$$\varepsilon_t = P_{i \sim D_t}(h_t(x_i) \neq y_i) = \sum_{i=1}^n D_t(i) [y_i \neq h_t(x_i)] < 0.5$$

- Valitaan $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$
- Päivitetään:

$$\begin{aligned} D_{t+1}(i) &= \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} \text{ jos } h_t(x_i) = y_i \\ e^{\alpha_t} \text{ jos } h_t(x_i) \neq y_i \end{cases} \\ &= \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \end{aligned}$$

jossa Z_t on normalisointi vakio, jonka avulla D_{t+1} on jakauma.

- Lopullinen hypoteesi on:

$$H(x) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(x_i)\right)$$

On osoitettu että AdaBoost-algoritmin opetusvirhe on ylhäältä rajoitettu, ja raja lähestyy iteraatio kerrallaan eksponentiaalisesti kohti nollaa. Käytännössä opetusongelman hankaluus kasvaa kierroksittain, ja lopulta heikko luokitin ei välttämättä pysty tuottamaan sattumaa parempaa tulosta. Tällöin algoritmi on pysäytettävä. AdaBoost:in on todettu kokeellisesti päätyvän ratkaisuun, joka toimii hyvin myös ennalta näkemättömällä aineistolla, eli ratkaisu yleistyy hyvin käytännössä. Menetelmällä ei ole havaittu myöskään taipumusta ylioppimiseen, joka ilmeisesti johtuu kyvystä maksimoida päätöspinnan etäisyys näytteisiin eli marginaali.

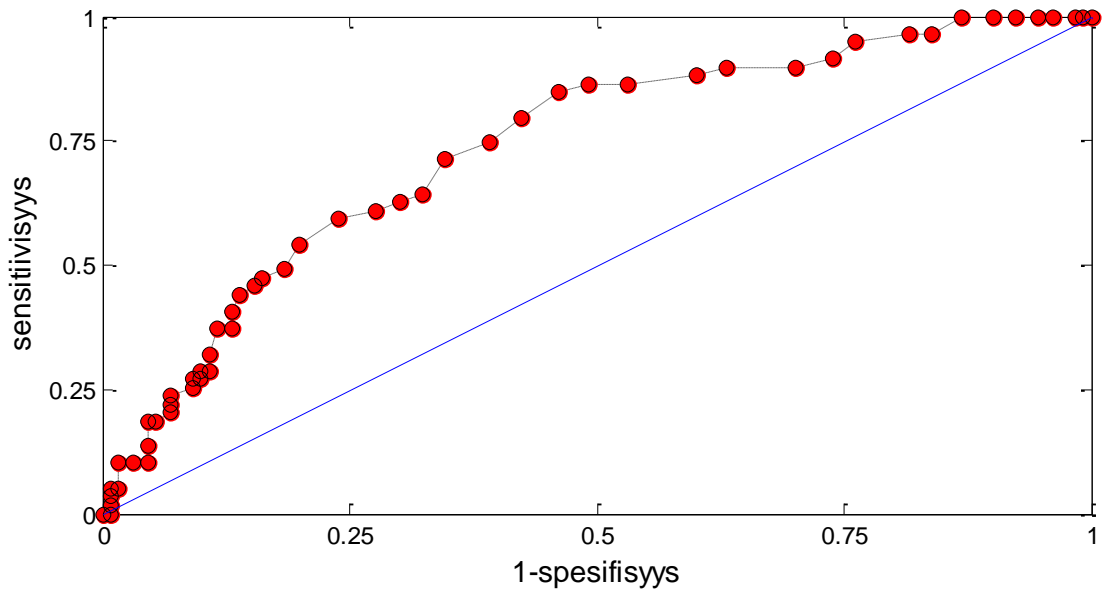
2.5. Luokittimen spesifisyys ja sensitiivisyys

Luokittimen tarkkuutta voidaan tarkastella monipuolisesti ROC-käyrän (receiver operating characteristic) avulla (Duda et al. 2001). Kahden luokan tapauksessa luokitin ajatellaan ilmaisimeksi, joka havainnoi jotakin ominaisuutta positiivisella tai negatiivisella ilmaisulla. Luokitin voi päätyä neljään erityyppiseen havaintoon, joiden todennäköisyydet riippuvat käytetystä toimintapisteestä. Positiivinen ilmaisu voi olla tietyllä todennäköisyydellä oikea tai väärä, kuten myös negatiivinen ilmaisu. ROC-käyrän avulla tarkastellaan luokittimen spesifisyyttä ja sensitiivisyyttä eri toimintapisteissä. Testiaineiston perusteella nämä tunnusluvut määritellään seuraavasti.

$$\text{Sensitiivisyys} = \frac{\text{Oikeat positiiviset havainnot}}{\text{Oikeat positiiviset havainnot} + \text{Väärät negatiiviset havainnot}}$$

$$\text{Spesifisyys} = \frac{\text{Oikeat negatiiviset havainnot}}{\text{Oikeat negatiiviset havainnot} + \text{Väärät positiiviset havainnot}}$$

Yleensä luokitinalgoritmin toimintapistettä on mahdollista muuttaa säätämällä esimerkiksi virhetyyppien painokertoimia tai kynnyсарvoa todennäköisyysjakaumiin perustuvissa menetelmissä. Luokittimen parametreja muuttamalla voidaan testiaineistolla tuottaa mielivaltaisen pieni väärin tai oikeiden positiivisten havaintojen osuus eli toimintapiste. Tyypillisten luokkatiheysfunktioiden jatkuvuudesta johtuen tuloksena saatavien väärin ja oikeiden positiivisten havaintojen osuuksista saadaan piirrettyä jatkuva käyrä (Kuva 2). Tämän ROC-käyrän vaaka-akselilla on väärin havaintojen osuus eli 1 - spesifisyys ja pystyakselilla oikeiden havaintojen osuus eli sensitiivisyys.



