Debreceni Egyetem

Informatikai Kar

TESTI GESZTUSOK SZÁMÍTÓGÉPES FELISMERÉSE

Témavezető:
Dr. Fazekas Attila
Egyetemi docens

Készítette:
Makara Csaba
V. PTM

Debrecen

2011

Tartalomjegyzék

1 Bevezetés

Ha körülnézünk közvetlen környezetünkben, nagyon kevés olyan tárgyat és technológiát találunk, amelynek létrejöttében ne játszott volna kisebb-nagyobb szerepet az informatika. A számítógépek a mindennapi munkavégzésünk alapvető eszközei lettek közvetett, vagy közvetlen formában.

Ez a folyamatos „fogyasztói igény” az újítások és jobb megoldások iránt garantálja az informatika és a számítógépek folyamatos és szükséges fejlődését, valamint alátámasztja az e területen folytatott kutatások szükségességét.

Az ember-számítógép kommunikáció (human-computer interaction, HCI) kutatásának fontosságát nagyon jól mutatja a mindennapi élet. A számítógépeket vezérelni kell. Ezt többféle beviteli eszközzel tehetjük meg, a mindennapos egértől és billentyűzettől egészen a specializált eszközökig, mint például a virtuális kesztyűk. A HCI kutatásának egyik célja, hogy az ember és a számítógép közötti kommunikációt gördülékenyebbé, az ember számára természetesebbé tegye.

A különböző eszközök irányítására sokféle megoldás létezik. Csakhogy az eszközök számának növekedésével a különböző irányító eszközök száma is növekszik. Az emberek asztalán manapság sajnos több távirányító van, mint könyv. Mindenképpen szükséges tehát egy természetesebb, eszközfüggetlen módot találnunk az eszközeink irányítására.

Mint mindig, a tudomány most is nyúlhat a természethez, mint ötletforráshoz. A kommunikációval kapcsolatos ötleteket a mindennapi életünkben kell keresnünk. Az emberek közötti kommunikáció állandó eszközei a gesztusok. A gesztusok sokfélék, különböző típusúak és sokféle jelentéssel bírhatnak. Hamarosan eljöhet az idő, amikor az intelligens otthon koncepciója mindennapossá válhat. Az egész otthonunkat, legalábbis annak legtöbb funkcióját számítógép fogja vezérelni. Mi sem lenne természetesebb annál, mint ha a saját házunkat gesztusokkal tudnánk vezérelni?

A dolgozatom célja, hogy egy szűk áttekintést adjak a gesztusfelismerő rendszerekről, működésük lépéseiről, valamint hogy bemutassak néhány lehetséges módszert e lépések megvalósítására. A dolgozat keretei nem engedik meg a téma minden részletre kiterjedő bemutatását, de igyekszem a gesztusfelismerő rendszerek lehető legtöbb vonatkozását bemutatni.

1.1 A dolgozat felépítése

Napjainkban a tudományos kutatások egyre többször ívelnek át a tudományok határain. Az új eredmények jellemzően több résztudományág kutatóinak együttes munkája alapján születnek. Így történik ez a HCI kutatások területén is, ahol pszichológusok, szociológusok, informatikusok és számos további tudományterület képviselőinek eredményeit hasznosítják.

Éppen ezért, a téma nem tisztán informatikai vonatkozásai miatt, a második fejezetben bemutatom azokat, a gesztusokhoz és az emberi kommunikációhoz kapcsolódó ismereteket, amelyeket szükséges megemlíteni a gesztusfelismerő rendszerek tárgyalásakor.

A harmadik fejezetben bemutatom, hogy milyen típusú gesztusfelismerő rendszerek és hozzájuk kapcsolódó perifériák léteznek. Továbbá szót ejtek a gesztusfelismerő rendszerek alkalmazásairól, valamint a multimodális rendszerekről.

A negyedik fejezetben részletesen tárgyalom a gesztusfelismerő rendszerek működésének lépéseit, és röviden bemutatok néhány módszert e lépések megvalósítására. A gesztusok szegmentálásának témáját részletesebben tárgyalom, valamint röviden áttekintem az előfeldolgozás, a gesztuskövetés és az osztályozás témaköreit.

Az ötödik fejezetben röviden bemutatom a Dr. Fazekas Attilával és Kovács Györggyel közösen fejlesztett Kő-papír-olló játékot, mellyel 2009-ben az Amszterdamban rendezett nemzetközi Intelligent Virtual Agents(IVA) konferencián második helyezést értünk el.

2. Gesztusok és a gesztusok szerepe

Ahogyan a bevezetésben is említettem, a gesztusok nagyon fontos szerepet játszanak a HCI témájú kutatásokban, mint az emberi kommunikáció alapvető és szükséges velejárói. A gesztusok egyik definícióját Kurtenbach és Hulteen (1990) adta[1]:

„A gesztus a test egy olyan mozdulata, amely információt hordoz. Integetve elköszönni egy gesztus. A billentyűzet egy gombjának lenyomása nem gesztus, mert az ujj mozgása a billentyűzet felé nem szignifikáns. Az számít, hogy melyik gombot nyomtuk meg."

A gesztusoknak tehát két fontos jellemzője van:

- jelentéssel bíró információt hordoznak
- hatnak a környezetre

2.1 Ember-ember kommunikáció

A kommunikáció megvalósításához szükség van egy adó félre, aki használja a gesztust, és legalább egy fogadó félre, aki értelmezi azt. Az emberek kulturális és nyelvi sokszínűségét figyelembe véve nagyon könnyű belátni, hogy ugyanazon gesztusok teljesen mást jelenthetnek a különböző kultúrákban. Az eltérés annyira szélsőséges lehet, hogy míg egy gesztus valakinek igent jelent, az a másik kultúrában jelenthet nemet is. Ezen felül - nagyon hasonlóan az íráshoz vagy a beszédhez - akár azonos kultúrán belül is előfordul, hogy az egyes gesztusokhoz társított jelentés emberenként különböző. Nem ritka az sem, hogy egy ember ugyanazt a gesztust, két különböző jelentéssel használja különböző helyzetekben. Sok gesztusnak ugyanaz, vagy nagyon hasonló a jelentése. Ezen felül, vannak olyan nyelvek, melyeknek szerves részét képezik a gesztusok, lehetetlen nélkülük „beszélni" az adott nyelvet.

Ha a leírtakhoz hozzávesszük azt a tényt, hogy a mindennapi gesztusok száma közel sem mondható alacsonynak, szembetűnik, hogy egy univerzális gesztusfelismerő rendszer megalkotása, amely felismeri az összes lehetséges gesztust, az összes lehetséges szituációban, igencsak nehézkes lenne.

Ha azt akarjuk felmérni, hogy a gesztusok, mint a kommunikáció fontos elemei, mennyire játszanak fontos szerepet az emberek közötti kommunikációban, elég arra a közkeletű véleményre gondolnunk, mely szerint mennyire személytelenek egyes napjainkban

rendszeresen használt kommunikációs formák, mint például a levél/email, a chat és a telefon. Ez a gondolat abból is fakadhat, hogy mikor ezen a kommunikációs formákat használjuk, nem kerülünk közvetlen kapcsolatba a másik féllel, és így a gesztusait sem észlelhetjük. Bár a közelmúltban születtek újabb megoldások, mint például a videó-telefon vagy a web kamerás beszélgetés, ezek sem hasonlíthatóak össze a személyes kommunikációval.

2.2 Gesztusok csoportosítása

Ahhoz, hogy a gesztusok halmazát leszűkítsük, valamilyen módon csoportosítanunk kell őket. Szerencsére a gesztusok különböző szempontok szerinti csoportosításai léteznek. Ismerve ezeket könnyebb kiválasztanunk, hogy egy adott feladathoz mire is van szükségünk.

A gesztusok lehetnek egyszerűek, ideértve az integetéstől kezdve egészen a jelnyelvig mindent, de valamilyen tárgyat is használhatunk. Például ha valamire rámutatunk vagy valamit mozgatunk. A gesztusokat funkciójuk szerint, három csoportba sorolhatjuk:

- szemiotikus gesztusok: melyekkel valamilyen jelentéssel bíró információt közvetítünk,
- ergotikus gesztusok: melyekkel a környezetet manipuláljuk,
- episztemikus gesztusok: melyek célja, hogy a környezetről információt gyűjtsünk.

Egy másik, a gesztusfelismerő rendszerek számára fontosabb csoportosítást Rime és Schiaratura[2] adta. Ez a csoportosítás a tárgyakat nem használó, („empty hand”), szemiotikus gesztusokat rendszerezi. Ezek lehetnek:

- Szimbolikus gesztusok: melyek szinte minden kultúrában ugyanazt jelentik. Ide tartoznak az Amerikai Jelnyelv gesztusai is,
- Deiktikus gesztusok: e gesztusokkal tudjuk felhívni valakinek a figyelmét egy tárgyra, vagy valamilyen eseményre. Tipikus példa erre, hogy „Tedd azt oda!”,
- Ikonikus gesztusok: e gesztusokkal tudjuk kifejezni a tárgyak milyenségét, alakját, méretét vagy mozgásának irányát. Például valaki a következőt mondja: „Ekkora halat fogtam”, és eközben a két tenyerét egy bizonyos távolságra helyezve fejezi ki a hal méretét,

- Pantomimikus gesztusok: ezekkel a gesztusokkal az ember „kezében lévő, láthatatlan tárgy” mozgása írható le. Például valaki azt mondja: „a labdát így kell eldobni”, és közben utánozza a mozgást.

Ezt az osztályozást McNeill(1992)[3] kiegészítette olyan gesztustípusokkal, melyek a kommunikációs folyamattal függenek össze:

- Ritmikus gesztusok: melyek során a kéz a beszéd ritmusára felfelé- és lefelé irányuló mozgást végez,
- Egyesítő gesztusok: melyek ikonikus, pantomimikus és deiktikus gesztusok variációi. Könnyebben követhetővé teszik a társalgást, valamint a téma látszólag nem összefüggő, de végeredményben összetartozó részeinek összefogását teszik lehetővé.

A kommunikáció részeként, a gesztusok nagyon szorosan kapcsolódnak a beszédhez. Csak a szimbolikus gesztusok interpretálhatóak az üzenet további elemeivel való összefüggések nélkül. Ezek, az összefüggéseket elősegítő elemek lehetnek további gesztusok, vagy beszéd, melyek ekkor a kommunikáció különböző tartalmi részeit fűzik össze. Tehát a gesztusokat osztályozhatjuk a beszéddel való viszonyuk szerint is:

- gesztusok, melyek helyettesítik a beszédet: szimbolikus, deiktikus
- gesztusok, melyek kiegészítik a beszédet: ikonikus, pantomimikus
- gesztusok, melyek a társalgás folyamatához kapcsolódnak: ritmikus, egyesítő

Látható, hogy a gesztusok és a beszéd szoros kapcsolatban állnak egymással. Ennek ellenére a gesztusok egy része teljesen egyedi, nyelvszerű jellé vált. A jelnyelv elemei annyira egyediek, hogy a megértésükhöz nincs szükség azt kísérő beszédre. Megjegyzendő, hogy az ikonikus gesztusok nem érthetőek meg beszéd nélkül, mivel ezek éppen a beszédet egészítik ki.

A gesztusok lehetnek statikusak, vagy dinamikusak. A statikus gesztusoknak nincsen mozgási komponensük, nincsen kezdő- és végpozíciójuk. Ilyen például, a nagyon sok kultúrában használt „Oké” jel: ökölbe szorítjuk a kezünket, és a hüvelykujjunkat felfelé tartjuk. A dinamikus gesztusoknak több fázisa (helyzete) van, valamint térbeli mozgás

jellemző rá. Néhány gesztusnak statikus és dinamikus jellemzői is vannak, például a jelnyelv egyes elemeinek.

2.3 Gesztusok jelentése

Ahogyan a fejezet elején is említettem, a gesztusok egyik jellemzője, hogy valamilyen jelentéssel bíró információt hordoznak. Ezt az információt általában az alábbiak segítségével lehet meghatározni:

- térbeli elhelyezkedés,
- bejárt út,
- maga a jel, amit mutatunk,
- érzelmi állapot.

E paraméterek meghatározásához, meg kell határozni a test helyzetét, alakját és mozgását. A megvalósítás részleteit a dolgozat negyedik fejezetében fejtem ki.

