



A Mobile 3D technológia lehetőségeinek kihasználása a térbeli tájékozódásban

Tukora B.

Pécsi Tudományegyetem, Pollack Mihály Műszaki Kar, Műszaki Informatika Tanszék, 7624 Pécs, Rókus utca 2.

ÖSSZEFOGLALÁS

A legújabb technológiai fejlesztések lehetővé teszik, hogy a globális helymeghatározó rendszerrel (GPS) nem rendelkező mobil eszközök (mobiltelefonok, PDA-k) a beépített kamerájuk által szolgáltatott vizuális információk alapján meghatározzák saját helyzetüket a térben, környezetüket virtuális objektumokkal egészítsék ki, és az ily módon létrehozott világot megjelenítsék a képernyőjükön. E koncepció egyik lehetséges felhasználási területe a régészeti ásatások, alaprajz szintjén megmaradt utca- vagy városrészletek eredeti állapotának bemutatása úgy, hogy a szemlélő mobil eszközének kijelzőjén mindig az aktuális helyzetének és orientációjának megfelelően rajzolódik ki a régvolt környezet. Előnye a már meglévő, hasonló célú rendszerekkel szemben az olcsósága, rugalmassága és széles körű használhatósága. A mobil eszköznek mindeközben igen összetett feladatot kell elvégeznie. A dolgozat a rendszer fejlesztése során felmerülő problémákat tárgyalja és ad rájuk megoldási javaslatokat.

(Kulcsszavak: mobil technológia, kamera kalibráció, 3D rekonstrukció, modellezés egyetlen kép alapján)

ABSTRACT

Utilizing the Mobile 3D technology's potentials in the spatial orientation

B. Tukora

University of Pécs, Pollack Mihály Faculty of Engineering, Department of Information Technics, H-7624 Pécs, Rókus u. 2.

Recent developments make it possible for the mobile devices without GPS system (such as mobile phones and PDA-s) to determine their spatial situation on the visual informations given by the embedded camera, complete the environment with virtual objects, and display the created world. This conception can be used at the visualisation of archeological fossils and the original condition of ruined sections of streets or towns, where the bygone environment is shown according to the viewer's position and orientation. Comparing with the similar existing systems the advantages are cheapness, flexibility and the possible wide range of usage. The mobile device used for this purpose must accomplish a very complex task. This paper discusses the arising problems during the development and gives proposals for solving them.

(Keywords: mobile technology, camera calibration, 3D reconstruction, single view modelling)

BEVEZETÉS

A korszerű mobiltelefonok a Mobile 3D technológia segítségével képesek az OpenGL szabvány által definiált 3 dimenziós objektumok, világok megjelenítésére Java platformon. A szabványosított fejlesztői környezetet a JSR-184 Java specifikáció rögzíti. Szintén

rögzített az eszközökhöz tartozó digitális fényképezőgépek kezelésének (JSR-135) illetve a kliens-szerver kommunikáció (JSR-120) programozásának módja is. A dolgozat egy olyan, mobil eszközökön futó alkalmazás fontosabb funkcióit tárgyalja, amely

- a környezetében jelen lévő referenciapontokat kellő biztonsággal azonosítja,
- a referenciapontok alapján meghatározza saját pozícióját és orientációját a térben,
- ennek megfelelően a memóriájában tárolt 3 dimenziós objektumokat beilleszti a környezetről készült képbe,
- végül megjeleníti virtuális elemekkel kiegészített helyszínt a kijelzőjén.

ANYAG ÉS MÓDSZER

A referenciapontok azonosítása

A mobil eszközön futó alkalmazás első feladata az, hogy beépített fényképezőgépével képet készítsen a helyszínről, felismerje rajta a környezet térbeli helyzetéről áruklódó elemeket, és ezek pozícióját képernyő-koordináta-rendszerben (2D) tárolja. Egy olyan algoritmus lenne az ideális, amellyel egyértelműen azonosítani tudnánk a képen megjelenő objektumokat: házakat, köztéri tárgyakat. Sajnos ilyen – általánosan használható, egyszerű betanítási funkcióval bíró – módszer nem létezik. A megoldást külön e célra készített referenciatárgyak, -pontok, vagy -ábrák környezetbe helyezése jelenti. Az ezekkel szemben támasztott követelmények: a könnyű felismerhetőség, a képen szereplő tárgyakról való biztos megkülönböztethetőség. Ezek kielégítéséhez kell megtalálnunk a megfelelő szint (színeket) és formát.