Kuva 2 Tyypillinen kokeellisesti määritetty luokittimen ROC-käyrä

Idealisesti ROC-käyrän alle jäävä pinta-ala on mahdollisimman suuri, kun taas satunnaisesti arvauksen tekevän luokittimen ROC-käyrä kasvaa diagonaalisesti. Teoreettisesti on olemassa ideaalinen käytetyistä piirteistä riippuva ROC-käyrä, johon voidaan pyrkiä estimoimalla luokkatodennäköisyyksiä. Sen avulla voitaisiin tarkastella luokittimesta riippumatonta havaintojen erottelukykä. Käytännössä edes jakaumien muotoa ei usein tunneta.

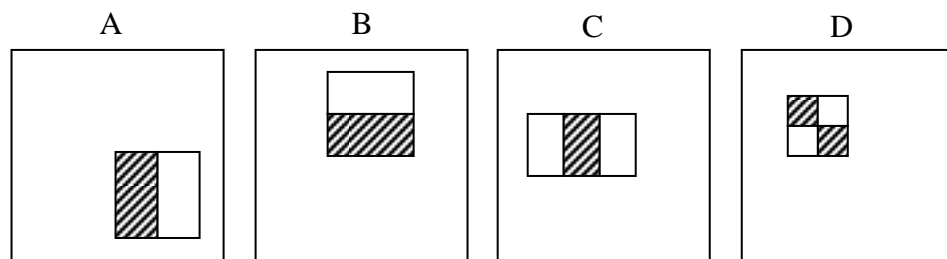
Empiirisesti muodostettua ROC-käyrää voidaan luokitinalgoritmien vertailun lisäksi käyttää toimintapisteen valinnassa. Tunnistimen tarkkuus on parhaimmillaan toimintapisteissä, jotka sijaitsevat lähellä kuvan vasenta ylänurkkaa. Siirtämällä pistettä käyrällä oikealle voidaan saavuttaa kompromissi eli lähes maksimaalinen sensitiivisyys siedettävällä spesifisyyden menetyksellä.

3. TUNNISTIMEN RAKENNE

3.1. Piirteet ja niiden laskeminen

Violan ja Jonesin menetelmässä (2004) kasvojen tunnistaminen perustuu harmaasävykuvasta laskettaviin äärimmäisen yksinkertaisiin piirteisiin, joita kutsutaan Haar-piirteiksi, koska ne muistuttavat Haar-kantafunktioita. Piirteiden laskeminen suoritetaan tarkasteluikkunan sisältä, jota liikutetaan kuvan yllä samalla kasvattaen sen kokoa vaiheittain. Jos opetusnäytteenä käytetyt näytteet ovat olleet kokoa 24 kertaa 24 pikseliä, niin aluksi voidaan suorittaa tunnistus tämän kokoisella ikkunalla ja skaalata sitä kertoimella $s = 1,25$ vaiheittain. Ikkunan siirtäminen voidaan toteuttaa aluksi muutaman pikselin Δ tarkkuudella ja jatkossa kerrottuna skaalauskerrotoimella $s \cdot \Delta$.

Piirteinä käytetään erotuksia, jotka saadaan aikaan vähentämällä toisistaan kuvassa olevien suorakulmaisten alueiden pikseliarvojen summia. Toiminnallisesti piirteet keräävät paikallista tietoa reunamuodostumista. Piirteitä on kolmen tyyppisiä, ja niissä suorakulmaiset alueet ovat akseleiden suuntaisesti vierekkäin ja samankokoisia. Kahden suorakulmion piirteessä (Kuva 3a ja 3b) alueet ovat leveys- tai pystysuunnassa päällekkäin. Kolmen suorakulmion piirteessä (Kuva 3c) uloimpien alueiden pikselisummista vähennetään keskimmäisen pikselisumma. Neljän suorakulmion piirteessä (Kuva 3d) diagonaalisten alueiden summat vähennetään toisistaan.

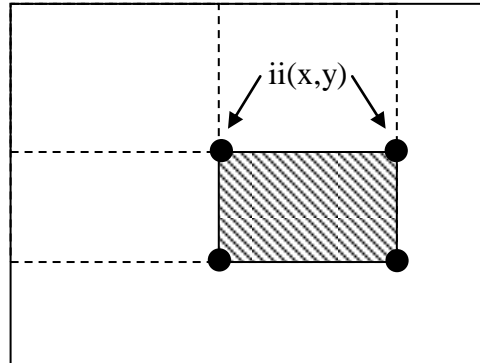


Kuva 3 Violan ja Jonesin menetelmässä käytetyt Haar-piirteet

Näiden piirteiden käyttö aiheuttaa päällekkäisen informaation muodostumista, koska muodostuva piirrevalikoima on käytännössä aina suurempi kuin kuvan pikseleiden lukumäärä. Piirteiden avulla muodostuva kuvan esitystapa on niin sanotusti redundantti. Lopullinen tunnistin ei kuitenkaan käytä kuin muutamia valikoituja piirteitä. Käytettävien piirteiden tyyppillä on ensisijaisesti painotettu niiden nopeaa ja helppoa laskentaa. Toisaalta piirrejoukon redundanssi voi parantaa luokitusongelman yhteensopivuutta luokittimen kanssa.

Aikaisemmat järjestelmät ovat käyttäneet skaalaongelman ratkaisuun kuvapyramidiä (Sung & Poggio 1998). Se tarkoittaa sitä, että alkuperäisestä kuvasta muodostetaan alinäytteistyksellä uusia pienempiä kopioita. Skaalattujen kuvien

laskeminen on melko raskas operaatio ja sitä ei ole välttämättä mahdollista toteuttaa reaaliaikaisesti matalatehoisessa sulautetussa järjestelmässä. Kuvapyramidin laskemiselta vältytään, kun käytettävät piirteet voidaan laskea vakioaikaisesti skaalauksesta riippumatta (Viola & Jones 2004).



