1. **dia**

Szeretettel köszöntök mindenkit, Bertók Kornél vagyok a Debreceni Egyetemről és a prezentációmban egy olyan rendszert fogok ismertetni, mellyel lehetőségünk nyílik fejmozgás alapú gesztusok felismerésére.

1. **dia**

Először a gesztusfelismerő rendszer kapcsán definiálni kell, hogy pontosan milyen gesztusokat akarunk ismertetni. A rendszerünk csak tudatos fejmozdulatok, mint mozdulatsorok felismerésére korlátozódik. Valós idejű felismerést valósítottunk meg egyszerű kameraképeken, mint képszekvenciákon. Téve mindezt viszonylag kevés információ, azaz kevés adat alapján.

A felismerni kívánt gesztusokra vonatkozóan megadtunk egy hatékony reprezentációt, mellyel mind térben, mind időben lehetőség nyílt a gesztusok szegmentálására. Valamint a felismeréshez létrehoztunk egy gesztusadatbázist, mellyel lehetőségünk nyílt a felismerés javítására, a felismert gesztusok rögzítése által, továbbá az adatbázis segítségével a jövőben tovább elemezhetőek a tudatos fejmozgások.

1. **dia**

A fejmozgás meghatározása előtt először célszerű annak a főbb tulajdonságait Tudatos fejmozgásokról beszélünk, így az mondható el, hogy egy gesztus végrehajtásának az ideje kb. 2-3 sec. Illetve fontos leszögezni, hogy az emberek általában nincsenek tudatában annak, hogy ugyanazt a mozdulatsort hogyan végzik el többször egymás után, így nagy valószínűséggel különböző ütemben fogják mindezt megtenni, vagyis a felismeréshez nem-lineáris illesztésre lesz szükség.

A felismerés előtt gesztus-szegmensekre kell bontani a fejmozgást. Ennek a tér-, és időbeli meghatározására az úgynevezett MHI reprezentációból indulunk ki. Amely lényegében egy időtől függő sablonként értelmezhető, ahol minden pixel értéke a fejmozgásnak és az időnek egy függvénye.

1. **dia**

Kicsit konkrétabban megfogalmazva az MHI egy képszekvencia mozgó objektumainak változásait írja le. A mozgás leírása több egymást követő képkockán keresztül történik és felhasznál hozzá egy maszkot, mely azokat a régiókat jelöli ki a szekvencia tau időbélyegű képén, melyek elmozdultak az ezt megelőző képhez képest. Az MHI reprezentációban tau értéket vesznek fel azok a pixelek, amelyek elmozdultak az előző képkockához képest, ezzel szemben a többi pixel fokozatosan elhalványul, végül idézőjelben „törlődik”. Csak úgy, mint ahogy az ábrán is látható.

1. **dia**

A fejmozgás irányát meg lehetne határozni az MHI reprezentációból származtatott gradiensek segítségével is (amelyeknek a fejpozíció megváltozásának az irányát kellene leírniuk), de a gyakorlati tapasztalatok során mi ennek az ellenkezőjét tapasztaltuk. Nem volt elég stabil a mozgás gradiensekre alkalmazott módszer. Talán túl kicsi az arcon belüli mozgó régiók területe.

Az MHI segítségével viszont meg tudjuk válaszolni azt a kérdést, hogy melyek azok a régiók a képen, melyek a mozgás tekintetében értékes információt kódolnak.

Egy FAST sarokdetektort alkalmaztunk a képre, mely véges számú jellemzőponttal tér vissza. A FAST eredményét pedig korlátoztuk az arcot tartalmazó ablakra.

1. **dia**

Következő lépésben célravezetőnek láttuk az optikai áramláshoz tartozó vektorokat kiszámítani a FAST által meghatározott jellemzőpontokra a szekvencia aktuális képkockája és az azt követő képkocka között.

Ugyanis ezekkel a vektorokkal lehetőségünk nyílt definiálni a fejmozgás irányát. Ehhez első lépésben képeztünk az „n” darab optikai folyam vektor számtani közepét. Valamint az átlátható kezelés érdekében egy szöget is rendeltünk a vektorokhoz, mégpedig az adott vektornak az „y” tengely pozitív oldalával bezárt szögét.

Itt azonban egy újabb problémával találtuk szembe magunkat: [0, 2PI] intervallumban értelmezett szögek túl sok értéket vehetnek fel (még akkor is, ha csak az egész értékeket nézzük), amely megnehezíti az illesztést. Ezért a számított szögekhez a teljes szög egy tizenhatodát rendeltük, mégpedig azt, amelyikbe esik. Tehát az illesztés során, majd csak 16 különböző szögértéket kell figyelembe venni.

1. **Dia**

Két szomszédos képkocka közötti fejmozdulat értéke szögben.

1. **dia**

Az előző módszer a gesztusok térbeli meghatározását hivatott segíteni, azonban a felismeréshez szükség van időben is szegmentálni a fejmozgást. Ezt az MHI átlagintenzitásának felhasználásával értük el. Egy időbeli szegmens alatt képkockán olyan sorozatát értjük, melyek kezdetén és végén alacsony az MHI átlagintenzitása.

Ezzel végre megadhatjuk a gesztus definícióját is: gesztus alatt a szekvencia szomszédos tagjaira számított szögek sorozatát értjük. Lásd példa.

1. **dia**

Azt mondtuk, hogy a gesztusok eltérő ütemben kerülnek végrehajtásra, ez azt jelenti, hogy különböző hosszúságú szögsorozatokat kell egymáshoz illesztenünk. Létrehoztunk egy gesztus adatbázist, abból a célból, hogy komplexebb döntéseket tudjunk hozni.

A gesztus adatbázis lényegében szögsorozatok gyűjteménye, melyben a sorozatokon bevezettünk egy osztályozást, így egy gesztushoz több sorozat létezik. Az adatbázis célja tehát a felismerés javítása, hiszen egyrészt több előre definiált gesztushoz tudunk illeszteni, másrészt az egyszer már felismert gesztusokkal ki lehet egészíteni az adatbázist, mely által jobban igazodunk a felhasználói szokásokhoz.

1. **dia**

A képszekvencia szegmenseihez számított szögsorozatokat a dinamikus idővetemítés c. eljárás segítségével illesztjük az adatbázis elemeihez. A felismert gesztust abba az osztályba soroljuk, mellyel a legkisebb az átlagos távolság.

DTW segítségével egy nD és mD vektort tudunk egymáshoz illeszteni. Lényegében egy táblázat kitöltését végezzük el, hogy meghatározzuk azt az optimális utat, mellyel el tudunk jutni a (0,0) pontból az (n,m) pontba.

1. **dia**

Lásd ábra.

1. **dia**

A rendszer tesztelése során egyrészt megtaláltuk azokat a paramétereket, mellyel a DTW algoritmus az adatbázis alapján kellő pontossággal tudja diszkriminálni a gesztusokat. Ezek a paraméterek a következőek:

Minimális gesztushossz. A használatára azért van szükség, mert a túl kicsi gesztusokat a DTW könnyen ráilleszti a kicsit is hasonlóakra. Jelen esetben min. 1 sec.

Maximális gesztushossz: ez befolyásolja a DTW futási idejét.