2.4 Kézi, arci és testi gesztusok

A *2.2-ben* bemutatott csoportosításokon kívül, rendszerezhetjük a gesztusokat aszerint, hogy mely testrész(ek) vesz(nek) részt a létrehozásukban. Ennek alapján beszélhetünk:

- kézi és kar gesztusokról, melyek jelentősége többek között a jelnyelv felismerőknél vagy a különböző szórakoztató programoknál (például virtuális környezet) van,
- fej és arci gesztusokról, például a fej mozgatása, a tekintet iránya, a száj mozgatása, kacsintás és az érzelmek kinyilvánítása,
- testi gesztusokról, melyek esetén az egész test részt vesz a mozgásban. Például analizálva egy táncos mozgását, generálhatunk a ritmusnak megfelelő zenét.

2.4.1 Kézi gesztusok

A gesztusok tehát a test különböző részeinek önmagukban, vagy több testrésszel összehangoltan végzett mozgása. Az ember testrészei közül a leginkább kifejező gesztusokat kézzel lehet létrehozni. A kézi gesztusok esetén két tulajdonságról beszélhetünk:

- kéz tartása: a kéz ujjainak statikus helyzete, mozgási komponens nélkül,
- gesztus: a kéz dinamikus mozgása, miközben az ujjak mozognak vagy statikusak maradnak.

A kézi gesztusok a következő kategóriákba sorolhatóak:

- gesztikuláció: a kéz és karok beszédet kísérő, spontán mozgása. Az emberi gesztusok 90%-a ebbe a csoportba tartozik,
- nyelvszerű gesztusok: beszéd közben, egy bizonyos szó helyettesítésére alkalmas gesztusok,
- pantomimek: egy cselekvés vagy tárgy leírására használható gesztusok, melyeket kísérhet beszéd,
- jelek: sok ember által használt „egyezményes” jelek, például az „Oké” jel,
- jelnyelvek: meghatározott jelentésű gesztusokat tartalmazó, jól definiált nyelvi rendszerek.

A felsorolás sorrendjében, az egyes csoportok egyre kevésbé függenek a beszédtől és a véletlenszerűségük is csökken, valamint nő az adott csoportok nyelvszerűsége. Az utolsó csoport elemeit már nevezhetjük önálló nyelveknek.

3 Gesztusfelismerő rendszerek

Mint mindent, egy gesztusfelismerő rendszert is valamilyen céllal készítünk el, valamilyen feladatot kell ellátnia. A létező gesztusok közül ki kell szűrnünk a számunkra hasznos gesztusokat, és ezekre kell felkészítenünk a rendszert.

Ebben a fejezetben szeretném megmutatni, hogy a gesztusfelismerő rendszereket milyen sokféle területen lehet használni a gyakorlatban, valamint hogy milyen típusaik vannak.

A gesztusfelismerő rendszerek felhasználhatóságának széles körét szeretném szemléltetni az alábbi példákkal, a teljesség igénye nélkül:

- jelnyelv-felismerés
- számítógépek és további eszközök (táv)vezérlése
- stressz és érzelmi állapot nyomon követése
- járműkezelési támogatás
- affective computing[1]
- törvényszéki azonosítás
- hazugságvizsgálat
- virtuális környezet vezérlése
- immerzív[2] játék technológiák

[1] affective computing: érzelmeket felismerő és feldolgozó, valamint érzelmek szimulálására képes rendszerekkel foglalkozó tudományterület.
[2] immerzív: az immerzív virtuális valóság közvetlen, egyes szám első személyű megtapasztalást biztosít.

3.1 A gesztusfelismerő rendszerek építésének megközelítései

Egy gesztusfelismerő rendszer építésekor az első kérdés, amivel szembetaláljuk magunkat az, hogy az adott feladathoz illeszkedően, a rendszer bemenetét milyen eszközzel biztosítsuk. Napjainkban a következő kétfajta megközelítés használatos.

3.1.1 Eszköz alapú rendszerek

Annak érdekében, hogy a gesztusokat felismerjük vagy kövessük, használhatunk különféle, érzékelőkkel ellátott eszközöket. Ezek lehetnek kesztyűk, vagy akár teljes testre kiterjedő ruhák is.

Ilyen eszközök esetében fontos szempontok a pontosság, a késleltetés, a felbontás, az ár, valamint az eszközt használó személy kényelme is. A kényelemi szempont különösen fontos mivel, ahogy azt a bevezetésben is írtam, a HCI kutatások célja, hogy természetes módszereket dolgozzanak ki a számítógéppel való kommunikációra. Az például már nehezen nevezhető természetes kommunikációnak, ha a viselendő kesztyűből több kábel is fut a számítógép felé.

Ezen eszközök előnye főleg a pontosságukban rejlik. Egyes virtuális kesztyűk képesek az ujjak 5 fokos hajlítását is érzékelni.

3.1.1.1 Gyorsulásmérésen alapuló rendszerek

Az egyik leghíresebb gyorsulásmérésen alapuló rendszert a Nintendo alkotta meg, ez a Wiimote[4]. Az rendszer kontrollere a *3.1 ábrán* látható. Ez az eszköz 3 tengely mentén méri a gyorsulást. Egy optikai érzékelő állapítja meg, hogy merre mutatunk az eszközzel. Egy giroszkópot tartalmazó kiegészítő segítségével a forgásokat is érzékeli. A játékkonzollal való kapcsolódásra Bluetooth technológiát használnak.

Egy másik eszköz az úgynevezett SoapBox (Sensing Operating and Activating Pheripheral Box) [5]. Tartalmaz egy 3 tengely mentén mérő gyorsulásmérőt, egy fényerősség érzékelőt, egy elektromos iránytűt és egy optikai távolság érzékelőt.

3.1 ábra. A Nintendo Wiimote kontrollerje.

Ezek a rendszerek a gesztusokat adatvektorként reprezentálják, mely a kontroller aktuális, 3 tengely szerinti gyorsulását reprezentálja. Az adatvektorok képezik a rendszer tanítási- és felismerési fázisainak alapját.

A felismerés folyamata három fázisból áll. Először klaszterizálják az adatokat egy „k-mean" algoritmus segítségével, majd HHM-t (Hidden Markov Model) használnak a rendszer tanítására és a gesztusok felismerésére, végül Bayes-osztályozással kiválasztják a megfelelő gesztust.

3.1.1.2 Kesztyű alapú rendszerek

A virtuális kesztyűk számos technológiát használhatnak, melyekkel mérhető az ujjak hajlítottsága, emellett tartalmazhatnak egy mozgásérzékelőt, mely az eszköz globális helyzetét detektálja. Egy ilyen eszköz látható a *3.2 ábrán*.

3.2 ábra. Cyberglove II

A [6]-ban leírt módszer szerint, a kéz helyzetét az ujjak hajlítottsága és a kéz állásának iránya alapján detektálják. A rendszerek, a gyorsulásmérésen alapuló rendszerekhez hasonlóan, tanítási-felismerési technikákat használnak. A felismerő rendszernek két komponense van:

- adatszerzés: a rendszernek ez a része dolgozza fel a beérkező adatot, majd továbbítja a gesztusmenedzser felé. A feldolgozás során optimalizálja az adatot, például zajszűrést hajt végre a beérkező jelen. Ahhoz hogy a rendszer felismerje a gesztust, a felhasználónak 300-600 milliszekundumig egyhelyben kell tartania a kezét.
- gesztusmenedzser: ez a komponens tartalmazza azokat a gesztusokat, melyeket a rendszer képes felismerni. A rendszer a beérkező adatok alapján próbálja meg felismerni a gesztust. Először az ujjak egymáshoz viszonyított elhelyezkedése alapján szűkíti a lehetséges megoldások körét, majd a kéz helyzetét és irányát hasonlítja össze a tárolt gesztusokkal.

3.3 ábra. A motion capture eljáráshoz használt érzékelőkkel ellátott ruha

Bár nem kesztyűk, de ide sorolhatjuk az olyan teljes testre kiterjedő, érzékelőkkel ellátott ruhákat *(3.3 ábra)*, melyeket a kesztyűkhöz hasonlóan használhatunk gesztus vagy mozgás detektálására. Ezek fő alkalmazási területe az úgynevezett motion capture eljárás[3]. Bár a természetesség kritériumát nem teljesíti, alkalmas lehet speciális gesztusfelismerő rendszerekhez.

[3] motion capture: vagyis mozgásrögzítés. Lényege, hogy rögzítik az alany mozgását, majd azt egy digitális modellre ültetik át. Fő alkalmazási területei a szórakoztatóipar, az orvostudomány és a hadiipar.

3.1.2 Gépi látás alapú rendszerek

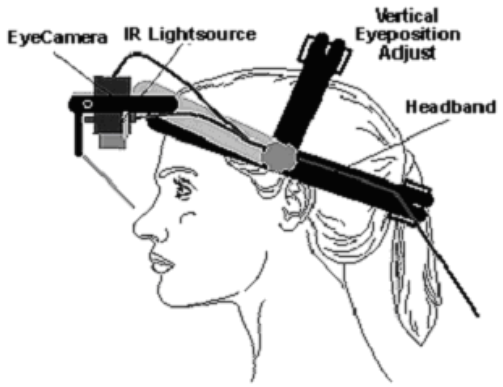
Bár a gépi látás alapú rendszerek is eszközöket használnak, mely ebben az esetben egy vagy több kamerát jelent, mégis külön kategóriába sorolhatjuk őket. Gépi látás alapú rendszerek esetén a bemenet egy kép, vagy videó folyam, amelyről a rendszer felismeri az adott személy gesztusát vagy testtartását.

A rendszer működhet egy vagy több kamerával is. Utóbbi esetben a cél a kép háromdimenziós modelljének megalkotása.

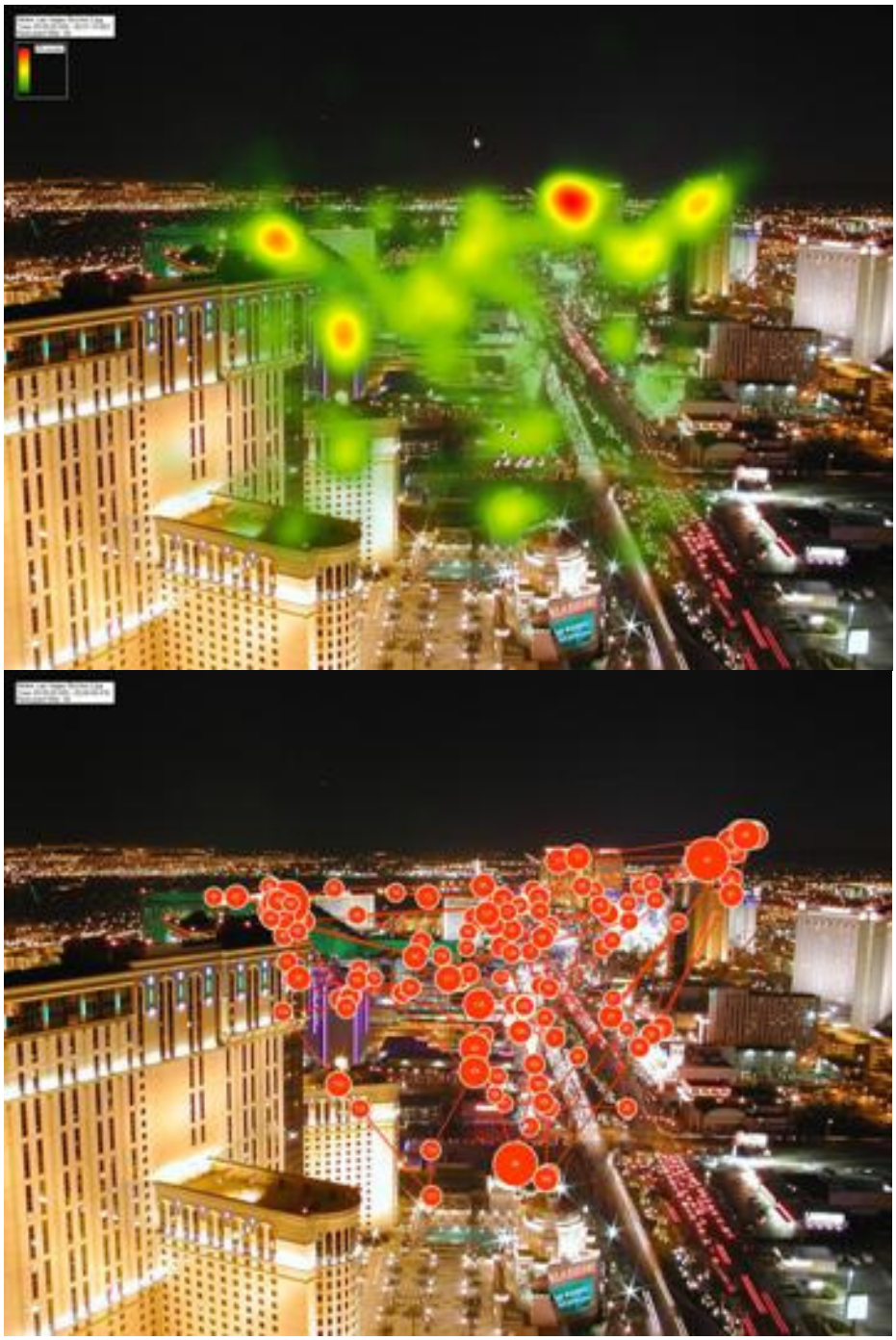
A gépi látás alapú rendszerek előnye - az eszköz alapúakkal szemben - a jobb felhasználhatóság. Használatához nincs szükség semmilyen eszközre (a kamerán kívül). Ezen felül figyelembe tudják venni a színeket és a textúrát, melyeket az eszköz alapú rendszerek nem.