Kuva 4 Pikselisumman laskeminen integraalikuvan avulla

Violan ja Jonesin menetelmän piirteiden nopea laskenta perustuu tekijöiden käyttämään vaihtoehtoiseen tapaan esittää kuvan sisältämä informaatio. Integraalikuva (Kuva 4) nimetyn esitystavan avulla voidaan jokainen tunnistimen tarvitsema suorakulmio-piirre laskea vakioaikaisesti. Piirteen sijainti tai skaala ei vaikuta laskenta-aikaan, koska arvot saadaan yksinkertaisesti indeksoimalla integraalikuvaa suorakulmioiden reunapisteissä. Myös kuvan muuntaminen integraali-kuvaksi tapahtuu yksinkertaisilla operaatioilla vakioidussa ajassa. Integraalikuva on samankokoinen kuin alkuperäinenkin, ja sen yksittäisen pikselin $ii(x,y)$ arvo saadaan summaamalla kuvan $i(x,y)$ pikseleitä origoa kohden seuraavasti:

$$ii(x,y) = \sum_{x' \leq x, y' \leq y} i(x', y')$$

3.2. Opetusnäyteaineisto

Opetusnäytteiden kerääminen on olennainen osa Violan ja Jonesin menetelmää. Opetusnäyteaineisto koostuu harmaasävyisistä kasvokuvista sekä laajasta joukosta vastaesimerkkejä. Käyttötilanteesta riippuen tunnistimen opetukseen käytetään yleensä vain suoraan edestä oikeinpäin kuvattuja kasvoja ja esimerkiksi profiilikasvokuvien tunnistamiseen luodaan erillinen tunnistin. Näitä tunnistintyypppejä voidaan mahdollisesti käyttää kokonaisjärjestelmässä rinnakkain esikäsittelyn jälkeen.

Kasvokuvat täytyy rajata yhtenevästi esimerkiksi silmien ja leuan muodostaman kolmion perusteella. Rajatut kuvat skaalataan esimerkiksi kokoon 24 kertaa 24 pikseliä, josta muodostuu tunnistimen perusresoluutio. Pienet erot kuvien rotaatioissa kuvausakselin suhteen eivät ole ongelma, ja monesti uusia opetusnäytteitä luodaan keinotekoisesti kuvaa pyörittämällä. Tällä tavalla saavutetaan tunnistin, joka sallii

hieman enemmän vaihtelevuutta kohteissa. Helposti lisää näytteitä saadaan myös peilaamalla kuvia pystyakselin suhteen. (Sung & Poggio 1998.)

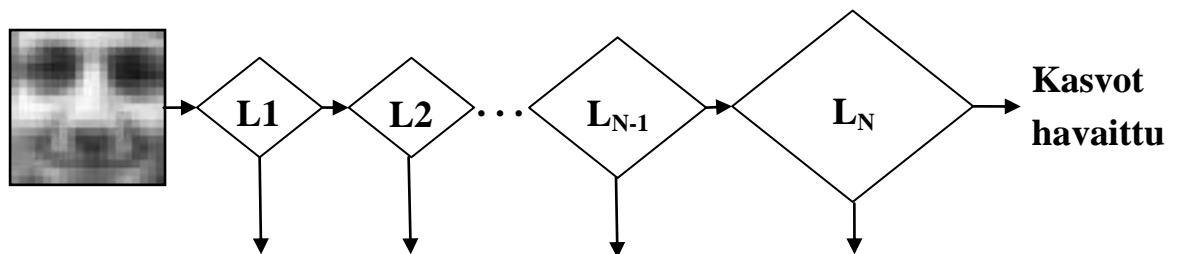
Kasvonäytteitä tulee kerätä riittävän kattavasti, jotta esimerkiksi silmälasien ja naamakarvoituksen mahdollisuus voidaan huomioida. Erilaisten valaistusten ja kameran kontrastin aiheuttamaa poikkeamaa voidaan vähentää esimerkiksi histogrammin normalisoinnilla. Joskus käytetään myös valogradientin poistoa sovittamalla valoisuusarvoihin tason funktio ja poistamalla sen vaikutus. Kuvista voidaan poistaa myös ylimääräinen tausta. (Sung & Poggio 1998.) Viola ja Jones (2004) ovat esikäsittelleet kuvat sekä opetus- että tunnistusvaiheessa normalisoimalla varianssin, jonka laskennassa integraalikuva on myös hyötyä.

Negatiivisten esimerkkien kerääminen on mahdollista suorittaa ottamalla erikokoisia palasia satunnaisesti kuvista, joissa kasvoja ei ole. Käytännössä tällä tavalla kerätyllä aineistolla ei vielä saavuteta parhaita tulosta, koska olennaisinta on löytää negatiivisia esimerkkejä, jotka luokituvat herkimmin väärin. Tällaiset näytteet sijaitsevat lähellä lopullisen luokittimen päätöspintaa. Yleinen tapa parantaa luokittinta on suorittaa opetusvaihe toistamiseen keräämällä seuraavan luokittimen opetusaineistoon näytteitä, jotka luokituvat väärin testattaessa erillisellä testiaineistolla. Tähän tarvittavaa testiaineistoa voidaan generoida kasvottomista kuvista yleensä helposti.

3.3. Luokitinalgoritmi

3.3.1. Rakenne

Violan ja Jonesin (2004) esittämän tunnistimen merkittävä nopeus etu aikaisempiin saavutetaan keskittämällä analysointi kuvan kaikkein potentiaalisimpiin osiin. Valtaosa selattavista tarkasteluikkunoista ei oletettavasti sisällä lainkaan kasvoja, joten on tärkeää säästää laskentaresursseja siirtymällä seuraavaan tarkasteluikkunaan tarpeeksi pian. Samalla täytyy kuitenkin saavuttaa riittävän korkea sensitiivisyys, jottei oikeita kasvoja hylätä liian aikaisessa tarkasteluvaiheessa.