A beépített fényképezőgéppel készült képeket a Java-specifikáció szerint raszteres formában, RGB színrendszerben kódolva kapjuk meg. Ez azt jelenti, hogy minden egyes képpontot három szám jellemez: a pont színének vörös (Red), zöld (Green) és kék (Blue) komponensei. Amennyiben az összetevők mennyiségét a 0..1 skálán adjuk meg, egy tiszta vörös képpont az 1;0;0 értékkel bír. A színelméleti alapoknak megfelelően a 0;0;0 a fekete, az 1;1;1 a fehér színre jellemző érték. Amennyiben a vörös színt világosítani szeretnénk, a vörös mellett a zöld és kék összetevők arányát is növelnünk kell (azonos mértékben): pl. 1;0,2;0,2. Egy tárgyat a természetben mindig más és más színűnek látunk: a környezeti fényeknek megfelelően sötétebbnek vagy világosabbnak, eltérő árnyalatúnak. A mobil eszközök egyszerű automata kameráinál fokozottan érvényesül ez a jelenség: egy vörös színű tárgy fehér háttér előtt szinte feketének látszik, mivel az automatika az egész kép átlaga alapján számítja ki a középtónust, így a fehér „elszürkül”, a sötétebb tárgyak feketék lesznek. Egy mobiltelefonnal tehát nem lehet „egy bizonyos” szint megkeresni, csak „hozzá hasonlót”.

Az RGB rendszerben a környezettől leginkább megkülönböztethető színek azok, amelyekben valamelyik komponens jóval kisebb vagy nagyobb arányban szerepel, mint a másik kettő. Ilyenek az alapszínek – a vörös (1;0;0), a zöld (0;1;0) és a kék (0;0;1) – mellett az ezek additív keveréséből származók: a sárga (1;1;0), a lila (1;0;1) és a zöldeskék (0;1;1). Ha a referenciapontokat középtónusú szürke alapon az említett színek valamelyikéből állítjuk elő, nagy esélyünk van arra, hogy különböző fényviszonyok mellett is felismerhetők maradnak. A meglévő bizonytalanságok miatt az eljárás sikere jelentősen múlik azon, hogy a lehető legkevesebb színt használjuk fel az azonosításhoz. Természetesen más tárgyaknak (ruháknak, hirdetéseknek) is lehet a referenciapontokhoz hasonló színük. Az egyértelmű azonosítás végett a pontokat mintázatba, összetett ábrába (tárgyra) kell helyezniük. A mintázatoknak könnyen dekódolhatóknak, felismerhetőeknek kell lenniük. Egy szürke alapon vörös négyzet például egyszerre négy pontot definiál, egyszerűen kiszámolhatók a sarokpontok koordinátái, alakja bizonyos

határok között könnyen ellenőrizhető, a szürke és vörös közötti csekély tónuskülönbség egyensúlyt ad a fényviszonyokban –, viszont a képen más vörös téglalapok is előfordulhatnak, és négy egyenrangú pont nem ad információt a négyzet orientációjáról. Ahhoz, hogy konkrétan milyen referenciatárgy vagy -ábra kerüljön felhasználásra, tudnunk kell, hogy a pozíció és orientáció meghatározását végző algoritmusnak hány, és milyen elrendezésű referenciapontra van szüksége –, így erre a kérdésre választ csak a következő fejezet után adhatunk.

A kamera pozíciójának és orientációjának meghatározása

A feladatunk az, hogy a referenciapontok alapján meghatározzuk a mobil eszközünk helyzetét és irányultságát a környezet tárgyaihoz viszonyítva. Ehhez definiálnunk kell egy világ-koordinátarendszert a helyszín egy tetszőleges pontján, tetszőleges (de logikus) irányultsággal. A referenciapontok egyértelműen utalnak e koordináta-rendszer origójának helyzetére és egységvektorainak irányára, nagyságára. Mindeközben az eszköz memóriájában egy „virtuális” koordináta-rendszerben definiált objektumok várakoznak. Amennyiben meg tudjuk határozni a világ-rendszerben a kamera helyzetét, és ugyanezen paraméterekkel állítjuk be a kamera virtuális koordináta-rendszerhez viszonyított helyzetét, a két koordináta-rendszer fedésbe kerül egymással: a „külső” és „belső” világ összeolvasható.