A gépi látás alapú rendszereknek van két nagyobb hátránya. Egyrészről az eszköz alapú rendszerekhez képest nagyon számításigényesek lehetnek, másfelől a rendszer nagyon érzékeny a különböző hátterekre és a fényviszonyokra. Ezek a rendszerek sokfélék lehetnek attól függően, hogy hány kamerát használunk, milyen e kamerák sebessége és késleltetése, két- vagy háromdimenziós-e a rendszer, valamint hogy a működés közben az időtényezőt is figyelembe vesszük-e.

Egy példa az ilyen rendszerekre a szemkövetés, melynek sok felhasználási területe van. Kísérleteket végeznek egy fejre szerelt kamerával *(3.4 ábra)*, amely követi a szem mozgását, ezáltal érzékelve, hogy az alany például a neki megmutatott kép melyik részét, és mennyi ideig nézi. Ezen kísérletek eredményei *(3.5 ábra)* sok helyen felhasználhatóak, például a web design területén.



3.4 ábra. Egy szemkövetésre alkalmas készülék vázlata.



3.5 ábra. Egy elvégzett szemkövetési kísérlet eredménye.

3.2 A gesztusfelismerő rendszerek típusai

A gesztusfelismerő rendszereket csoportosíthatjuk aszerint, hogy mennyire komplexek. A legegyszerűbb szinten azok a rendszerek állnak, melyek működtetése kizárólag gesztusokkal történik. Ha a működtetésében már nem csak gesztusok, hanem más érzékszervek is részt vesznek, akkor multimodális rendszerekről beszélünk. A legmagasabb szinten a társalgási rendszerek állnak, melyek képesek a mindennapi életünkben megszokott kommunikációhoz hasonló körülményeket teremteni.

3.2.1 Csak gesztust felismerő rendszerek

Az ebbe a kategóriába tartozó rendszerek a legegyszerűbbek. Ezen rendszerek, attól függően, hogy mennyi és milyen gesztusokat képesek detektálni, használhatóak akár bizonyos inputeszközök (távirányítók vagy egér) kiváltására, de akár jelnyelv felismerésre is. Felismerhetnek statikus és dinamikus gesztusokat vagy ezek kombinációját is.

A sok gesztus felismerésére képes rendszereknél a kéz pontos követésére van szükség. Ez megvalósítható a számítógépes látás eszközeivel, de olyan eszközökkel is, mint a virtuális kesztyű. Eszköz alapú rendszer esetén a detektálni kívánt gesztust leíró adatok birtokában, a gesztusfelismeréséhez több módszert is alkalmazhatunk, mint például statisztikai módszereket vagy neurális hálókat.

Gépi látás esetén két megközelítést alkalmazhatunk:

- modell alapú: a gesztusfelismeréshez ebben az esetben a felhasználó kezéről egy háromdimenziós modell készül,
- kép alapú: a gesztusfelismeréshez a kézről készült képekből számolhatjuk ki a szükséges sajátságokat.

Ezen rendszerek egy fajtája, amelyek a természetes gesztusokat (például a kéz dinamikus mozgását) detektálják. Egy korai példa erre az 1920-as években készült elektronikus hangszer, a theramin *(3.6 ábra).*

3.6 ábra. A theramin két kézzel működtethető. A kezek pozícióját figyeli a vertikális és horizontális tengely mentén. Az egyik kézzel a hangerő, a másikkal a hangmagasság vezérelhető.

A szimbolikus gesztusokkal dolgozó rendszereket gyakran használják immerzív virtuális környezetekben. Ezek a rendszerek előre betanított gesztusokkal dolgoznak, melyekkel közlekedhetünk a virtuális környezetben és manipulálhatjuk azt.

Ezen rendszerek hátránya, hogy a felhasználónak előre be kell tanulnia a rendszer által kezelt gesztusokat, ami a használt gesztusok számának növekedésével egyre nehezebb. E mellett a gesztus szegmentálása is nehézkes, mert a felhasználó nem parancsnak szánt gesztusait el kell különíteni a tényleges parancsoktól. Például, ha a virtuális környezetben való haladás parancsát a mutató ujjal való mutatás jelenti, akkor ezt az egyszerű kéztartást véletlenül sem használhatjuk másra, mert rögtön elindulnánk abba az irányba, ahová mutatunk.

3.2.2 Multimodális rendszerek

A multimodális rendszerek jellemzően egynél több forrásból származó bemenettel dolgoznak. A multimodális rendszerek esetében a gesztusok mellett a hang jelenik meg, mint másik modalitás. A kézi és hang általi utasítások kombinálásának számos előnye van. Ezek közül is a legfontosabb, hogy a felhasználó számára - a csak gesztust felismerő rendszerekhez képest - növelik a kommunikáció természetességét.

A beszéd nagyon jól kiegészíti a gesztusokat. A rendszer irányítására használhatóak önmagukban és kombinálva is, ezzel maximalizálva a hatékonyságot. A természetesség

növelése mellett a beszéd és a gesztusok kombinálása növeli a pontosságot és gyakran gyorsítja a detektálást is.

Kognitív pszichológiai kutatások szerint, ha az ember több feladatot végez egyszerre, melyek különböző érzékszerveket használnak, akkor a feladatokat az agy modularizálja és párhuzamosan „futtatja”. Ezt az úgynevezett többszörös erőforrás elméletet a kísérletek is alátámasztották.

A legfontosabb feladat egy multimodális rendszer megalkotásakor, az inputok integrálása. Ahhoz, hogy a felhasználó, a hangja és gesztusai által közvetített parancsát a rendszer értelmezni tudja, a két bemenetet egyesíteni kell egyetlen szemantikus reprezentációvá. Ennek egy egyszerű megközelítése az időbélyegek használata. Ha a felhasználó például rámutat egy tárgyra, és ezzel egy időben azt mondja: „Tedd az asztalt oda!”, akkor a rendszer ahelyett, hogy két külön bementként értelmezné, az időbélyegek egymáshoz közelisége alapján, egy parancsként értelmezheti.

3.2.3 Társalgási rendszerek

Az emberi kommunikációban fontos szerepet játszanak azok a gesztusok, amelyek a társalgás folyamatára fejtik ki hatásukat. Ezek a ritmikus és az egyesítő gesztusok. Például, amikor a szánk elé tartjuk a mutató ujjunkat jelezve ezzel a másik félnek, hogy hagyja abba a beszédet, vagy értetlenül nézünk a beszélgető partnerre jelezve, hogy elvesztettük a beszélgetés fonalát.

A társalgási rendszerek alapötlete, hogy a számítógéppel úgy tudjunk kommunikálni mindenféle hétköznapi eszközt (hang, tekintet, gesztusok, testbeszéd) használva, mint egy másik emberrel.

A társalgás alapvetően a beszédből és az arra érkezett válaszokból áll. A társalgási rendszerek definíció szerint képesek arra, hogy „kapcsolatot” építsenek ki az emberrel. Önmagában véve a gesztusfelismerők és a hangfelismerők nem társalgási rendszerek, mert nem értik meg az embert a szó szoros értelmében véve. A társalgási rendszerek magja egy intelligens ágens, ami akár kontrollálni is tudja a társalgást, multimodális kimenetek segítségével.

A legfontosabb különbség a társalgási- és a multimodális rendszerek között a bemenetek megértésének módjában van. A multimodális rendszerek nem foglalkoznak azzal, hogy a felhasználó üvölt vagy suttog, nem veszi észre, ha szarkasztikusan beszél, és nem von le ezekből következtetéseket. Emellett nem érzékelik az ember tényleges jelenlétét, a testbeszédet és a szemkontaktust abban a formában, mint az emberi kommunikáció esetén megszokott.

Bár egy ilyen rendszer fejlesztése meglehetősen nehéz feladat, alkalmazása több előnnyel is jár. A felhasználónak nincs szüksége semmilyen új tudásra a rendszer használatához, valamint ez a rendszer mutatja a lehető legtermészetesebb, már-már emberszerű viselkedést.

Egy társalgási rendszernek verbális és nem verbális bemeneteket kell felismernie, valamint ezekre szintén verbális és nem verbális választ kell adnia. A nem verbális kimenethez természetesen valamilyen grafikai megoldást kell alkalmazni. Lennie kell egy emberszerű alaknak a képernyőn, akihez beszélhetünk.

Egy nagyon érdekes próbálkozás ebbe az irányba a Lionhead Studios által, a 2009-es Electronic Entertainment Expo-ra (E3) készített Milo nevű projekt *(3.7 ábra).* Ez a projekt a Microsoft által fejlesztett Kinect[4] technológia bemutatására készített tech demo volt.

3.7 ábra. A Milo project

[4] Kinect: kontrollermentes vezérlő technológia a Microsoft Xbox 360 platformjához.

Peter Molyneux, a Lionhead Studios akkori kreatív igazgatója elismerte, hogy a rendszer, mely eredetileg játékprogramnak készült, nem tényleges társalgási rendszer. Így fogalmazott: „… vannak trükkök a dologban, de ezek a trükkök működnek.”.

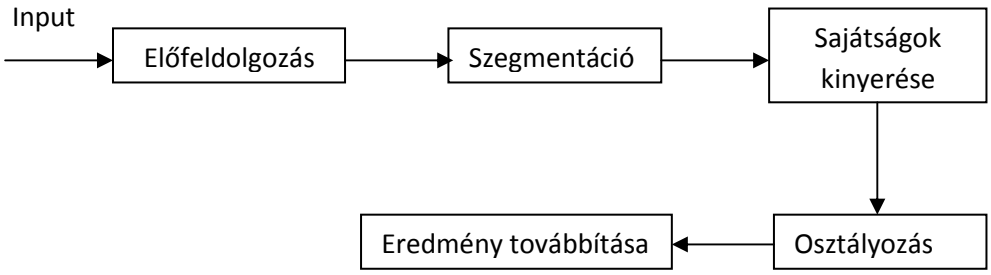
A játék egy beépített „szótáron” alapul, mely a beszélgetés kulcsszavait hasonlítja össze a rendszer beépített mintáival, és így képes emberi kapcsolatteremtésre. Képes beszélni, gesztikulálni és reagálni a felhasználó viselkedésére, ezzel életszerű társalgást szimulálva.

A rendszer gépi látás alapú, egy web kamerán keresztül látja az embert, és mikrofon segítségével érzékeli a hangját.

4 A gesztusfelismerő rendszerek megvalósítása

Ebben a fejezetben bemutatom egy gesztusfelismerő rendszer működésének folyamatát, amire a továbbiakban egyszerűen rendszerként hivatkozom. Milyen lépésekből áll egy ilyen rendszer működése, és milyen technikákat alkalmazhatunk az implementáció során a lépések megvalósításához? A továbbiakban gépi látás alapú rendszerekkel foglalkozom, de a bemutatott módszerek egy része eszköz alapú rendszerek esetén is használható.

Egy tipikus gesztusfelismerő rendszer a működése során a következő ábrán látható lépéseket hajtja végre:



4.1 ábra. Egy gesztusfelismerő rendszer által végrehajtott lépések.

A lépések (a megvalósításkor használt különböző technikáktól függően) nem biztos, hogy ennyire elkülönülnek egymástól. Egyes módszerek egyszerre több lépést is megvalósítanak.