Tarkasteluikkunassa ei kasvoja

Kuva 5 Luokitinkaskadi, jossa loppuosan komponenttiluokittimet L_{N-1} , L_N ovat monimutkaisempia

Laskentaresurssien käyttöä rajoitetaan rikkomalla raskas yksittäinen luokitin useammaksi komponenttiluokittimeksi (Kuva 5). Peräkkäisistä binääriluokittimista

muodostuu kaskadi, jossa eri vaiheilla on erilaiset suunnittelukriteerit. Alkuvaiheen luokittimet ovat kaskadissa yksinkertaisempia ja niiden toiminta ei vaadi monen piirteen laskemista tarkasteluikkunasta. Seuraavan vaiheen luokitin saa tarkasteltavaksi edellisen hyväksymät tapaukset, joten luokitteluongelma vaikeutuu, ja suurempi määrä piirteitä tarvitaan päättelyyn. Komponenttiluokittimien suunnittelu ja rakenne selitetään seuraavassa luvussa.

Kaskadirakenteen tehokkuus perustuu siihen, että on mahdollista rakentaa nopea luokitin, joka tunnistaa lähes kaikki kasvot, ja samalla spesifisyys on tyydyttävällä tasolla. Yksittäisen komponenttiluokittimen monimutkaisuus voidaan pitää kurissa rajoittamalla käytettyjen piirteiden määrää opetusvaiheessa, ja tarvittava spesifisyys saavutetaan valitsemalla sopiva toimintapiste ROC-käyrän avulla.

Tyypillisesti kasvojen paikannusjärjestelmältä vaaditaan yli 90 % sensitiivisyyttä ja spesifisyyttä, jolla väärin havaintojen osuus on kertaluokassa välillä 10^{-5} ja 10^{-6} . Tämä vastaa keskimäärin noin yhden väärän havainnon tekemistä 320 kertaa 240 pikselin kuvan tarkasteluikkunoilla. Se on siedettävän tarkkuuden alaraja, koska monissa kuvissa on vain yhdet kasvot. Tavoitteiden saavuttamista helpottaa havainto, että esimerkiksi kymmenen vaiheen pituisessa kaskadissa yksittäisen komponenttiluokittimen täytyy saavuttaa noin 99 % sensitiivisyys ja ainoastaan 70 % spesifisyys. (Viola & Jones 2004.) Tämä voidaan päätellä siitä, että kaskadin sensitiivisyys D ja väärin positiivisten havaintojen osuus F saadaan komponenttiluokittimien vastaavista arvoista kaavoilla:

$$F = \prod_{i=1}^K f_i$$

$$D = \prod_{i=1}^K d_i$$

Perimmäinen syy kaskadirakenteen tehokkuuteen on se, että vain murto-osa tarkasteluikkunoista joudutaan analysoimaan kaskadin loppuvaiheen monimutkaisemmilla luokittimilla, koska kasvojen esiintyminen on erittäin harvinainen tapahtuma.

Vertailtaessa yhdestä osasta koostuvaa luokitinta ja kaskadia ei luokitusvirheissä ole juuri eroa. Sen sijaan nopeusero on valtava johtuen laskettavien piirteiden määrästä. (Viola & Jones 2004.) Tyypillisellä testiaineistolla tunnistaminen vaatii kaskadin avulla keskimäärin vain kahdeksan piirteen laskemisen (Jones & Viola 2003).

3.3.2. Komponenttiluokittimet

Violan ja Jonesin (2004) tunnistin pohjautuu oletukseen, jonka mukaan tehokas ja tarkka kasvojen tunnistaminen on mahdollista pienellä joukolla hyvin valikoituja Haar-tyyppisiä piirteitä. Piirreavaruuden valtavan koon ja redundanssin vuoksi ei ole mielekäästä tai tarpeen laskea kaikkien piirteiden arvoja turhaan. Varsinainen ongelma

on valikoida opetusaineiston perusteella kaikkein informatiivisimmat piirteet. Piirteiden laskeminen on tehokkuussyistä aloitettava kriittisimmistä yksityiskohdista.

Oleelliset piirteet voidaan tunnistaa soveltaen AdaBoost-algoritmia. Tämä onnistuu käyttämällä heikkoja luokittimia, joiden toiminta perustuu vain yhden Haar-piirteen arvoon. Tavanomaisesti boosting-algoritmi kasvattaa heikkojen ja erikoistuneiden luokittimien joukkoa, joiden vahvuudet kompensoivat toistensa heikkouksia. Käytettäessä yhden piirteen heikkoja luokittimia kasvatetaan erikoistuneiden piirteiden joukkoa. Tuloksena saadaan suuresta piirreavaruudesta valikoitua tehokkaita piirteitä erittäin aggressiivisesti.

Heikko komponenttiluokitin koostuu yhdestä Haar-tyyppisestä piirteestä f , kynnyksarvosta θ , etumerkistä p , tarkasteluikkunasta $x \in \mathbb{R}^{24 \cdot 24}$ ja epäyhtälöstä. Se on muotoa:

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{jos } pf(x) < p\theta \\ 0 & \text{muulloin} \end{cases}$$

Minimivirheen tuottavan luokittimen löytäminen vaatii runsaasti laskentaa, koska aluksi määritetään optimaaliset kynnyksarvot jokaiselle mahdolliselle piirteelle ja sitten valitaan paras piire (Viola & Jones 2004). Laskennan kertaluokka on $O(NT \log N)$, jossa N on näytejoukon koko ja T piirreavaruuden koko. Tästä johtuen opetusvaihe saattaa kestää jopa viikkoja ilman rinnakkaislaskentaa. Tähän opetusvaiheen pullonkaulaan on myöhemmin kiinnitetty huomiota, ja esilaskennan avulla on saavutettu kertaluokka $O(NT)$, jolloin on saavutettu 15-60 -kertainen nopeus. Parhaimmillaan valitsemalla piirteet perustuen nopeasti estimoitaviin tilastollisiin tunnuslukuihin, on päästy kertaluokkaan $O(Nd^2 + T)$ varsin kohtuullisella muistin käytöllä. (Pham & Cham 2007)

Menetelmän ensimmäisenä valitsemat piirteet ovat tulkittavissa intuitiivisesti. Violan ja Jonesin (2004) luoma tunnistin tarkastelee esimerkiksi ensin silmien alueen suhteellista tummuutta poskien ja nenän alueeseen nähden. Seuraavaksi kiinnitetään huomiota nenän alueen valoisuuteen reunustaviin silmiin nähden. Kokonaisuudessaan heidän tunnistimensa koostuu 6060 Haar-tyyppisestä piirteestä, jotka ovat jakautuneet 38 erilliseen komponenttiluokittimeen.

