



## **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 referenciátárgyak, -pontok, vagy -ábrák környezetbe helyezése jelenti. Az ezekkel szemben támasztott követelmények: a könnyű felismerhetőség, a képen szereplő tárgyaktól való biztos megkülönböztethetőség. Ezek kielégítéséhez kell megtalálnunk a megfelelő színt (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” színt 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, felismerhetőknak 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-koordináta-rendszert 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 várokoznak. 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-koordináta-rendszerhez 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 térront, illetve a vetítési síkon előállt képe a kamera gyújtópontjából kiinduló 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

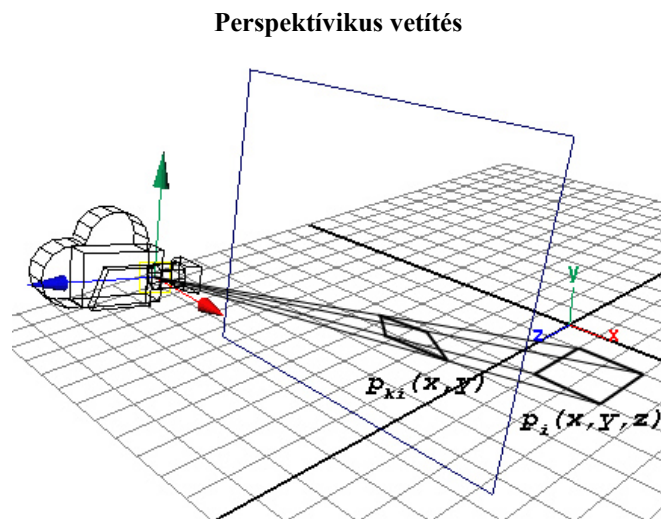


Figure 1: Perspective projection

Számításunk kiinduló értékei az 1. ábra szerint:

$$\underline{\underline{p}} = [\underline{p}_1, \underline{p}_2, \dots, \underline{p}_n] = \begin{bmatrix} p_{1x} & p_{2x} & \dots & p_{nx} \\ p_{1y} & p_{2y} & \dots & p_{ny} \\ p_{1z} & p_{2z} & \dots & p_{nz} \end{bmatrix}, \text{ vagyis a referenciapontok 3 dimenziós}$$

koordinátái a világ-koordinátarendszerben, illetve

$$\underline{\underline{p}}_k = [\underline{p}_{k1}, \underline{p}_{k2}, \dots, \underline{p}_{kn}] = \begin{bmatrix} p_{k1x} & p_{k2x} & \dots & p_{knx} \\ p_{k1y} & p_{k2y} & \dots & p_{kny} \end{bmatrix}, \text{ a referenciapontok 2 dimenziós}$$

koordinátái a képernyő-koordinátarendszerben.

A kamera helyzetét leíró paramétereket homogén transzformációs mátrix formájában keressük:

$$\underline{\underline{H}} = \left[ \begin{array}{ccc|c} \underline{\underline{R}} & \underline{\underline{k}} \\ \hline 0 & 0 & 0 & 1 \end{array} \right] = \left[ \begin{array}{ccc|c} \underline{e}_1 & \underline{e}_2 & \underline{e}_3 & \underline{k} \\ \hline 0 & 0 & 0 & 1 \end{array} \right] = \begin{bmatrix} e_{1x} & e_{2x} & e_{3x} & k_x \\ e_{1y} & e_{2y} & e_{3y} & k_y \\ e_{1z} & e_{2z} & e_{3z} & k_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ ahol}$$

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

## 2. ábra

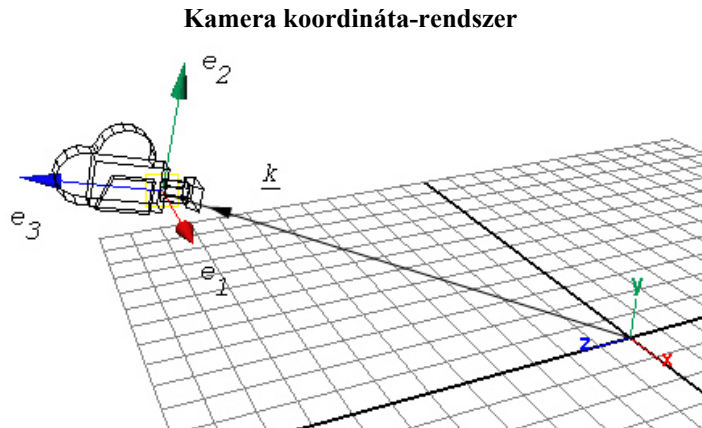


Figure 2: Camera coordinate-system