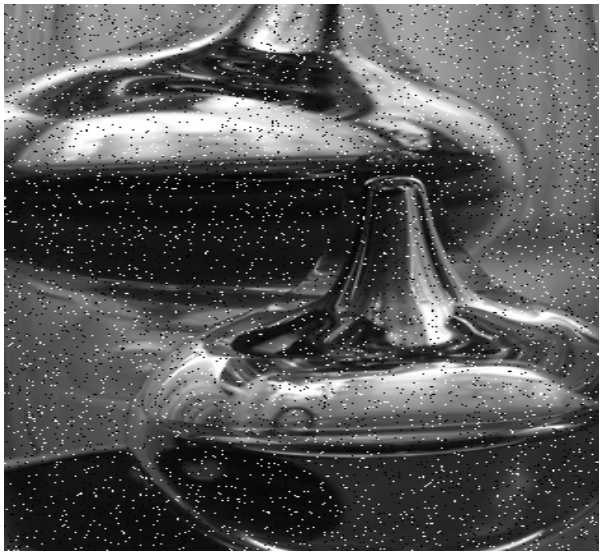
A program, mely a rendszert tartalmazza, egy videó eszközről (általában webkameráról) kapja a bemenetet videó folyamként, mely képek sorozata. A rendszer fontos paramétere a sebessége, azaz hogy hány képet tud feldolgozni másodpercenként. A rendszer sebessége az azt felépítő módszerek számításigényétől függ, így érdemes a rendszert olyan módszerek segítségével felépíteni, amelyekkel a rendszer céljához mérten megfelelő sebességet érhetjük el.

A fejezet további részében, az egyes lépések megvalósítására mutatok be néhány megközelítést és módszert, a szegmentáció témáját pedig részletesebben tárgyalom.

4.1 Előfeldolgozás

Az előfeldolgozás során a bementi adatfolyamról érkező képeken a további feldolgozás megkönnyítéséhez, és a gesztus pontosabb detektálásához szükséges lépéseket végezhetjük el. A bemenetről, mint képről, elmondható hogy lehetnek hibái. Ezeket okozhatja valamilyen interferencia, mely a képrögzítés során keletkezett. Ekkor mondjuk, hogy a kép zajos. Másfelől a rögzítés közbeni külső fényviszonyok nem biztos, hogy megfelelőek a rendszer működése szempontjából. Alapvetően rosszak vagy folyamatosan változóak is lehetnek.

Ha a bemenetként kapott kép zajjal terhelt, akkor ez nagyon erős intenzitásváltozások formájában jelenik meg a képen.



4.2 ábra. A képzajok egyik formája, az úgynevezett salt and pepper zaj, ami a képen véletlenszerűen elhelyezkedő fekete és fehér képpontokként jelenik meg.

A zajt képjavítási eljárásokkal csökkenthetjük vagy szüntethetjük meg, melyek kiegyenlítik az intenzitásegyenetlenségeket. Az egyik leggyakrabban használt képjavítási eljárás a szűrők használata. A szűrőket mind a kép- mind a frekvenciatartományban használhatjuk. A képtartományban való zajszűrésre alapvető technikák a környezeti átlagolás, a mediánszűrő és az intervallum módszer. Ezekkel az egyszerű módszerekkel az a probléma, hogy homályosítják a képet. A homályosítás csökkentésére használhatunk küszöbölést: a képpont csak akkor változzon meg, ha az eredeti és az új világosságérték

különbsége elér egy bizonyos küszöböt. A képtartományban végzett szűrések előnye, hogy hatékonyak és viszonylag gyorsak.

A frekvenciatartományban végzett képjavítás Fourier-transzformáción alapul. A frekvenciatartományban a zaj magasabb frekvenciájú, azaz rövidebb hullámhosszú bázisfüggvényekként jelenik meg. Ha ezeket kiszűrjük, azaz nem vesszük figyelembe a kép visszaállításakor, akkor a zaj hatása csökken. Mivel az élek is hirtelen átmenetet jelentenek a képen, ezért a magas frekvenciájú bázisfüggvények elhagyásával az élek sem lesznek tökéletesen visszaállíthatók, a visszaállított kép homályosabb lesz, mint az eredeti. A probléma megoldására használható szűrő az alul-áteresztő szűrő, de használhatunk Butterworth szűrőt is, ami kevésbé homályos képet eredményez.

A zajokon kívül, a kép másik hibája lehet, hogy egyenlőtlenül megvilágított vagy kevéssé kontrasztos. Erre megoldás a kontraszt növelése, melyet a világosságkódok eloszlásának módosításával érhetünk el. A probléma kezelésére egy egyszerű megoldás, az úgynevezett hisztogram kiegyenlítés. Ezzel az eljárással el tudjuk érni, hogy a hisztogram csúcsait széthúzva, a kép kontrasztossága megnőjön.

Továbbá a kép lehet homályos. Ekkor a képen az élek nem rajzolódnak ki eléggé. A megoldást itt az élkiemelés jelenti, melynek lényege az egyes képrészletek közötti átmeneti tartomány szűkítése és az elmosódások korrigálása. A módszer hátránya, hogy a zajokat is kiemeli. Az eljárást rendszerint konvolúciós szűrővel, például a Laplace operátorral valósítják meg. Az eljárás azt jelenti, hogy minden képpont világosságkódját helyettesítjük a szűrő által kijelölt környezetében lévő képpontok világosságkódjának valamilyen súlyozott átlagával.

4.2 Szegmentáció

A rendszer működésének következő lépése a szegmentáció, melynek során megtaláljuk a kép számunkra fontos részleteit, amivel a rendszer a továbbiakban dolgozni fog. Például kézi gesztusfelismerés esetén magát a kezet kell megtalálnunk. A továbbiakban bemutatok néhány megközelítést és módszert, melyekkel megvalósíthatjuk a szegmentációt.

4.2.1 Szín alapú szegmentálás

A gépi látás alapú gesztusfelismerő rendszerek képekkel dolgoznak, ezen belül is színinformációkkal. Ezen színinformációk alapján tudjuk detektálni a képen lévő objektumot, amit keresünk. A gesztusok detektálása során, a képen bőrszínű területeket keresünk. A bőrszínű területek szegmentációját nehezíti, hogy a képen lévő bőr színét sok tényező befolyásolhatja:

- megvilágítás: az egymás után beérkező bemeneti képeken lévő objektumok színei nagyon hírtelen megváltozhatnak, akár egy lámpa le- vagy felkapcsolása, akár a megvilágítást befolyásoló egyéb körülmények, például árnyékolás miatt.
- a kamera jellemzői: ha ugyanarról az emberről, azonos megvilágítással, de két különböző kamerával készítünk képet, a kamera jellemzői miatt is eltérőek lehetnek a színek.
- etnikum: azon felül, hogy az azonos etnikumba tartozó emberek bőrszíne is eltérhet egymástól, a különböző etnikumok bőrszíne nagyon széles skálán mozog a fehértől, a sárgán át a feketéig.
- egyéni jellemzők: a bőr színét befolyásolja az ember kora és neme.
- egyéb tényezők: a küllemet befolyásoló tényezők (smink, szemüveg), a kép hátterének színe, az ember mozgása és árnyékhatások, mind-mind befolyásolják a képen lévő bőr színét.

Néhány módszer[7] figyelmen kívül tudja hagyni e tényezők egy részét. A megoldás lényege, hogy a nem látható színtartományokban dolgoznak. Például infravörös képek esetén, a bőr színe invariáns a megvilágításból, etnikumból vagy árnyékolásból adódó változásokra. Az ilyen módszerek hátránya, hogy a rendszer működéséhez szükséges felszerelés nagyon költséges, valamint a felhasználási területek száma nem túl nagy.

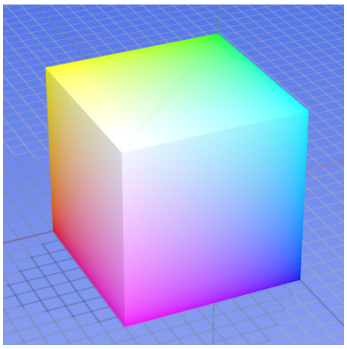
4.2.1.1 Színterek

A bőr szegmentálásának első lépése a használni kívánt színtér kiválasztása. Színtérnek nevezzük az olyan modelleket, amelyek meghatározott színeket reprezentálnak intenzitásértékek segítségével. A színterek általában 1-4 dimenziósak, ahol egy dimenzió egy intenzitás értéket jelent. A legalapvetőbb színtér az RGB. Szinte minden további színteret valamilyen transzformációval az RGB-ből származtathatunk.

Az RGB-ből származtatott színterek előnye, hogy a képeken jobban szétválaszthatóvá válnak a bőr és nem bőr tartományok, valamint a komponensek sokkal kevésbé függenek a változó fényviszonyoktól.

RGB színtér

Az RGB színtér *(4.3 ábra)* áll legközelebb az emberi szem érzékeléséhez. Az alapelv, hogy a vörös(red, R), zöld(green, G) és kék(blue, B) színek keverésével állítjuk elő a színeket.

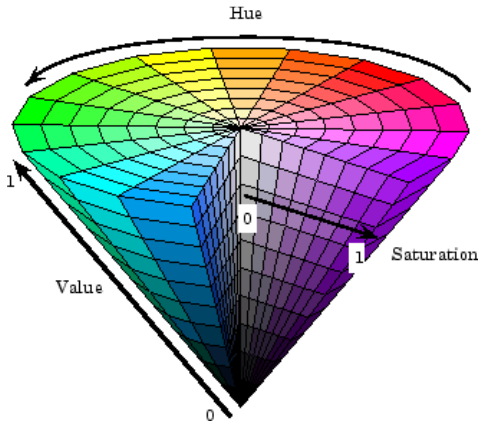


4.3 ábra. Az RGB színteret háromdimenziós kockával szokás szemléltetni.

Az úgynevezett normalizált RGB színtér esetén, a három normalizált komponens összege egy. A normalizált RGB előnyös tulajdonsága, hogy néhány feltétel teljesülése mellett, invariáns az objektumokat érő fényviszonyok változására. A normalizált RGB színteret gyakran használják bőr detektálására [8,9].

HSV színtér

A képfeldolgozásban az RGB színtérnél hasznosabb a HSV színtér *(4.4 ábra)*. Összetevői a színárnyalat (Hue), a telítettség (Saturation) és a szín értéke (Value). Ez utóbbit szokás világosságnak (Lightness), intenzitásnak (Intensity) vagy ragyogásnak (Brightness) is hívni.



4.4 ábra. Ez a színtér egy háromdimenziós kúppal reprezentálható, ahol a szín árnyalata a kúp körkörös metszete, a telítettsége a középponttól mért távolsága és az értéke a kúp csúcsától mért távolsága.

Az RGB-ből HSV-be transzformálás invariáns a megvilágítottság problémáira, valamint az objektumnak a fényforrástól függően vett irányára, ezért nagyon jó választás bőrdetektáláshoz [10].

YCbCr színtér

Az YCbCr színtér szintén népszerű bőrszín detektálás esetén. A színt 3 komponenssel reprezentálja: a fényességgel (luminance, Y), melyet az RGB komponensek súlyozott összegéből számolhatunk, valamint két színinformációból (Cr, Br), melyeket úgy kapunk, hogy kivonjuk a fényesség értékét a B és R komponensekből.

4.2.1.2 Bőrszín osztályozása

A bőr detektálásának célja, hogy a kép képpontjairól el tudjuk dönteni, hogy bőr, vagy nem bőr képpontok. Ennek eldöntésére nagyon sok módszert fejlesztettek ki. A továbbiakban bemutatok ezek közül néhány gyakrabban használt technikát.

Explicit megadás

Ennek a módszernek a lényege, hogy explicit módon megadjuk a használt színtér azon tartományait, amelyekbe a bőr képpontok eshetnek. Ezt különböző szabályok megadásával tehetjük meg. Például[11]:

$$(R, G, B) \text{ bőr képpont ha:}$$

$$R > 95 \text{ és } G > 40 \text{ és } B > 20 \text{ és}$$

$$max\{R, G, B\} - min\{R, G, B\} > 15 \text{ és}$$

$$|R - G| > 15 \text{ és } R > G \text{ és } R > B$$

Ez a módszer nagyon egyszerű és gyors osztályozást eredményez. A módszer nehézségét a megfelelő színtér kiválasztása és a pontos szabályok meghatározása jelenti. Erre a problémára születtek megoldások. Ezek közül az egyik[12] egy RGB színtérből kiinduló tanuló algoritmus, amely mind a megfelelő színteret kiválasztja, mind a szükséges szabályokat előállítja.

Bayes osztályozó

A Bayes osztályozó egy paraméter nélküli osztályozó. A paraméter nélküli osztályozók ahelyett, hogy egy bőrszín egy explicit modelljével dolgoznának, az osztályozást tanuló adatok segítségével valósítják meg.