3.3.3. Kaskadin muodostaminen

Kaskadin rakenteen suunnittelu on haastava optimointitehtävä. Komponenttiluokittimien ja niissä käytettävien piirteiden lukumäärän sijasta tyydytään yleensä asettamaan vaatimuksia sensitiivisyydelle ja spesifisyydelle eri vaiheissa. Kaskadiin lisätään vähitellen uusia komponenttiluokittimia, joiden käyttämien piirteiden määrää kasvatetaan AdaBoost-algoritmilla, kunnes asetetut tarkkuusvaatimukset täyttyvät. Tarkkuutta testataan erillisen testinäyteaineiston avulla, jota ei käytetä opetuksessa.

Ensimmäisessä vaiheessa opetusaineiston kasvoja sisältämättömät negatiiviset esimerkit on kerätty sattumanvaraisesti. Tämä joukko ei oletettavasti ole tilastollisesti

kovinkaan edustava, koska kuvainformaation dimensio on niin suuri. Seuraavassa vaiheessa voidaan käyttää hyväksi ensimmäisen tason vielä väärin tunnistamia kohteita, joita löydetään generoimalla lisää negatiivisia esimerkkejä. Teoriassa olisi mahdollista kasvattaa alkuperäistä negatiivisten esimerkkien joukkoa riittävästi heti alkuvaiheessa, mutta laskennallisesti se ei ole mahdollista. Suoritettu uudelleenotanta mahdollistaa moninkertaisen opetusaineiston käytön, koska suuri osa mahdollisista näytteistä voidaan poissulkea hyödyttöminä kaskadin myöhemmissä opetusvaiheissa olemassaolevan luokittimen osan avulla.

3.4. Esi- ja jälkikäsitely

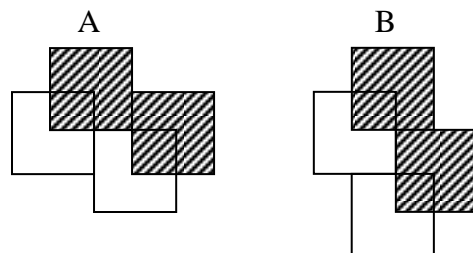
3.4.1. Tasorotaatio ja profiilikasvot

Tunnistimen perusversio on erikoistunut vain suoraan edestä otettuihin kuviin kasvoista. Muista suunnista otettuja kuvia varten voidaan rakentaa erikseen omat tunnistimet vastaavilla opetusnäytteillä. Tällöin jokainen tarkasteluikkuna voitaisiin syöttää eri kuvaussuuntiin erikoistuneiden tunnistimien läpi ja yhdistää havainnot. Toinen tapa tasorotaation tapauksessa on käyttää samaa tunnistinta, mutta suorittaa laskenta useilla käännettyillä tarkasteltavan kuvan versioilla. Kääntämiseen tarvittava laskenta on kuitenkin niin raskasta, että se dominoisi käytetyssä ajassa. (Jones & Viola 2003.)

Tehokkaampi lähestymistapa on suorittaa kasvojen tunnistus kahdessa vaiheessa. Aluksi toinen luokitin antaa arvion kuvaussuunnasta, jonka perusteella valitaan seuraavaksi käytettävä erikoistunut tunnistin. Jos kuvassa ei ole lainkaan kasvoja niin valittu tunnistin on satunnainen, jonka ei pitäisi vaikuttaa järjestelmän toimintaan heikentävästi.

Kuvaussuunnan arviointiin käytettävän menetelmän tulee olla erittäin tehokas, kuten varsinainen tunnistinkin on. Sopiva luokitin voidaan muodostaa päätöspuun (Quinlan 1993) avulla, joka käyttää syötteenä samoja Haar-piirteitä. Myös neuroverkkojen käytöstä on saatu hyviä tuloksia vastaavassa tehtävässä (Rowley et al. 1996). Päätöspuun rakentaminen suoritetaan ilman harvennusvaihetta ja lehtisolmujen jakamista jatketaan kunnes jäljellä on yksi näyte tai haluttu maksimisyvyys on ylitetty.

Edellä kuvattujen piirteiden joukkoa täytyy täydentää kiertyneitä kuvia varten. Diagonaalisten Haar-piirteiden avulla saavutetaan parempia tuloksia. Ne muodostuvat neljän suorakulmaisen alueen erotuksista (kuva 6). (Jones & Viola 2003.)



Kuva 6 Diagonaaliset Haar-piirteet

Tarvittavien eri kuvaussuuntiin erikoistuneiden tunnistimien määrä riippuu niiden sietämän rotaatiovääristymän määrästä. Jos esimerkiksi tunnistus onnistuu vielä käännettäessä kuvaa $\pm 15^\circ$, tarvitaan ainakin 12 tunnistinta. Käytännössä jokaista tunnistinta ei tarvitse muodostaa erikseen. Aluksi kolmen tunnistimen avulla voidaan kattaa astevälit $-15^\circ-15^\circ$, $15^\circ-45^\circ$ ja $45^\circ-75^\circ$. Puuttuvat yhdeksän luokitinta saadaan suoraan näistä laskennallisesti akseleiden suuntaisia piirteitä käyttävistä luokittimista.