Legegyszerűbb esetben a referenciapontok világ és virtuális koordináta-rendszerbeli helyzete megegyezik, így a világ-koordinátarendszerhez viszonyított helyzet meghatározásával automatikusan megkapjuk a virtuális-rendszerbeli helyzetünket is.

A 3 dimenziós pontok a középpontos (perspektívus) projekció szabályai szerint vetülnek a képernyő síkjára (1. ábra). Középpontos vetítés során minden egyes térpont, illetve a vetítési síkon előállt képe a kamera gyűjtőpontjából kiindul egyazon egyenesre esik. Mivel a vetítéskor a mélységre vonatkozó adat elveszik, a perspektívus projekció elvileg nem invertálható folyamat: ez adja a számítás nehézségét.

1. ábra

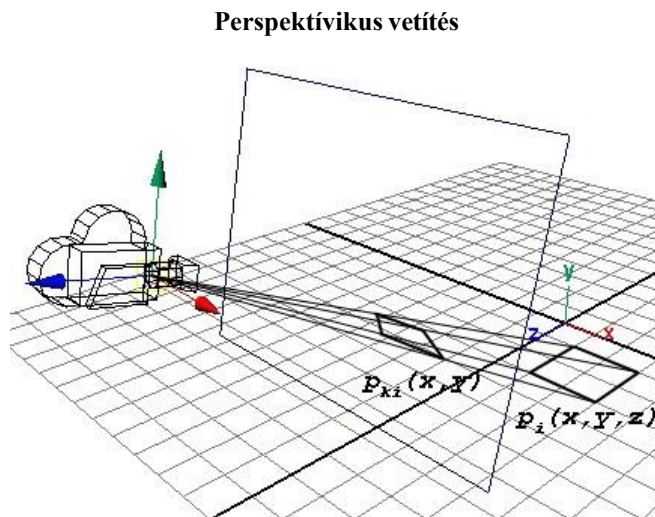


Figure 1: Perspective projection

Számításunk kiinduló értékei az 1. ábra szerint:

$$\underline{p} = [p_1, p_2, \dots, p_n] = \begin{bmatrix} p_{1x} & p_{1y} & p_{1z} \\ p_{2x} & p_{2y} & p_{2z} \\ \dots & \dots & \dots \\ p_{nx} & p_{ny} & p_{nz} \end{bmatrix}, \text{ vagyis a referenciapontok 3 dimenziós}$$

koordinátái a világ-koordinátarendszerben, illetve

$$\underline{p} = [p_1, p_2, \dots, p_n] = \begin{bmatrix} p_{k1x} & p_{k1y} \\ p_{k2x} & p_{k2y} \\ \dots & \dots \\ p_{knx} & p_{kny} \end{bmatrix}, \text{ a referenciapontok 2 dimenziós}$$

koordinátái a képernyő-koordinátarendszerben.

A kamera helyzetét leíró paramétereket homogén transzformációs mátrix formájában keressük:

$$\underline{H} = \begin{bmatrix} \underline{R} & \underline{k} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & e_3 & k \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} e_{1x} & e_{1y} & e_{1z} & k_x \\ e_{2x} & e_{2y} & e_{2z} & k_y \\ e_{3x} & e_{3y} & e_{3z} & k_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ ahol}$$

$\underline{R}$ a kamerához rendelt koordináta-rendszer forgatómátrixa az alap-koordinátarendszerben, $\underline{e}_1$, $\underline{e}_2$, $\underline{e}_3$ a kamera-koordinátarendszer egységvektorai az alap-koordinátarendszerben, $\underline{k}$ pedig a kamera-koordinátarendszer origójának pozíciója az alap-koordinátarendszerben (2. ábra).

2. ábra

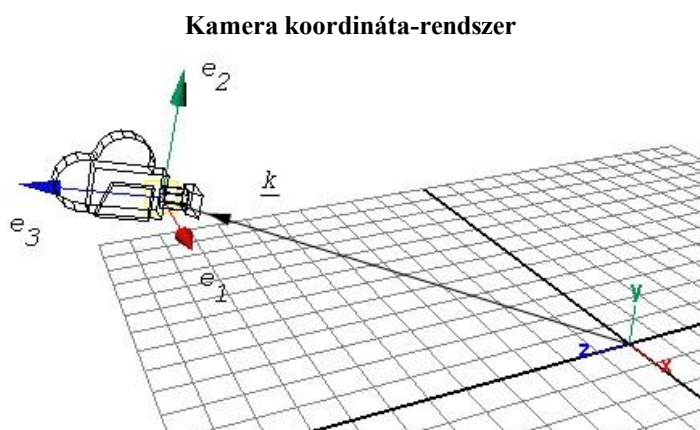


Figure 2: Camera coordinate-system

Feladatunk tehát olyan, a lehető legkevesebb tagból álló referenciapont-halmaz keresése, amely segítségével a $\underline{H}$ homogén transzformációs mátrix elemei kiszámíthatók.