Ebben a hisztogram alapú megközelítésben, a színteret részekre, úgynevezett binekre osztják, amelyek egy-egy pontosan meghatározott színtartományt jellemeznek egy értékpárral (2D-s esetben) vagy értékhármassal (3D-s esetben). A binek 2 vagy 3 dimenziós hisztogramot alkotnak, melyre lookup table-ként(LUT) szokás hivatkozni. Minden binben tároljuk, hogy a tanító képen az adott tartományba eső színek hányszor fordultak elő. A tanítás után a hisztogramot normalizáljuk, és az értékeket diszkrét valószínűségi eloszlásokká konvertáljuk:

$$P_{skin}(c) = \frac{skin[c]}{Norm}$$

ahol a $skin[c]$ a hisztogram, a c színt leíró binjében tárolt érték, a $Norm$ pedig az összes bin értékének összege, vagy a legnagyobb bin érték.

Az így kiszámított $P_{skin}(c)$ érték valójában egy $P(c|skin)$ alakban írható feltételes valószínűség. Annak valószínűsége, hogy ha c színt látunk a képen, akkor az bőr képpont. Ennek kiszámítására a Bayes szabályt alkalmazzuk:

$$P(skin|c) = \frac{P(c|skin)P(skin)}{P(c|skin)P(skin) + P(c|\overline{skin})P(\overline{skin})}$$

Az erőforrás igényes számításokat elkerülhetjük, ha a valószínűségek pontos értékei helyett a $P(c|skin)$ és $P(c|\overline{skin})$ arányát vizsgáljuk. Az előző képletet felhasználva:

$$\frac{P(c|skin)}{P(c|\overline{skin})} = \frac{P(c|skin)P(skin)}{P(c|\overline{skin})P(\overline{skin})}$$

Ha bevezetünk egy küszöb értéket[13] és ehhez hasonlítjuk valószínűségek arányát, akkor a kapott szabály alapján eldönthetjük, hogy az adott képpont bőr képpont vagy sem:

$$\frac{P(c|skin)}{P(c|\overline{skin})} > \Theta$$

ahol

$$\Theta = K \times \frac{1 - \mathrm{P(skin)}}{\mathrm{P(skin)}}$$

Minden $P(skin)$ valószínűséghez választható olyan K érték, mellyel ugyanazt a Θ küszöböt kapjuk.

Gauss osztályozó

A bőr szín szerinti szegmentációjának egy másik, gyakran használt módja a paraméteres modellek használata. Ezek segítségével egy általános modellt kapunk, melynek használata kevesebb tanító adatot és tárhelyet igényel. Egy ilyen paraméteres modell a Gauss osztályozó.

Az úgynevezett Single Gauss osztályozó (SGM) csak kontrollált fényviszonyok között működik. Ekkor a bőrszín osztályozása normalizált színtérben modellezhető egy többváltozós normális(Gauss) eloszlással. Ez a modell egy elliptikus Gauss sűrűségfüggvényen(pdf) alapul, amit a következőképpen definiálnak:

$$p(c|skin) = \frac{1}{2\pi|\Sigma_s|^{1/2}} * e^{-\frac{1}{2}(c-\mu_s)^T\Sigma_s^{-1}(c-\mu_s)}$$

ahol c a színvektor és μ_s és Σ_s az eloszlás paraméterei. Ezen paraméterek a tanuló adatokból származtathatóak:

$$\mu_s = \frac{1}{n}\sum_{j=1}^{n} c_j$$

$$\Sigma_s = \frac{1}{n-1}\sum_{j=1}^{n}(c_j - \mu_s)(c_j - \mu_s)^T$$

ahol n, a c_j bőrszínminták számát jelenti.

A $p(c|skin)$ valószínűséget rögtön használhatjuk is a c szín „bőr szerűségének" eldöntésére, vagy a c szín és μ_s sajátérték vektor Mahalanobis távolságával, illetve a Σ_s kovariancia mátrix-segítségével számolva[14] is hasonló eredményt kapunk:

$$\lambda_s(c) = (c - \mu_s)^T{\Sigma_s}^{-1}(c - \mu_s)$$

Gauss keverék eloszlás

Egy ennél kifinomultabb modell a Gauss keverék modell, amely képes komplex eloszlások leírására is. Ez a modell, a Single Gauss modell általánosítása. A Single Gauss modellel ellentétben, ez a módszer nagyobb sikerrel használható változó fényviszonyok

mellett, valamint kevésbé érzékeny az etnikum miatti bőrszín különbségekre. Az eloszlásfüggvény a következőképpen néz ki:

$$p(c|skin) = \sum_{i=1}^{k} \pi_i * p_i(c|skin)$$

ahol k a komponensek száma, π_i a kombinálási paraméterek száma, $p_i(c|skin)$ Gauss eloszlásfüggvények, melyeknek külön sajátérték vektora és kovariancia mátrixa van. A normalizációs megszorítás miatt $\sum_{i=1}^{k} \pi_i = 1$.

A modell tanítása az úgynevezett EM algoritmussal történhet[4.7]. A bőrszín osztályozását pedig, a $p(c|skin)$ érték alapján, egy meghatározott küszöbhöz való hasonlításával lehet megvalósítani.

Elliptikus határ modell

Lee és Yoo[15] több szintér vizsgálatával kimutatta, hogy a szimpla Gauss modell nem közelíti jól különböző színterekben a bőrszín közel ellipszis alakú klaszterét. A Gauss modell, alapvető szimmetriája és a bőrszín klaszter aszimmetriája miatt sok esetben fals pozitív eredményt ad.

E probléma megoldására szolgál az elliptikus határ modell, mely hasonlóan egyszerű és gyors, mint a Gauss modell, de annál jobb eredményeket produkál. A következőképpen definiálható:

$$\Phi(c) = (c - \phi)^T \Lambda^{-1} (c - \phi)$$

A modell tanítása két lépcsős. Először a tanító adat 5%-át kiszűrik, mint zajos és elhanyagolható adatot, majd meghatározzák a modell paramétereit:

$$\phi = \frac{1}{n} \sum_{i=1}^{n} c_i \qquad \Lambda = \frac{1}{N} \sum_{i=1}^{n} f_i (c_i - \mu)(c_i - \mu)^T$$

$$\mu = \frac{1}{N} \sum_{i=1}^{n} f_i c_i \qquad N = \sum_{i=1}^{n} f_i$$

ahol n a különböző c_i tanító színvektorok száma és f_i a c_i szín vektor bőrmintáinak száma. Az osztályozást ennél a módszernél is egy küszöbérték meghatározásával valósíthatjuk meg:

$$\Phi(c) < \Theta.$$

4.2.2 Textúra alapú szegmentáció

A képfeldolgozás területén sokféle textúra alapú szegmentációs technikát használnak, melyek főleg szürkeskálás képekkel dolgoznak. Ezek jól használható módszerek, többek között az orvosi képfeldolgozásban, ipari területeken és dokumentum szegmentációhoz.

A bőrterületek textúrája a képeken általában simának mondható. A szín alapú szegmentáció legnagyobb problémája, hogy sok esetben ad fals pozitív eredményt. A képeken lehetnek olyan területek, amelyek a használt modell szerint bőr színűek, de mégsem tartoznak a bőrfelülethez. A textúra alapú technikákat a szín alapú technikákkal együtt használva javíthatunk a szegmentálás eredményén.

Gábor szűrő

A Gábor szűrő egy úgynevezett lineáris band-pass szűrő, mely egy választott hullámhossz körüli tartományon kívül szűr. Nagyon hasznos textúrák leírására. A Gábor szűrő impulzusválasza egy Gauss függvény és egy harmonikus függvény szorzatából áll elő.

A Gábor szűrő szoros összefüggésben van a Gábor waveletekkel. A Gábor waveletek használatához általában létrehoznak egy szűrő halmazt, melyben a Gábor szűrő skálázott és elforgatott változatai szerepelnek. Ezután egy konvolúciós eljárással úgy nevezett Gábor teret hoznak létre, amik nagyon jól használhatóak textúrák leírására. A [16]-ben a Gábor szűrőket Sobel éldetektálással együtt használva tudták megnövelni a bőr detektálásának pontosságát közel 25%-al, de a rendszer a rendszer érzékenysége körülbelül 3%-kal csökkent.

Együttes előfordulási mátrix

A kétdimenziós együttes előfordulási mátrix a képen lévő ismétlődő változások mérésének segítségével detektálja a textúrákat. A mátrix az egymástól adott távolságban és irányban elhelyezkedő, azonos szürkeárnyalatú pixelpárok számát adja meg. A mátrix felépítése után a kép több sajátságát határozhatjuk meg, például az entrópiát, kontrasztját, korrelációját és homogenitását. Ezen paraméterek segítségével tudjuk jellemezni az adott textúrát.

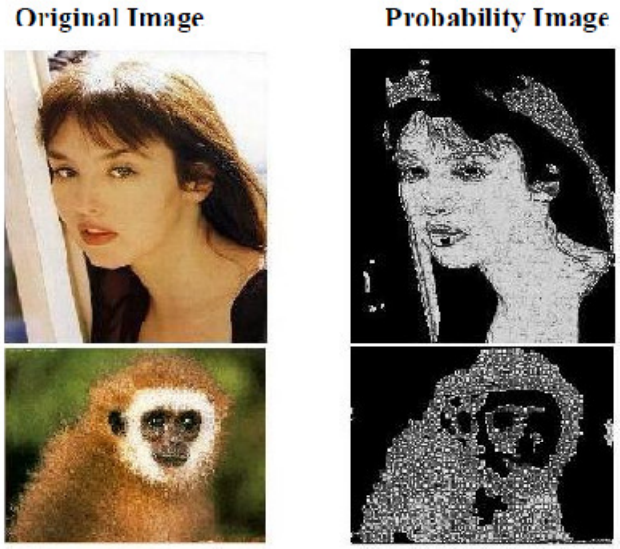
Az együttes előfordulási mátrix használatával kevésbé számításigényessé tehetjük a rendszert, de ezért cserébe a pontossága is romlani fog.

Ehhez nagyon hasonló eszköz az úgynevezett szomszédsági szürkeskálás különbség mátrix (NGTDM), amely az intenzitásváltozások mérésén alapul. Az NGTDM-ből származtatható képi sajátságok például a durvaság, a kontraszt és komplexitás.

Valószínűségi térkép

A [17]-ben a színinformációkat és a textúra tulajdonságait szintén együttesen használták a bőr detektálására. Ennek eredményeképpen azok a területek, melyek nem a bőrhöz tartoznak, viszont a színük azonos a bőrével, kiszűrhetőek a textúrájuk tulajdonságai alapján.

Első lépésben minden képponthoz, egy körülötte lévő $W \times W$ blokkban kiszámolják a képpont és a szomszédjai színének Euklideszi távolságát. Eztán a távolságok segítségével készítenek egy úgynevezett „durvasági térképet”. Az utolsó lépésben a kiszámított durvasági értékek segítségével elkészítik a valószínűségi térképet *(4.5 ábra)*.



4.5 ábra. A jobb oldali valószínűségi térképeken látható, hogy a bőr területeken simább, míg a bőr színű, de nem bőr területeken durvább képet kaptak. Forrás:[4.13].

Watershed algoritmus

A watershed transzformációt először 1979-ben Beucher és Lantuejoul használta szegmentációs problémák megoldására. Egy képen látható buborékokat szegmentáltak vele.

A watershed transzformáció az adott képen lévő gradiensek nagyságait egy topológiai felületként értelmezi. A kép jellemzőit és képpontjait alapul véve egy megfelelő leképezést használ, és így egy topológiai felületet ad, amelyen a magas értékek jelzik az eredeti képen lévő objektumok éleit.

A módszer a következő analógiát használja: képzeljünk el egy tájat hegyekkel és völgyekkel, ahol a kisebb völgyeket folyamatosan víz árasztja el. A víz mennyiségének növekedésével a kisebb völgyek egy idő után túlcsordulnak, és a víz átfolyik egy másik völgybe is, és így tovább: a kisebb völgyek (területek) egyesülnek nagyobb tavakká.

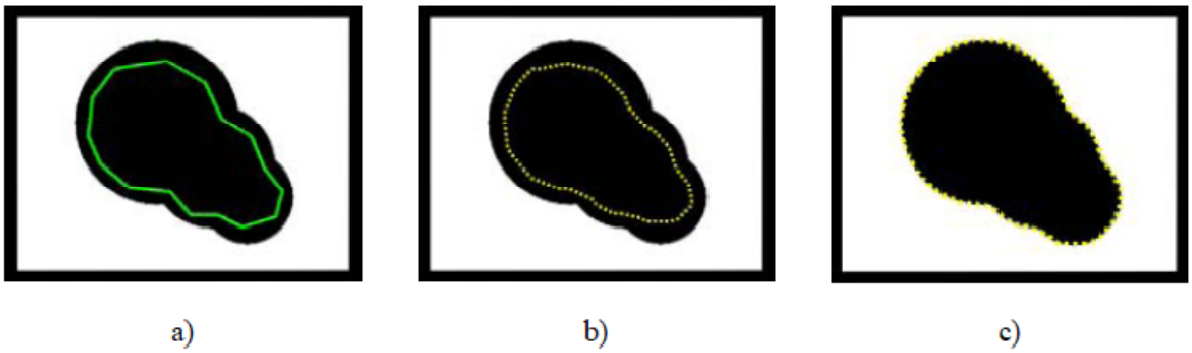
Ezzel a módszerre szegmentált képek hierarchiáját kapjuk, amelyből vagy valamilyen előzetes tudás alapján, vagy manuálisan próba-hiba módszerrel, ki kell választanunk a megfelelőt.