3.4.2. Väri-informaation hyödyntäminen

Silloin kun tutkittavasta kuvasta on saatavilla väri-informaatio, voidaan kasvojen tunnistamista tehostaa etsimällä ihonvärisiä kohteita. Ihonväriä ei toki esiinny vain kasvoissa, mutta yhdistettynä muuhun analyysiin, voidaan karsia selkeitä tunnistusvirheitä. Ihonväriä voidaan etsiä jälki- tai esikäsittelevä vaiheessa, jolloin myös vähennetään tutkimista vaativien kuvan alueiden määrää. Violan ja Jonesin tunnistimen tapauksessa suurempi etu saavutetaan jälkikäsittelevä vaiheessa, jolloin voidaan uhrata enemmän aikaa luokitustarkkuutta parantavaan jatkokäsittelyyn.

Tutkimuksissa on selvinnyt, että suurin erottava tekijä eri ihmisten ihonväreissä on valoisuusarvo kuvattaessa valkoisessa valossa. Värikomponenttien arvot esimerkiksi HSV-väriavaruudessa (hue, saturation, value) muodostavat tiheän ryhmittymän, jos valoisuuskomponentti V jätetään huomioimatta. Ihonvärijakauman mallinnus tässä kaksiulotteisessa avaruudessa on osoittautunut mahdolliseksi monihuippuisen normaalijakauman avulla. Sovittaminen tapahtuu EM-algoritmillä. Sovitetun jakauman tiheysfunktio on:

$$P(x, \theta) = \sum_{i=1}^k \alpha_i \frac{1}{\sqrt{(2\pi)^d |\Sigma_i|}} \exp \left\{ -\frac{1}{2} (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right\},$$

jossa $\theta = \{\alpha_i, \mu_i, \Sigma_i\}$, $x = (H, S)$. (Huynh-Thu et al. 2002.)

Ihonvärimallin avulla voidaan laskea uskottavuusarvot kuvan pikseleille. Kuvan kynnystäminen iho- ja muihin alueisiin tapahtuu asettamalla sopiva kynnysarvo uskottavuudelle. Kynnysarvot voidaan säätää adaptiivisesti jakauman eri komponenteille. Tämä tapahtuu pienentämällä kynnysarvoa ja tutkimalla muutosta, joka tapahtuu ihoksi luokiteltavan alueen koossa. Sopivalla arvolla pinta-alan muutos on pienin. (Huynh-Thu et al. 2002.)

$$\text{pikseli} = \begin{cases} \text{ihoa,} & \text{jos } P(x, \hat{\theta}) > \text{kynnysarvo} \\ \text{ei ihoa,} & \text{muulloin} \end{cases}$$

Violan ja Jonesin (2004) tunnistimen väärien havaintojen määrää voidaan karsia jatkokäsittelmällä tarkasteluikkunat ihonvärimallin avulla. Aluksi havaittujen alueiden pikselit luokitellaan kynnystämällä. Sitten lasketaan ihonvärin osuus mahdollisella kasvoalueella. Jos osuus on riittävän suuri niin alue luokitellaan kasvoiksi.

3.4.3. Päälekkäiset havainnot

Tunnistimen käyttö aiheuttaa vääjäämättä päällekkäisiä havaintoja samoista kasvoista, koska eri skaalauksella toimivat tunnistimet sietävät pieniä skaalavääristymiä. Kasvohavaintojen järkevä jatkokäsittely esimerkiksi biometrisinä tunnisteina vaatii tarkkaa kohdistamista. Tämän vuoksi havainnot on ryhmiteltävä ja yhdistettävä.

Havaintojen yhdistäminen tapahtuu aluksi ryhmittelemällä havainnot, joita rajaavien suorakulmioiden yhteinen pinta-ala on asetettua kynnysarvoa korkeampi. Yhdistäminen tapahtuu laskemalla keskiarvot ryhmien havaintojen keskipisteille ja leveyksille.

Päälekkäisten havaintojen yhdistäminen sisältää riskin, jos kahdet kasvot esiintyvät kuvassa oikeasti päällekkäin. Tällöin yhdistettyjen havaintojen keskipiste on väärin kohdistettu kasvojen välille, joten molempien kasvojen paikannus on epäonnistunut.

4. TUNNISTUS KÄYTÄNNÖSSÄ

4.1. Virheanalyysi ja tarkkuuden arviointi

Violan ja Jonesin tunnistimeen väitetään sisältyvän monia hyviä piirteitä. Sen sanotaan olevan monia aikalaisiaan nopeampi, tarkempi ja virhesietoisempi. Tunnistimesta on saatavilla kokonainen ROC-käyrä, jolla sitä voidaan vertailla muihin menetelmiin, joista on usein saatavilla tarkkuuslukema vain yhdessä toimintapisteessä. (Viola & Jones 2001.) ROC-käyrän pohjana toimii laajalti käytetty yhdistetty MIT+CMU¹ testiaineisto (Rowley et al. 1998), joka koostuu 130 kuvasta, joissa on yhteensä 507 merkittyä kasvoa.

Seuraavissa kappaleissa tarkastellaan kahta erikoistilannetta, joissa erilaisten kasvojen paikannusjärjestelmien suoriutuminen vaihtelee. Suoritetuissa testeissä käytetty järjestelmä perustuu Intelin tukemaan avoimen lähdekoodin OpenCV-kirjastoon², jossa Violan ja Jonesin menetelmä (2004) on toteutettu. Käytetty kaskadiluokitin on alkuperäisten tekijöiden luoma versio. Tarkasteluikkunoiden skaalauskerroksena s käytettiin lukua 1,1. Havaintojen jälkikäsitteilyä ihonvärimallin avulla tai kuvaussuunnan tunnistusta ei käytetty.