Számítási módszer az eltűnési pontok alapján

A keresett megoldás a világ-koordinátarendszer x-y alapsíkjában elhelyezkedő egyetlen

négyszög csúcsai alapján geometriai módszerrel kiszámítható (Haralick, 1988), abban az esetben, ha ismerjük a kamera fókusz távolságát (a gyújtópont és a vetítési sík közti távolságot). Ezzel az a probléma, hogy ha általános célú alkalmazást szeretnénk írni, a programunkat futtató valamennyi mobil eszköz beépített kamerájának a fókusz távolságát ismernünk kellene. Esetleg előzetes kalibrációval a probléma megoldható lenne (például egy referenciakép adott távolságról való lefényképezésével), de ezzel a rendszer jelentősen veszítene rugalmasságából, hordozhatóságából. Olyan módszer is létezik, amelyhez nem kell előzetesen megadnunk a fókusz távolságot (Blaskó, 2000): itt egy 7 ismeretlenes (a kamera 3 térbeli koordinátája, 3 forgatási szöge és a fókusz távolság) nemlineáris egyenletrendszer iterációs megoldása adja az eredményt. A mobil eszközök teljesítménye azonban ehhez az algoritmushoz még gyenge. A fontosabb 3 dimenziós tervezőprogramok hat térpont alapján számítanak (ld. 3D Studio Max programot), de itt a pontoknak két különálló síkban kell lenniük. Az igazi megoldás az lenne, ha sikerülne egyetlen síkban elhelyezkedő referenciapontok képe alapján a fókusz távolság tudta nélkül meghatározni a paramétereket, elfogadható sebességgel.

Ahhoz, hogy ilyen egyszerű algoritmust írjunk, ki kell használni a képen megjelenő rejtett információkat is. Az eltűnési egyeneseket, illetve horizontvonalakat régóta használják nem rögzített fényképezőgéppel lefotózott mintázatok, szövegek síkjának meghatározásához, a minta vagy szöveg kiterítéséhez. Ez adta az ötletet ahhoz, hogy hasonló elven működő algoritmust keressek a jelen problémához.

Tudjuk, hogy az egy síkban elhelyezkedő párhuzamosok perspektívikus vetítés esetén (mint ahogyan az emberi szem is leképzei a környezetét) a „végtelenben” találkoznak (3. ábra). A végtelenbe futó egyenesek az ún. eltűnési egyenesek, a találkozási helyek pedig az eltűnési pontok. Egy sík eltűnési pontjai egy egyenesen vannak: a sík horizontján.

3. ábra

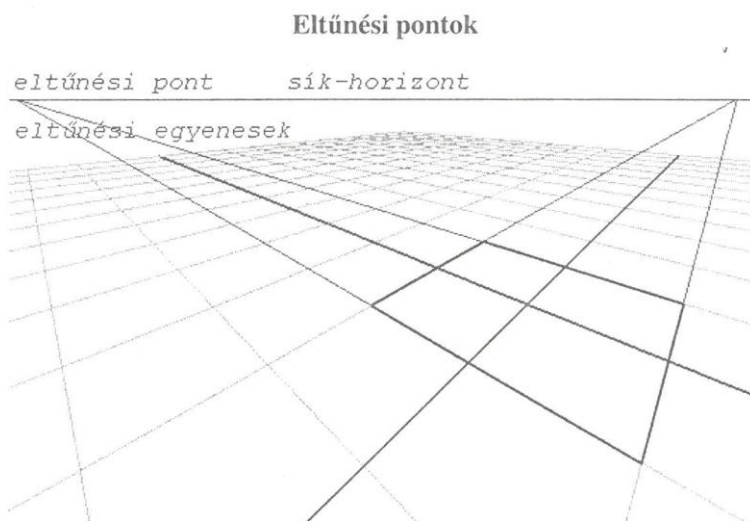


Figure 3: Vanishing points

Az eltűnési pontok és a horizont helyzete a képen a kamera orientációjától függ: ezek szerint ha ki tudjuk számolni a négy pont által meghatározott két eltűnési pontot, kiszámíthatjuk a kívánt irányultságot. Mivel a fókusz távolság változásával a kép