A táj túltöltődésének, vagyis a kép túlszegmentálásának megakadályozására bevezethetünk olyan paramétereket, amelyekkel befolyásolhatjuk a folyamatot. A [18]-ben arcdetektálásra használták ezt az algoritmust, és két kritériumot vezettek be. A depth (mélységi) és a hue (színárnyalat) kritériumot, melyek gátat szabtak az algoritmusnak, és így jó eredmény kaptak.

4.2.3 Aktív kontúr modell

Az aktív kontúr, vagy más néven snake, jól használható technika automatikus képszegmentálásra. E mellett modellezésre, éldetektálásra, alakzatmodellezésre és mozgáskövetésre is alkalmas. A modell központi eleme egy energiaminimalizálásra törekvő görbe, amely a képen található élekhez vagy objektumokhoz igazodik *(4.6 ábra)*. Az aktív kontúrt kontroll pontokkal definiálhatjuk.

A görbe alakját a formájából származtatott belső, és a kép színintenzitásaiból származtatott külső energiák határozzák meg. Az energiák meghatározásával lehet a görbét rásimítani az adott objektum éleire. A görbe viselkedése dinamikus, mert a modell folyamatosan minimalizálja a görbe energiáját leíró funkcionált.



4.6 ábra. A képsorozaton a snake működése figyelhető meg. A snake inicializáláskor(a), 5 iteráció után(b) és 12 iteráció után(c).

Az aktív kontúrok két típusát[19] különböztethetjük meg, a paraméteres és a geometriai aktív kontúrt.

A paraméteres aktív kontúrokat paraméteres görbékkel határozhatjuk meg, melyeket az említett külső és belső energiák befolyásolnak, és ettől az élek irányába mozdulnak. A görbe olyan formát vesz fel, ahol minimalizálja a külső és belső energiák súlyozott összegét.

A külső(external) erő egyik komponense, a potenciális energia, mely a legkisebb értékét azon a ponton veszi fel, ahol a legnagyobb az intenzitás gradiens, tehát az él mentén. A külső erő másik komponense a nyomóerő. A belső energia határozza meg a görbe tenzióját vagy simaságát. Az egyik komponense az elasztikus(elastic) erő, amely a görbe

összetartásáért felelős, a másik pedig az elhajlási(bending) erő, amely a túlhajlást akadályozza meg.

A geometriai aktív kontúr előnye a paraméteres változattal szemben, hogy egy görbe evolúciós elméleten alapulnak, melyben a görbének csak geometria szabályokhoz kell igazodnia, és nem meghatározott paraméterekhez. Ez az evolúció a kép alapján megy végbe.

Az aktív kontúr lehet nyitott vagy zárt. A modellnek két hiányossága van: a görbe inicializálását körültekintően kell végezni, valamint az objektum határán lévő konkáv részeket a modell nem tudja kezelni[20].

Matematikai háttér

Az aktív kontúr, az azt leíró kontroll pontok koordinátáinak halmaza. Paraméteres módon definiálható:

$$\vec{v}(s) = (\vec{x}(s), \vec{y}(s))$$

ahol $x(s)$ és $y(s)$ a kontúron lévő pontok koordinátái és s a kontroll pontok indexe.
A belső energia az elasztikus és hajlítási erők összegeként fejezhető ki:

$$E_{int} = E_{elastic} + E_{bend} = \alpha(s)\left|\frac{dv}{ds}\right|^2 + \beta(s)\left|\frac{d^2v}{ds^2}\right|^2$$

ahol α a görbe folyamatosságát kifejező, állítható konstans, β pedig a görbe görbeségét kifejező, szintén állítható konstans. Az elasztikus és elhajlási erők a következőképpen definiálhatóak:

$$E_{elastic} = \int_s \alpha(\vec{v}(s) - \vec{v}(s-1))^2\, ds$$

$$E_{bend} = \int_s \beta(\vec{v}(s-1) - \vec{v}(s) + \vec{v}(s+1))^2\, ds$$

Az energia minimalizált funkcionálját a következőképpen kapjuk:

$$E^*_{snake} = \int_0^1 E_{snake}(v(s))ds = \int_0^1 E_{int}(v(s)) + E_{image}(v(s)) + E_{con}(v(s))ds$$

ahol, E_{image} a kép energiája és E_{con} a görbe külső energiája.

A kép energiáját leíró funkcionál felelős azért, hogy az aktív görbe megközelítse az élt vagy más objektumot. A kép egész energiáját több különböző funkcionál súlyozott kombinációjaként állíthatjuk elő, például:

$$E_{image} = w_{line}E_{image} + w_{edge}E_{edge}$$

A legalapvetőbb kép funkcionál maga az intenzitás: $E_{line} = I(x, y)$. Ebben az esetben a snake a w_{line} súly előjelétől függően a legközelebbi világosabb vagy sötétebb egyeneshez fog tartani. Egy él megtalálása szintén egy egyszerű funkcionál segítségével kivitelezhető: $E_{edge} = |\nabla I(x, y)|^2$. Ebben az esetben a snake, a nagy gradiensű kontúrokhoz fog tartani.

Mivel a modell folyamatosan minimalizálja az energiáját leíró funkcionált, az aktív kontúrokkal lehetséges a mozgás követése is. Az él elmozdulása után a görbe a minimalizálás miatt követi az elmozdult élt. Megjegyzendő, hogy az élek túl gyors mozgása miatt a görbe átugorhat egy másik lokális minimumhelyre, de az átlagos kamerák sebességével a módszer jól működik.

Gradient Vector Flow

A módszer korábban említett hátrányainak kiküszöbölésére hozták létre a Gradient Vector Flow módszert. A módszer alapja egy vektormező, melyet a kép színintenzitásaiból hozhatunk létre energiafüggvények minimalizálásával. A minimalizálás néhány lineáris egyenletrendszer elvégzését jelenti. Az egyenletrendszer felírása, szürkeárnyalatos vagy bináris kép gradienseinek segítségével történik. Az ilyen vektormezőn mozgó snaket GVF-snake-nek nevezzük [21]

A GVF-snake előnye, hogy inicializálása könnyebb. A snake lehet az objektumon belül, vagy akár metszheti is az objektumot. Hátránya, hogy nagyon érzékeny a paraméterekre és rendkívül számításigényes.

4.2.4 Viola-Jones objektumdetektáló

Ez a módszer Paul Viola és Michael J. Jones nevéhez fűződik [22]. A detektáló arcfelismeréshez készült, de gesztusfelismerő rendszerekben is használható, ha a cél valósidejű detektálás. A rendszer érzéketlen a kép minőségére.

A Viola-Jones detektor nem színintenzitásokkal dolgozik, hanem képrégiók jellemzőivel, melyek a Haar-féle bázisfüggvényekre emlékeztetnek. A módszer egyik újítása az úgynevezett integrális képek használata, melyek segítségével a kép sajátságainak kiértékelése nagyon felgyorsul. Az integrális kép képpontonként számolható néhány elemi művelettel. Ennek segítségével a Haar-sajátságok kiszámolása minimális időt vesz igénybe.

Még a kép egy kisebb részét leíró Haar-sajátságok száma is nagyon nagy, ezért a rendszer a működés gyorsasága érdekében kiválasztja azon kritikus jellemzőket, melyek jól írják le a képet. Ilyen jellemzők például:

- két téglalap jellemző: két szomszédos téglalapban elhelyezkedő képpontok értékének különbsége,
- három téglalap jellemző: két szélső téglalapban elhelyezkedő képpontok összegét vonja ki a középső téglalap képpontjainak összegéből,
- négy téglalap jellemzők: átlós téglalap párok képpont összegeinek különbségét határozza meg.

A jellemzők kiválasztása után szükség van egy osztályozásra, amit általában egy tanuló algoritmussal valósítanak meg.

A módszer fontos tulajdonsága, hogy a megvalósításához több bonyolultabb osztályozót kombinálnak egy megfelelő struktúrában, ezzel növelve a detektor sebességét. Az ötlet alapja, hogy gyakran egyszerű eldönteni, hogy a keresett objektum a kép melyik részén fordulhat elő nagy valószínűséggel, ezért a keresést célszerű ezeken a területeken végrehajtani. A rendszer egy képen általában gyorsan megtalálja a keresett objektumot, ráadásul a fals pozitív eredmények száma alacsony.

A Viola-Jones detektor szürkeskálás képek esetén ér el gyors és pontosnak mondható eredményt. Színes képek használata esetén a rendszer először szürkeskálássá alakítja a képet, majd ezen dolgozik tovább. A rendszer gyorsaságából fakadóan alkalmas valósidejű gesztuskövetésre.

Az Integrál kép előállítása

A szürkeskálás képek tárolásának általános módja egy mátrix, amelynek elemei a kép egy-egy képpontjának intenzitásértékét tárolják. Az elemek értékei [0-255] egész számok intervallumából kerülnek ki.

Az integrál kép egy a képpel azonos méretű mátrix, melynek minden eleme a mátrixban tőle balra és fent elhelyezkedő téglalapban található értékek összegét tartalmazza.

4.3 Gesztuskövetés

Dinamikus gesztusok felismerését, például jelnyelv felismerést végző rendszerek esetén szükséges a gesztusok követése. A nagyobb rendszerek túlnyomó részt dinamikus gesztusokkal dolgoznak, így a gesztus osztályozása több kép alapján történik. A követés statikus gesztusokkal dolgozó rendszerek esetén is hasznos lehet. A rendszerek többsége nem osztályoz minden képet. A beérkező videó folyamból, valamilyen tudás alapján kiválasztja azt a képet, amelyen a gesztus látható és csak azt osztályozza.

Ahhoz, hogy az objektumot követni lehessen, először detektálni kell. A fejezet előző részében ismertetett eljárásokkal, vagy a célnak megfelelő egyéb szegmentációs módszerekkel meg kell határozni a keresett objektum helyzetét.

Az objektumkövetési módszerek lényege, hogy megadják a kérdéses objektum trajektóriáját(útvonalát) az egymás után érkező képeken való lokalizációjával. A követésnek kétféle megközelítése van:

- különálló esetben az objektum lehetséges helyét egy detektáló algoritmus keresi meg minden képen,
- kapcsolódó esetben a rendszer minden lépésben, az előző lépés eredményének ismeretében és felhasználásával keresi az objektumot.

Mindkét megközelítésben az objektum az alakját vagy küllemét leíró sajátságokkal írható le. Ez lehet akár az objektum színe, az élei vagy a textúrája. Ezeket a sajátosságokat a kép szegmentálása közben nyerhetjük ki. Attól függően, hogy az objektum leírására milyen sajátságot választunk, különböző típusú követési módszereket alkalmazhatunk[23]:

- pont alapú követés esetén a detektált objektumot pontok, valamint a közöttük fenn álló kapcsolatok írják le. Ebben az esetben a kérdéses objektumot minden képen külön kell detektálnunk. A pont alapú követési módszerek érzékenyek a képhibákra, valamint a detektálási folyamat hibáira.
- kernel alapú követés esetén az objektum alakját vagy küllemét leíró információk alapján valósítjuk meg a követést. Használhatunk például az objektumot körbezáró négyszöget, amelynek a képsorozaton történő mozgását követjük nyomon.

- sziluett alapú követés esetén minden képen meghatározzuk az objektum helyzetét és az objektumot leíró információk alapján követünk. Ebben az esetben valamilyen, az objektumot jól leíró modellel dolgozunk.

Egy objektum követéséhez több kamerát is használhatunk. Ennek egyik előnye, hogy mélységi információkat is használhatunk a követés folyamán. A másik előnye, hogy a két kamera a tér jóval nagyobb részét látja be. A kihívást ebben az esetben, a két kamera képének összehangolása jelenti, amit manuálisan és automatikus módszerekkel is megoldhatunk. Ezen felül problémát jelent, hogy a mélységi információkkal dolgozó algoritmusok számításigénye magas.