4.2. Piirteiden peittyminen

Sekä paikallisten piirteiden tunnistimien että tietämyslähtöisten kasvojen paikannusjärjestelmien yleinen ongelma on kasvojen peittyminen. Edellä esitellyn järjestelmän käyttämät ensimmäiset piirteet vaikuttaisivat toisaalta myös riippuvan esimerkiksi tummentumista silmien kohdalla. Piirteiden peittyminen on hyvin yleistä esimerkiksi ryhmäkuvissa, joissa takana olevan henkilön kasvot jäävät osittain edessä olevan henkilön taakse. Samankaltaisia ongelmia aiheuttavat myös valaistuksesta johtuvat jyrkät varjot sekä kuvan väärä tarkennus.

Jotta tunnistimen sietokykyä voitiin testata kvalitatiivisesti, luotiin testausta varten eräästä korkealaatuisesta kasvokuvasta erilaisia versioita. Näissä pyrittiin häivyttämään paikallisia piirteitä eri tavoin, joko sumeuttamalla tai maalamalla jokin kuvan suorakulmainen alue mustaksi tai valkoiseksi. Kuvasta peitettiin vuorotellen toinen tai molemmat silmät, suun alue sekä vaihteittain osa otsasta ja leuasta alkaen kuvan reunoista. Lisäksi kokeiltiin poskialueiden tai toisen kasvopuoliskon peittämistä.

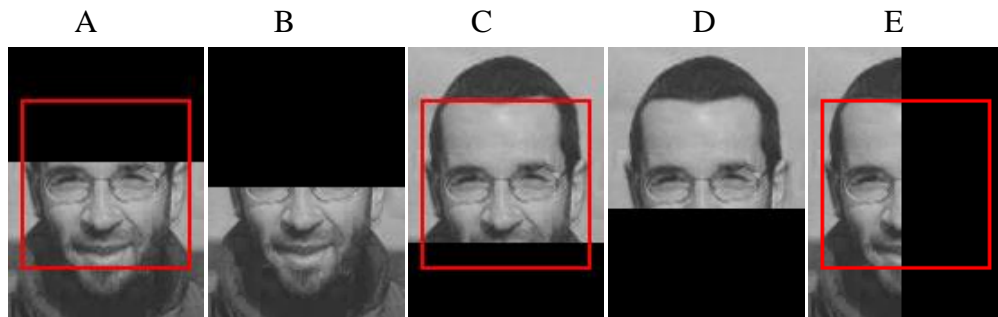
¹ Aineisto saatavilla osoitteessa <http://vasc.ri.cmu.edu/NNFaceDetector/>

² <http://opencvlibrary.sourceforge.net/>



Kuva 7 Silmien alueen peittäminen

Menetelmä sietä yllättävän hyvin piirteiden peittymistä. Peittymisen tyyppi vaikutti tunnistamiseen vain yhdessä tapauksessa: Silmien ympäristön korvaaminen leveyssuuntaisella palkilla esti tunnistamisen, kun palkin väri oli valkoinen (Kuva 7). Pelkkien silmien peittäminen yhdessä tai erikseen ei vaikuttanut tunnistamiseen. Mielenkiintoinen havainto oli, että tunnistettujen kasvojen koko ja sijainti ei juurikaan vaihdellut.



Kuva 8 Kasvojen osien peittymisen vaikutus tunnistamiseen

Kuva 8on esimerkkejä ääritapauksista, joissa tunnistus vielä juuri onnistui tai ei onnistunut. Otsan peittäminen ei haitannut, vaikka raja vedettiin aina kulmakarvojen korkeudelle asti (Kuva 8a). Sen sijaan, kun sekä otsa että silmät peittyivät (Kuva 8b), niin tunnistaminen estyi aina. Suun alueen peittäminen ei myöskään haitannut tunnistamista lainkaan (kuva 8c). Kun raja vedettiin silmien alapuolelle asti (kuva 8d), niin ongelmia ilmeni vain, jos alue korvattiin mustalla värillä. Kasvojen sivuttainen peittäminen ei aiheuttanut ongelmia, vaikka vain puolet kasvoista oli näkyvissä (kuva 8e).

4.3. Tasorotaatio

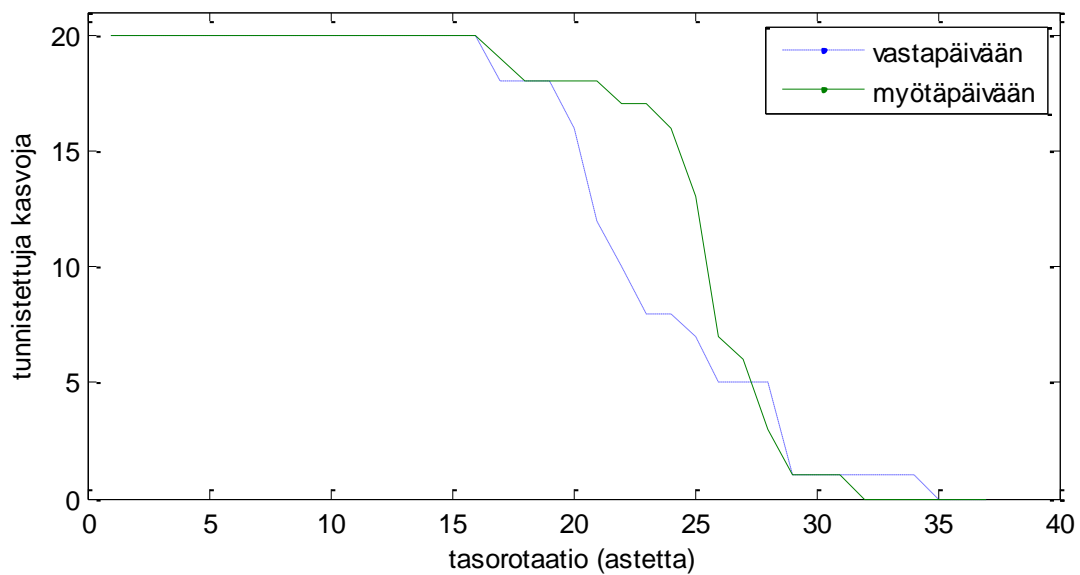
Opetusaineiston valinnalla ja keinotekoisella kasvattamisella voidaan parantaa merkittävästi tunnistimen kykyä sietää kasvojen rotaatiota kuvausakselin suhteen. Tässä suhteessa käytetyn kaskadin opetusaineiston laatu on kuitenkin tuntematon. Rotaation sietokyvyn määrittäminen on tärkeää siksi, että voidaan määrittää tarpeellinen eri kuvaussuuntiin erikoistuneiden tunnistimien määrä.