kisebb vagy nagyobb lesz, de belső arányai nem változnak (hiszen csak a vetítési síkot közelítjük, illetve távolítjuk a gyújtóponttól), a fókusztávolságtól függetlenül ugyanazon értékeket kell kapnunk. A négyzet képernyő-koordinátáinak abszolút értékeiből pedig az orientáció ismeretében visszaszámálható a fókusztávolság, illetve a kamera pozíciója. A négy sarokpont egyenrangú, ezért nem állapítható meg, hogy a világ-koordináta-rendszer x-y alapsíkjának melyik síknegyedéből szemlélődünk. Az algoritmushoz tehát elegendő egy négyzet síkbeli képe, de egyik csúcsának mindenképpen megkülönböztetettnek kell lennie.

Ha megfigyeljük az ábrát, észrevehetjük, hogy minél kisebb szöget zár be a horizonttal a négyzet képének alsó és felső oldala, annál közelebb kerül az egyik eltűnési pont a kép középhez, a másik viszont a végtelenbe távozik a kép valamelyik oldalán. Nagyon kicsi szögeknél egyetlen pixelnyi függőleges leolvasási tévedés óriási eltérést okozhat az eltűnési pontok helyzetét illetően. Ezt figyelembe kell vennünk az alkalmazás végső formába öntésénél. Amennyiben az alsó és a felső oldal párhuzamos, csak egyetlen eltűnési pont jelenik meg. Ekkor a sík horizontvonalára a pontot érintve szintén párhuzamos lesz az oldalakkal.

Bár az eredmények nem függenek a fókusztávolságtól, de a számítás pontossága igen. Kis látószög, tehát nagy fókusztávolság esetén az egyenesek összetartása „gyengül”, az eltűnési pontok távolodnak egymástól, növelve az esetleges leolvasási hibát. Szerencsére a fix gyújtótávolságú automata fényképezőgépek nagy látószöget használnak, így ennek a jelenségnek nincs számottevő hatása.

A számítási módszernek leginkább megfelelő referenciaábra kiválasztása

A mobil eszközök beépített kamerája – egyelőre – kivétel nélkül gyenge vagy közepes minőségű, jelentős hordótorzítással. Mivel a nagyon távol fekvő eltűnési pontoknál nem engedhető meg számottevő leolvasási hiba, lehetőség szerint el kell kerülnünk ennek az esetnek az előfordulását. Ez úgy oldható meg, hogy – bár a számításhoz négy pont is elég lenne –, két egymáshoz képest 45 fokkal elforgatott négyzet képéből alkotjuk meg a referenciaábrát. Az irányultság meghatározásához szükséges kitétetett csúcs külön szint kell kapjon. A 4. ábra egy lehetséges megoldást mutat.

