



M Ű E G Y E T E M 1 7 8 2

*Virtuális és kiterjesztett valóság
támogatása pontfelhőkkel*

Készítette: Papp Viktor

Másodéves Építőmérnök BSC hallgató

Konzulensek: Dr. Lovas Tamás

BME Fotogrammetria és Térinformatika Tanszék

Somogyi József Árpád

BME Fotogrammetria és Térinformatika Tanszék

Tartalomjegyzék

| | |
|--|----|
| 1. Bevezetés..... | 3 |
| 2. Mérések..... | 5 |
| 2.1 A földi lézerszkennelésről általában..... | 5 |
| A mérés..... | 5 |
| 2.2 Lézerszkenneres mérés..... | 7 |
| 2.3 Mélységkamerás mérés..... | 8 |
| 2.4 Képpont alapú térrekonstrukció..... | 8 |
| 3. Nyers adatok vizsgálata..... | 9 |
| 4. Modellezés..... | 11 |
| 4.1 Nyílt forráskódú programokkal történő modellalkotás..... | 11 |
| 4.2 Kereskedelmi programokkal történő modellalkotás..... | 14 |
| 4.3 Modellek értékelése..... | 15 |
| 5. VR/AR technológia..... | 17 |
| 5.1 Modellek beillesztése VR környezetbe..... | 17 |
| 5.2 AR kártya létrehozása..... | 19 |
| 5.3 A VR/AR technológia további felhasználási lehetőségei..... | 20 |
| 6. Gazdasági elemzés..... | 23 |
| 6.1 Hardver..... | 23 |
| 6.2 Szoftverek..... | 24 |
| 6.3 Felhő alapú számítástechnika - Cloud Computing..... | 25 |
| 7. Összefoglalás..... | 28 |
| Köszönetnyilvánítás..... | 30 |
| Irodalomjegyzék..... | 31 |

1. Bevezetés

A dolgozatomban az igen gyorsan fejlődő virtuális (VR) és kiterjesztett (AR) valóság alkalmazhatóságát vizsgálom az építőipar, a gyártástechnológia és az élet egyéb területein, valamint annak a lehetőségét, hogyan támogathatják a különböző módszerekkel előállított pontfelhők a technológiát.

A városi felmérésekben egyre nagyobb hangsúlyt kapnak a villanyoszlopok, világítótestek felmérései. A felmért állományok kataszteri célokra túl a világítótestek cseréjét, karbantartását is segíthetik azok állapotának felmérésével, esetleg típusának automatikus vagy félautomatikus meghatározásával. Dolgozatomban ebből kifolyólag három különböző villanyoszloppal foglalkoztam (1. ábra 1. ábra), amiket különböző módszerekkel mértem fel: lézerszkennéssel (Faro Focus 3D 120S), mélységkamerával (Kinect Xbox 360) és fényképezőgéppel (Sony Nex 7 Alpha). A mérést követően a nyers adat több okból sem bizonyult megfelelő minőségűnek. Egyrészt, mivel ezek az objektumok magasak, a felső részüket nagyon körülményes és időigényes lenne a kívánt felbontásban felmérni, így az oszlopok felső része zajos és helyenként hiányos volt (2. ábra). Másrészt, ezek a szabadban vannak, emiatt a nap is zavaró tényező a mélységképes mérésnél. A fényképezés esetén az oszlop keskeny mérete jelentett problémát.

Bár a pontfelhők alapján előállítottam az oszlopok felszínmodelljét, az a fentebb említett, nem ideális körülmények miatt a további munkámhoz nem volt megfelelő. További feldolgozással előállítottam az oszlopok vektoros modelljeit, amelyeket már jól lehet kezelni VR/AR környezetben.