Pont alapú követés

Pont alapú követés esetén beszélhetünk determinisztikus és statisztikai módszerekről. A determinisztikus módszerek úgynevezett mozgási heurisztikákat használnak. Ezek a módszerek a pontok kapcsolatát a kapcsolat költsége alapján írják le. Egy, az aktuális képen lévő objektumhoz megadják, az előző kép összes objektumával való kapcsolatának költségét. A költségszámítást és a legalacsonyabb költségű kapcsolat kiválasztását többféle algoritmussal, például mohó keresési algoritmusokkal is meg lehet valósítani.

A statisztikai módszerek kevésbé érzékenyek a képzajra és az objektumok hirtelen irányváltásaira. Ezek a módszerek az objektum tulajdonságait, úgymint helyzetét, sebességét és gyorsulását egy állapottér alapú megközelítéssel modellezik. A modell eredményei általában megfelelnek a képen detektált objektum tulajdonságainak. Az objektum állapotának meghatározására használhatunk Kálmán szűrőt.

Kernel alapú követés

A kernel alapú követés esetén az objektumot egy geometriai primitív, például egy négyszög reprezentálja, melynek képről képre követjük a mozgását. Ez a mozgás általában valamilyen forgatás, vagy affin mozgás. A kernel alapú követő módszereknek két fajtájáról beszélhetünk: a sablonokról, valamint az úgynevezett Multiview leíró modellekről.

Az önálló objektumok követésére sablonillesztést használhatunk. A sablonokat általában színinformációkkal írjuk le, de a fényviszony változások miatti érzékenység

áthidalására gradiens értékekkel is dolgozhatunk. A módszer, brute-force jellege miatt nagyon számításigényes. Ezen javíthatunk, ha az objektum múltbeli helyzete alapján leszűkítjük a sablon illesztés területét.

A sablon illesztési eljárás esetén a sablonokat a rendszer a program futása során generálja, hirtelen irányváltás esetén a rendszer elveszítheti az objektumot. A módszer gyorsítására használhatunk úgynevezett Multiview leíró modelleket, melyek abban különböznek az előbbi megközelítéstől, hogy az objektumok különböző irányból vett nézeteit a rendszer egy tanító algoritmus segítségével, előre megtanulja.

Sziluett alapú követés

A sziluett alapú követés előnye az olyan komplexebb objektumok esetében jelenik meg, melyek nem túl jól követhetők geometriai primitívekkel. Ez a megközelítés minden képen megkeresi az objektumot, az előző kép által előállított modell segítségével. Ez a modell lehet hisztogram, él vagy kontúr alapú. A sziluett alapú követési módszereknek két csoportjáról beszélhetünk.

Alakillesztés használatakor a rendszer mindig megkeresi a keresett objektum sziluettjét az aktuális képen. Ez hasonlóan végezhető el, mint a sablonillesztés. Az aktuális képet megelőző képen lévő objektum sziluettje alapján meghatároz egy várható sziluettet és ezt illeszti. A használt modell általában egy él kép, melyet minden kép alapján frissítünk, ezzel elkerülve az esetleges nézőponti vagy megvilágítási problémákat.

A módszer egy kezdeti kontúrból indul ki. Ezután ezt a kontúrt minden lépésben hozzáigazítja az objektum aktuális helyzetéhez. Az igazítás elvégzéséhez szükséges feltétel, hogy az objektum csak annyira mozdulhat el az aktuális képen az előzőhöz képest, hogy még átfedésben maradjanak. A kontúr igazítására kétféle módszert használhatunk: az állapottér alapú modellt, és az aktív kontúr modellt. Az első módszer esetben az objektum állapota, a kontúr alakja és mozgási paraméterei alapján határozható meg. Az utóbbi módszer analógiája a már ismertetett aktív kontúr modellnek (*4.2.3)*.

4.4 Osztályozás

A rendszer utolsó lépése az osztályozás. Ahhoz, hogy osztályozni tudjuk a felismerni kívánt gesztust, valamilyen módon le kell tudnunk írni. Az előzőekben bemutatott módszerek egy részében fellelhetőek olyan lépések, melyek a gesztust leíró adatokat állítottak elő. Például egyes követési technikák esetén a trajektóriák, vagy az aktív kontúr modell esetén maga az előálló kontúr. Attól függően, hogy milyen leírókat (sajátságokat) szeretnénk használni, kinyerésükre az eszközök széles skálája áll rendelkezésünkre. A szegmentálási módszereken felül használhatunk például éldetektálási vagy szkeletonizációs technikákat, távolság transzformációt vagy mozgás analízist. Ezen felül a szegmentálás során megtalált tartomány geometria jellemezőit is felhasználhatjuk. Ezekből a sajátságokból készíthető az úgynevezett sajátság vektor, melynek segítségével az osztályozó algoritmusok el tudják végezni az osztályozást.

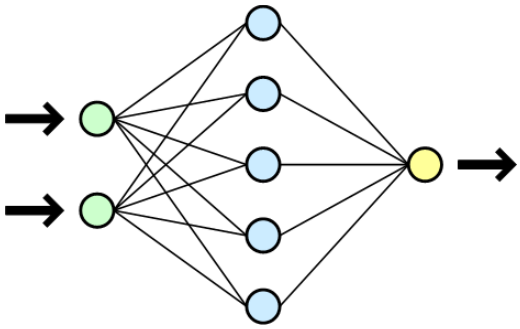
A sajátságok választásánál két dologra kell törekednünk. Egyrészről gyorsan számíthatóak legyenek, másfelől a választott sajátságok alapján jól el lehessen különíteni a felismerni kívánt gesztusokat.

Statikus gesztusok osztályozása esetén, többek között használhatunk mintaillesztést, geometriai leírók szerinti osztályozást vagy neurális hálókat. A dinamikus gesztusok osztályozásakor a plusz tényezőként megjelenő időtényező miatt ezek a technikák kevésbé vagy egyáltalán nem használhatóak. Ebben az esetben nagyon jó választás a Hidden Markov Modell alkalmazása.

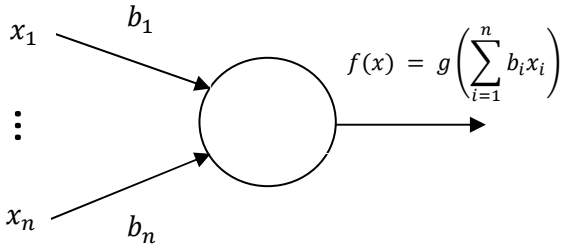
A Neurális hálók

Mint alapvető osztályozási technikát, a neurális hálókat *(4.7 ábra)* szeretném részletesebben ismertetni. Az elméleti hátterének bemutatása jóval túlmutat a dolgozat keretein, ezért csak a működésének elvét ismertetem. A neurális hálók elméletének megalkotásakor a követett minta az emberi agy működése volt.

A neurális hálók neuronokból *(4.8 ábra)* épülnek fel, melyek rétegekbe rendeződnek. A hálónak van egy kimeneti és egy bemeneti rétege, ezek között pedig rejtett rétegek helyezkednek el.



4.7 ábra. Egy rejtett réteggel rendelkező neurális háló egyszerűsített nézete.



4.8 ábra. Egy n darab bemenettel rendelkező neuron.

Minden neuronnak van egy úgynevezett aktivációs értéke, melynek alapján generál egy kimeneti értéket, melyet elküld a vele kapcsolatban álló többi neuronnak. Továbbá minden kapcsolathoz tartozik egy súly érték. Az aktivációs érték, a neuron bemeneti értékeinek(x_i) és a bemenethez tartozó súlyok(b_i) szorzatának összege. Ez alapján az úgynevezett aktivációs függvény(g) határozza meg a kimenetet. Tehát egy neuron kimenete hatással van a vele kapcsolatban álló neuronok aktivációjára.

A leggyakrabban használt neurális háló az úgynevezett többrétegű előrecsatolt neurális háló. Egy ilyen hálóban több rejtett réteg van. Az ilyen hálóban csak előrecsatolások vannak, vagyis egy adott rétegben lévő neuron nem küldi el kimenetét egy előző rétegben lévő neuronnak.

A neuronháló használatának első lépése a tanítás. Ennek során a háló egy tanítóvektor halmazt dolgoz fel. Ebben a halmazban bemenet-kimenet párok vannak. A tanítás folyamán azt szeretnénk elérni, hogy a neuronok közötti kapcsolatokhoz tartozó súlyokat, valamint az aktivációs függvény paraméterét sikerüljön úgy beállítani, hogy a háló a tanító halmaznak megfelelően, adott bemenetre a megfelelő kimenetet adja. A tanítás során figyelni kell arra, hogy nehogy túltanítsuk a hálót, mert ebben az esetben a tanítóhalmaz elemeire jó eredményt fog adni, de másra nem. A háló tanítására sokféle algoritmus használható, például a széles körben használt Backpropagation (hiba visszaterjesztő) algoritmus. Ennek az algoritmusnak az a lényege, hogy miután a hálón végigterjedt a bemenet, kiszámítjuk a hibát, majd az visszaterjed a hálóba. Ennek alapján a súlyok olyan értékeket vesznek fel, melynek eredményeképpen csökken hiba.

A betanított neurális háló már felhasználható osztályozási feladatra. Ekkor a háló bemenetét a korábbiakban előállított sajátság vektor képzi.

5 Multimodális kő-papír-olló játék

A Dr. Fazekas Attilával és Kovács Györggyel közösen fejlesztett multimodális Kő-papír-olló játék[24] fejlesztésekor az volt a célunk, hogy az ember-számítógép interface-ek tesztelésekor felmerülő egyik legnagyobb problémát hidaljuk át. Nevezetesen az ilyen rendszerek tesztelésekor az emberek annak tudatában, hogy megfigyelik őket, feszültté válhatnak, és ebből kifolyólag nem viselkednek természetesen.

Erre a problémára egy lehetséges megoldás, ha a tesztelő figyelmét elvonjuk a jelenlegi szituációról egy olyan környezetet teremtve, melyben a játék megnyerésére koncentrál. Ekkor a tesztelő reakcióinak megfigyelésekor kevesebb téves adatot kapunk.

5.1 A rendszer komponensei

A rendszer önállóan fejlesztett komponensekből áll. A rendszer bemeneteit a játékos beszéde, valamint arci- és kézi gesztusai alkotják. A kimeneteit egy virtuális ágensen megjelenő arci gesztusok és érzelmek, a játék során generált kézi gesztusok, valamint a virtuális ágens beszéde képezi. Az említett bementeket a rendszer különböző részei hasznosítják.

5.1.1 Arc analízis

A HCI kutatások mintájaként az emberi kommunikáció szolgál. Az emberek közötti kommunikációban nagyon fontosak az arcon megjelenő kifejezések és érzelmek, melyek hatást gyakorolnak a társalgás egészére, és mint ilyen, nagyon fontos tényezők az ember-gép interfészek működésének szempontjából is.

A rendszer, az arc elemzésért felelős része egy webkamera segítségével érzékeli a játékos arcát, és folyamatosan nyomon követi a megjelenő érzelmet. A megjelenő érzelmet a következő 4 kategória egyikbe sorolja: vidám, szomorú, unatkozó vagy természetes.

Ezt a komponenst az OpenCV programkönyvtár részeként implementált Viola-Jones osztályozó technikát felhasználva valósítottuk meg.

5.1.2 Beszédfelismerő

A rendszer nem igényelte komplex beszédfelismerő rendszer alkalmazását. A következő szavakat ismeri fel: „yes”,”no”,”rock”,”paper”,”scissors”. Ennek megvalósításához a HTK szoftver rendszert alkalmaztuk.

5.1.3 Talking Head

Az általunk használt első, érzelmek kifejezésére képes, magyar fejlesztésű Talking Head[25] a gépi játékos „megtestesítője”. A komponens bemenetei az egész rendszer kimenetnek szánt beszéd szöveges formában, valamint a megjelenítendő érzelem.

A Talking Head négy féle érzelem megjelenítésére képes: vidám, szomorú, unatkozó és természetes *(5.1 ábra)*.

A rendszer építésekor a foto realisztikus megjelenítés helyett, az emberszerű viselkedés megvalósítása volt a cél. A beszéd generálására a Profivox rendszert használtuk. A Talking Head ajakmozgása szabályosan követi a generált beszédet.

A Talking Head képes véletlenszerű megnyilvánulásokra: képes pislogni, véletlenszerű irányokba nézni és véletlenszerűen érzelmeket kifejezni. E mellett a játék állásától függően is mutat ki érzelmeket.

5.1 ábra. A Talking Head természetes és vidám arcai

5.1.4 Gesztusfelismerő

A gesztusfelismerő komponens a Kovács György által fejlesztett, OpenIP[26] rendszerre épül.