4. ábra

Referencia minta

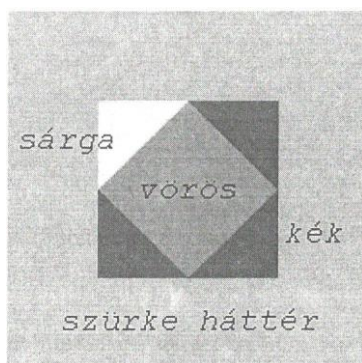


Figure 4: Reference pattern

5. ábra

Az algoritmus eredménye

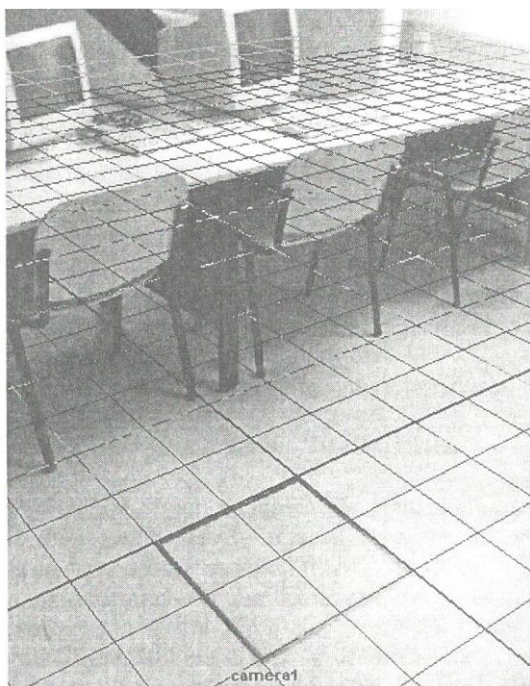


Figure 5: The result of the algorithm

6. ábra

Virtuális objektummal kiegészített tér



Figure 6: Room amplified with virtual object

A virtuális objektumok belső ábrázolása

A 3 dimenziós mobil technológia sikerének záloga a megfelelő minőségű virtuális objektumok korszerű programokkal támogatott létrehozása és az elkészült világ hordozható eszközök file-formátumába való konvertálhatósága. A 3D Studio Max program 7-es verziójában helyet kapott a .m3g formátumba való exportálás is. A .m3g a mobile 3D szabvány (más néven JSR-184 Java specifikáció) 3 dimenziós világokat tároló állományainak alapformátuma. A 3D Studio Max széles körű konvertálhatóságot biztosít a főbb CAD rendszerek felé, illetve felől, így például meglévő épülettervek is beilleszthetők az alkalmazásba.

A mobil eszközök Java alkalmazásait JAR csomagokban állítja elő a fordító. Egy ilyen csomag tartalmazza a Java kódot, illetve a program által használt összes file-t (képeket, zenét, .m3g file-okat). Az egyes helyszínek leíró .m3g file-ok vagy a csomag részét képezik, vagy futásidőben szerver-kliens kommunikációval tölthetők le a mobil eszközre. Az előbbi esetben minden egyes helyszínre újra le kell fordítani a teljes kódot, az utóbbinál egy általánosan használható programot kapunk. A felhasználó által letöltött adatmennyiség és a fejlesztő munkája egy újabb helyszín bevonása során a két módszernél nagyjából azonos, szerzői jogi szempontból az előbbi a szerencsésebb.

EREDMÉNY ÉS ÉRTÉKELES

Az 5. ábra kidolgozott algoritmus eredményét mutatja. A kép egy SonyEriccson K700i mobiltelefon beépített kamerájával készült, ezután áttöltöttem a számítógépre. Az algoritmust megvalósító program a számítógépen futott. A referenciapontok felvétele manuálisan történt. A terem padlójára rajzolt négyzet képe alapján a program kiszámolta a kamera helyzetét. Mivel a négyzetnek nincs kitüntetett csúcsa, origónak a képen legfőleg megjelenő pontokat választottam. A kapott eredményeket Maya nevű 3 dimenziós tervezőprogrammal ellenőriztem: létrehoztam egy kamerát az adott paraméterekkel, a fotót a kiszámított látószögnek megfelelően háttérképnek állítottam be. Látható, hogy a jelentős torzítású digitális fényképezőgéppel készült fotóra is viszonylag nagy pontossággal sikerült ráilleszteni a virtuális koordináta-rendszert.

A 6. ábra a valós tér egy lehetséges virtuális objektummal való kiegészítését mutatja, különböző irányokból.

A mobil eszközök megjelenő képessége nem teszi lehetővé a fenti képeknek megfelelő minőséget: a virtuális objektum kevésbé „sima”, az árnyékolás durvább. Egyelőre nincs lehetőség virtuális tárgyaknak valós időben változó mozgó képbe való beillesztésére. A mobil technológia azonban rohamosan fejlődik: bár a legelső 3D-s képességekkel felvértezett eszköz csak tavaly jelent meg, az azóta eltelt néhány hónap alatt közel tízszeresére növekedett a mobil 3D technológiát használó készülékek teljesítmény-mutatója. Mindenképpen érdemes tehát ilyen irányú fejlesztésekbe időt és energiát fektetni.

IRODALOM

- Glaskó G. (2000). Vision-based Camera Matching Using Makers. 4th Central European Seminar on Computer Graphics (CESCG 2000)
- Haralick R. M. (1988). Determining Camera Parameters from the Perspective Projection of a Rectangle. Pattern Recognition, Vol. 22, No 3, pp. 225-230.
- www.jbenchmark.com

Levelezési cím (*Corresponding author*):

Tukora Balázs

Pécsi Tudományegyetem, Pollack Mihály Műszaki Kar

Műszaki Informatika Tanszék

7624 Pécs, Rókus u. 2.

University of Pécs, Pollack Mihály Faculty of Engineering

Department of Information Technics

H-7624 Pécs, Rókus u. 2.

Tel.: 36-72-512-534, Fax: 36-72-501-534

email: tuxi@morpheus.pte.hu