Ominaisuuden tutkimiseksi muodostettiin kahdestakymmenestä erityyppisestä kasvosta keinotekoisesti kierrettyjä kopioita. Edustettuina oli eri ihonvärejä sekä kumpikin sukupuoli, ja osalla kohteista oli silmälasit tai parta. Aluksi kasvokuvat

kohdistettiin asettamalla silmien keskipisteiden välinen akseli vaakasuoraan. Sen jälkeen luotiin myötä- ja vastapäivään käännettyjä kopioita yhden kallistusasteen välein.

Monissa tapauksissa asteluvun kasvaessa tarpeeksi, alkoi tunnustetun alueen koko vaihdella ja sen keskipiste siirtyi selvästi. Syynä tähän on samasta kohteesta eri skaaloissa tehtyjen havaintojen yhdistäminen. Kahdessa tapauksessa kuvan kohdistus vaihtui radikaalisti keskimääräistä korkeammalla asteluvulla, ja tunnistus onnistui jopa yli 39° kulmassa. Nämä muuttuneella kohdistuksella suoritettut havainnot jätettiin huomioimatta tarkastelussa.

Rotaation sietokyky (Kuva 9) oli keskimäärin vastapäivään $27 \pm 6^\circ$ ja myötäpäivään $24 \pm 4^\circ$. Tunnistus ei häiriintynyt kummassakaan suunnassa käänntökulman ollessa alle 15° .



Kuva 9 Tunnistettujen kasvojen määrän riippuvuus käänntökulmasta

4.4. Johtopäätökset

Edellä tarkasteltiin Violan ja Jonesin (2001) tunnistimen toimintaa kahdessa ongelmallisessa ja usein esiintyvässä käytännön tilanteessa. Tunnistimen toimintaa vaikeutettiin peittämällä kasvojen paikallisia piirteitä ja kääntämällä kuvausakselia keinotekoisesti. Kokeissa havaittiin tunnistimen sietävän näitä häiriötyyppejä hyvin. Tuloksista ei voi vetää suoria johtopäätöksiä menetelmän toiminnasta yleensä, koska käytetty opetusaineisto vaikuttaa merkittävästi tunnistimen vääristymien sietokykyyn. Koneoppimiseen perustuvilla menetelmillä on tosin tunnetusti parempi sietokyky kuin menetelmillä, jotka perustuvat rakenteellisesti paikallisten piirteiden löytämiseen (Yang et al. 2002).

Piirteiden keinotekoinen peittäminen osoitti, että varsinkin silmien ja poskipäiden välisen alueen merkittävä muutos estää havaitsemisen. Käytetystä menetelmästä on tehty parannettu versio, joka pystyy ennustamaan peittymisen lajin kahdeksasta perustapauksesta. Paikantamiseen voidaan käyttää erillistä tunnistinta, joka toimii ilman peittyneellä alueella sijaitsevia piirteitä. (Lin et al. 2003.)

Tasorotaatiokokeissa havaittiin melko suurta vaihtelua eri kasvokuvien suurimmassa tunnistuskulmassa. Kuitenkin pienin kallistuskulma ($\pm 15^\circ$), jossa tunnistus aina onnistuu, vastaa aiempia tutkimustuloksia (Viola & Jones 2004). Tämä kulma lienee hyvä lähtökohta suunniteltaessa eri kallistuskulmiin erikoistuneiden tunnistimien määrää.

LÄHTEET

- Duda, R.O., Hart, P.E. & Stork, D.G. 2001. Pattern classification. 2nd ed. New York, John Wiley & Sons, Inc. 654 p.
- Freund, Y. & Schapire, R.E., 1996. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences* 55, 1, pp. 119-139.
- Huynh-Thu, Q., Meguro, M. & Kaneko, M. 2002. Skin-color extraction in images with complex background and varying illumination. *Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision*, Nara, Japan, December 11-13, 2002. Los Alamitos, CA, USA, IEEE Computer Society. pp. 280-285.
- Jones, M. & Viola, P. 2003. Fast Multi-view Face Detection. Cambridge MA, USA, *Mitsubishi Electrical Research Laboratories, Inc*, Technical Report TR2003-96. 8 p.
- Pham, M. & Cham, T. 2007. Fast training and selection of Haar features using statistics in boosting-based face detection. *Proceedings of IEEE 11th International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 14-20, 2007. IEEE. pp. 1-7.
- Quinlan, R.J. 1993. C4.5: Programs for Machine Learning. San Mateo CA, USA, Morgan Kaufmann Publishers Inc. 302 p.
- Rowley, H., Baluja, S. & Kanade, T. 1998. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20, 1, pp. 23–38.
- Sung, K.K. & Poggio, T. 1998. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20, 1, pp. 39–51.
- Viola, P. & Jones, M. 2004. Robust real-time face detection. *International Journal of Computer Vision*, 57, 2, pp. 137-154.
- Yang, M.H., Kriegman, D. & Ahuja, N. 2002. Detecting Faces in Images: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, 1, pp. 34-58.
- Lin, Y.Y, Liu, T.L. & Fuh, C.S. 2003. Fast object detection with occlusions. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1, 1, pp. 587-594.