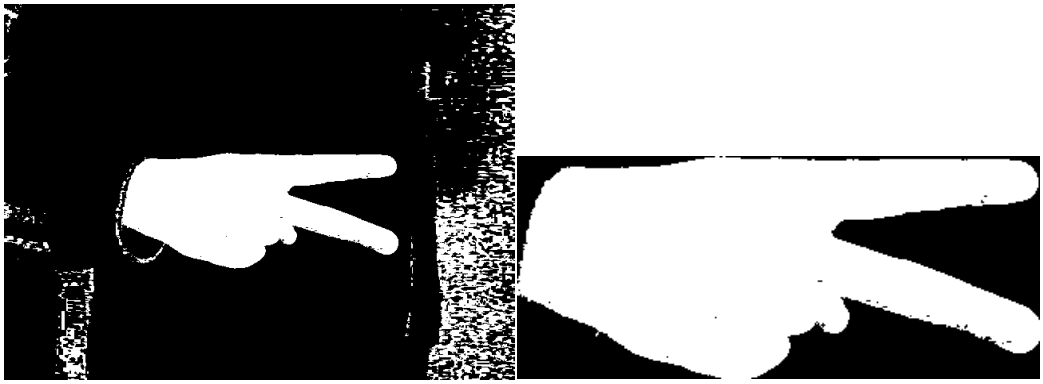
A játékos kezét a rendszer által használt két webkamera egyike követi. A játék lefolytatásához egyértelműen szükséges, hogy a játékos által mutatott kő, papír vagy olló jelet a rendszer felismerje.

A rendszer inicializálásakor beolvasásra és feldolgozásra kerülnek a gesztusfelismerést segítő mintaképek, melyek bekerülnek a VMS-be (Visual Memory System), amiben minden képhez eltároljuk a játékos gesztusainak osztályozásához szükséges adatokat és transzformáltakat.

A rendszer folyamatosan figyeli a játékos kezét és követi annak mozgását. Ezt a webkameráról érkező képek folyamatos szegmentálásával, és a detektált kéz középpontjának elmozdulásának követésével valósítja meg. Meghatározott szabályok segítségével meg tudja állapítani, hogy a kéz nyugalmi állapotban van vagy a három lendítést tartalmazó folyamat valamelyik szakaszában jár.

A harmadik lendítést követően a rendszer véletlenszerűen kiválasztja a gépi játékos által mutatott jelet, és megjeleníti a képernyőn. Ezután a játékos által mutatott jelet tartalmazó képet a gesztusfelismerő szegmentálja és osztályozza.

A szegmentálás első lépéseként a rendszer, explicit módon megadott szabályok alapján detektálja a képen található bőrszínű képpontokat. A bőrszínű képpontok által kijelölt tartományok kerületének és területének meghatározása után, kiválasztja a legnagyobb tartományt. A kamera által szolgáltatott kép esetleges alapvető, vagy a szegmentáció során keletkezett hibáinak javítására, a tartományt matematikai morfológiai eszközökkel, dilatációs és eróziós műveletekkel javítjuk *(5.2 ábra)*.



5.2 ábra. A szín szerinti szegmentáció eredménye, és a kiválasztott komponens, azaz a játékos által mutatott jel

Ezután a rendszer kinyeri a kiválasztott komponens kontúrját, majd meghatározza az euklideszi távolságtranszformáltját. Ennek a távolságtranszformáltnak és a VMS-ben tárolt gesztusok kontúrjainak segítségével mérőszámokat határozunk meg, melyek jól fejezik ki a detektálni kívánt gesztus és a VMS elemeinek hasonlóságát. Ennek alapján meghatározhatjuk, hogy a játékos kő, papír vagy olló jelet mutatott.

5.2 A játék folyamata

Ha a játékos leül a kamera elé, a rendszer megkérdezi, hogy szeretne e játszani. Ha a játékos igennel válaszol, a rendszer megvárja, míg a játékos megmozdítja a kezét. A játék akkor kezdődik el, ha a játékos egy gyors mozdulatot tesz a kezével. Ezután a játékosnak a kő-papír-olló játék szabályai szerint további két alkalommal kell meglendítenie a kezét. A harmadik lendítés idejét az első két lendítés alapján határozza meg a rendszer.

Ekkor a rendszer megpróbálja meghatározni, hogy a játékos által mutatott jel kő, papír vagy olló. Eközben véletlen módon megjeleníti a képernyőn a gépi játékos választását: egy kő, papír vagy olló jelet.

A rendszer megállapítja, hogy ki a nyertes, majd a Talking Head közli az eredményt a játékossal. A rendszer a Talking Headen ekkor megjelenő érzelmet néhány egyszerű szabály segítségével, a játék jelenlegi állása és a játékos arcán megjelenő érzelem alapján határozza meg.

6 Összefoglalás

A modern társadalmunkat előre hajtó tényezők közül az informatika fejlődése az egyik legfontosabb. A mindennapi életünket megkönnyítő, vagy munkánk szerves részét képező eszközök nagy része legalább részben ennek a tudományágnak köszönhető.

Az informatikai kutatások részterületei közül nagyon fontos az ember-gép kommunikáció területe. E terület kutatásával és új megoldások születésével a számítógépek kezelése, a velük való munka és a segítségükkel folytatott szórakozás a jövőben talán minden ember számára természetesebbé, emberközelibbé válik. Gondolok itt azokra az emberekre, akik talán koruk, neveltetésük vagy kulturális hátterük miatt idegenkednek, vagy egyáltalán nem érdeklődnek a számítógépek iránt még akkor sem, ha ez megkönnyítené az életüket, mindennapi munkavégzésüket. Napjainkban is léteznek Magyarországon olyan kisvállalkozások, melyek a leltárukat számítógép használata helyett papír alapú rendszerrel végzik. Hogy ennek csak pénzügyi okai vannak, vagy a számítógépektől való idegenkedés is közre játszik, nem tudhatom. Ennek ellenére az a véleményem, hogy az emberiség előre haladásában nagyon fontos tényező a gyors és pontos munkavégzés, aminek megvalósításában a számítógépek segítséget nyújthatnak.

Az összes mai tudományágról elmondható, hogy az általuk felgyűjtött tudásanyag hatalmas. Így van ez az informatikával is. Egy részterületének részterületét sem lehet teljes mértékben feldolgozni egy diplomamunka keretei között. Egy nagyon jó barátom egyszer azt mondta, hogy egy téma bemutatásakor nincs olyan, hogy készen vagyunk, csak a határidő érkezhet el. Az adott témához mindig hozzá lehetne tenni még valamit. Dolgozatom írása során beláttam, hogy teljesen igaza van.

Dolgozatomban az ember-gép kommunikációval kapcsolatos kutatások egyik fontos részterületébe, a gesztusfelismerő rendszerek témájába nyújtottam betekintést. Bemutattam az emberek közötti kommunikáció egyik fontos kellékét, a gesztusokat és szerepüket a mindennapi kommunikációban, valamint a felismerésükre képes rendszerek típusait.

Ezután szűken áttekintettem egy gesztusfelismerő rendszer építésének és működésének lépéseit, és bemutattam néhányat a megvalósításukhoz használható

módszerek közül. E lépések közül a szegmentációval és az ehhez kapcsolódó módszerekkel részletesebben foglalkoztam.

Dolgozatom utolsó részében a Dr. Fazekas Attilával és Kovács Györggyel közösen fejlesztett multimodális Kő-papír-olló játékot ismertettem.

7 Köszönetnyilvánítás

Szeretnék köszönetet mondani témavezetőmnek Dr. Fazekas Attilának a téma feltárása közben nyújtott útmutatásáért és tanácsaiért. Valamint szeretném megköszönni Kovács Györgynek a diplomamunkám elkészülését segítő beszélgetéseket, elméleti és gyakorlati tanácsokat.

8 Irodalomjegyzék

[1] G. Kurtenbach and E.A. Hulteen: Gestures in Human-Computer Communication , In Laurel, Brenda, Ed., *The Art of Human-Computer Interface Design.* Reading, Mass.: Addison-Wesley Publishing Co., May 1990.

[2] B. Rime, L. Schiaratura: Gesture and speech. In Fundamentals of Nonverbal Behavior, R. Feldman and B. Rime Eds. Press Syndicate of the University of Cambridge, New York, 239-281, 1991

[3] D. McNeill: *Hand and Mind: What Gestures Reveal About Tought.* University of Chicago Press. 1992

[4] T. Schloemer, B. Poppinga, N. Henze, and S. Boll: Gesture recognition with a wii controller. In *Proceedings of the Second International Conference on Tangible and Embedded Interaction (TEI'08)*. ACM, 2008.

[5] J. Kela, P. Korpipaa, J. Maentyjaervi, S. Kallio, G. Savino, L. Jozzo, and S. Marca. Accelerometer-based gesture control for a design environment. *Springer. Personal And Ubiquitous Computing*, 285–299, 2006.

[6] M. Deller, A. Ebert, M. Bender, and H. Hagen: Flexible gesture recognition for immersive virtual environments. In *Tenth International Conference on Information Visualization (IV 2006)*, pages 563–568. IEEE, July 2006.

[7] D.A. Socolinsky, A. Selinger, J.D. Neuheisel: Face recognition with visible and thermal infrared imagery, Comput. Vision Image,Understanding 91 72–114, 2003

[8] J. Yang, W. Lu, A. Waibel, Skin-color modeling and adaptation, ACCV98, 1998.

[9] M.H. Yang, N. Ahuja: Gaussian Mixture model for human skin color and its application in image and video databases, Proceedings of SPIE: Conference on Storage and Retrieval for Image and Video Databases, vol. 3656, pp. 458–466, 1999

[10] Q.H. Thu, M. Meguro, M. Kaneko: Skin-color extraction in images with complex background and varying illumination, Sixth IEEE Workshop on Applications of Computer Vision, 2002.

[11] Peer, P., Kovac, J., Solina, F.: Human skin colour clustering for face detection. In submitted to Eurocon 2003 – International Conference on Computer as a Tool, 2003.

[12] Gomez, G., Morales: E. Automatic feature construction and a simple rule induction algorithm for skin detection. In Proc of the ICML Workshop on Machine Learning in Computer Vision, 31-38, 2002

[13] Jones, M. J., Rehg, J. M.: Statistical color models with application to skin detection. In Proc of the CVPR ’99, vol. 1, 274-280, 1999

[14] Terrillon, J.-C., Shirazi, M. N., Fukamachi, H., Akamatsu S.: Comparative performance of different skin chrominance models and chrominance spaces for automatic detection of human faces in color images. In Proc. of the International Conference on Face and Gesture Recognition, 54-61, 2000

[15] Lee, J. Y., Yoo, S. I.: An elliptical boundary model for skin color detection. In Proc of the 2002 International Conference on Imaging Science, Systems and Technology, 2002

[16] F. Jiao, W. Gao, L. Duan, G. Cui: Detecting Adult Image using Multiple Features *ICII 2001*.

[17] Abin, A.A., Fotouhi, M., Kasaei, S.: *Skin segmentation based* on *cellular learning automata,* 6th International Conference on Advances in Mobile Computing and Multimedia, 2008

[18] Guerfi, S., Gambotto, J.P., Lelandais, S.: Implementation of the watershed method in the hsi color space for the face extraction. In: IEEE Conference on Advanced Video and Signal Based Surveillance, 2005. AVSS 2005, pp. 282-286. IEEE Computer Society Press, Los Alamitos (2005).

[19] C. Xu and J. L. Prince: Snakes, Shapes, Gradient Vector Flow, IEEE Transactions on ImageProcessing

[20] B. Leroy, I. Herlin and L. D. Cohen: Multi-Resolution Algorithms for Active Contour Modells, 1996

[21] L. D. Cohen and I. Cohen: Finite-Element Methods for Active Contour Modells and Balloons for 2-D and 3-D Images, 1993

[22] Paul Viola and Michael J. Jones.: Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE CVPR, 2001.

[23] Yilmaz, A., Javed, O. & Shah, M.: Object tracking: A survey. *ACM Computing Surveys* 38, 13, 2006

[24] Gy. Kovács, Cs. Makara, A. Fazekas: The Multi-modal Rock-Paper-Scissors Game, in Proc. Of International Conference on Intelligent Virtual Agens, from 2009-09-14 to 2009-09-16, Amsterdam, The Netherlands, 546--565

[25] Gy. Kovács, Zs. Ruttkay and A. Fazekas: Virtual Chess Player with Emotions, in Proc of Fourth Hungarian Conference on Computer Graphics and Geometry pp. 182-188, 2007

[26] OpenIP rendszer: http://code.google.com/p/openip