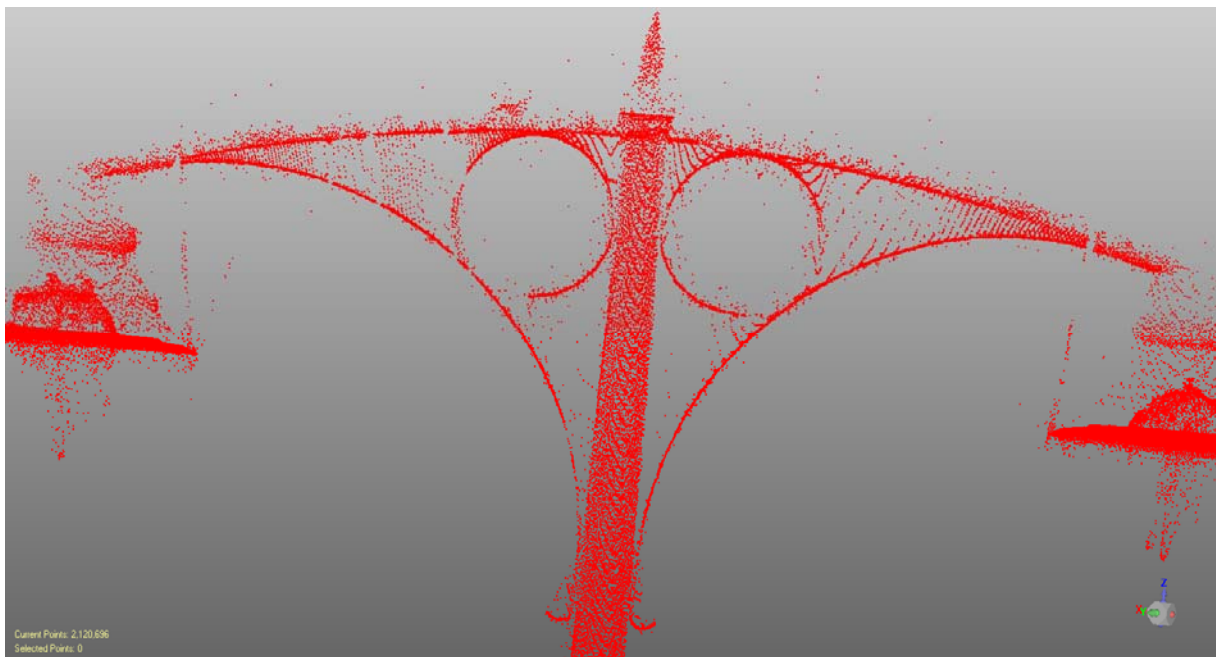
Bár a modellezés időigényes volt, sikerült létrehozni olyan modelleket, amelyek a valóságtól csak kis mértékben térnek el, kellőképpen pontosak és részletesek. A modellezéshez a villanyoszlopok nyers pontfelhőjét és a felszínmodellezéssel reprodukált képét használtam fel. Ezekhez a lépésekhez egyaránt felhasználtam kereskedelmi (Autodesk AutoCAD, Geomagic Studio) és nyílt forráskódú (Blender, Cloud Compare) programokat.

Az elkészített modellek már beilleszthetők egy VR környezetbe, valamint azok alapján elkészíthető az úgynevezett AR kártya, mely mobil alkalmazás segítségével látványos 3D megjelenítést tesz lehetővé.

Végül gazdasági szempontok figyelembevételével következtetéseket vontam le a pontfelhők VR/AR alkalmazhatóságáról.



1. ábra: A modellezés céljából kiválasztott villanyoszlopok



2. ábra: A villanyoszlop pontfelhőjének felső része

2. Mérések

2.1 A földi lézerszkennelésről általában

Az objektumok szkennelésénél földi lézerszkennert alkalmaztam, ami viszonylag új távérzékelési módszernek tekinthető, széles körű alkalmazása az 1990-es évek végére tehető. Működési elvét tekintve aktív távérzékelési technológiák közé sorolható, a műszerek az általuk kibocsátott lézersugarat alkalmazzák a távolságok meghatározása során. A külföldi szakirodalomban terrestrial laser scanning-nek hívják ezt a technológiát, innen származik a TLS mozaikszó rövidítés.

Attól függően, hogy a lézerszkennert milyen módon méri meg a műszer-tárgy távolságot, megkülönböztetünk időméréses (ToF – Time of Flight) és fázisméréses (PB – Phase Based) műszereket. Előbbi előnye a nagyobb mérési távolság, míg utóbbi esetében nagyobb mérési pontossággal számolhatunk. A technológia fejlődésével megjelent a hullámforma/hullámalak (full waveform) rögzítésének lehetősége is.

Annak ellenére, hogy a földi lézerszkennelés kutatási szinten is csupán pár évtizedes múltra tekint vissza, a TLS három generációját különböztetjük meg. A lényegi áttörés 2005-ben következett be, a harmadik generációs műszerekkel (manapság csak ilyenek vannak a piacon, a méréseket is ilyenekkel végeztük). A nagy különbséget az előző generációkhoz képest a felmérésre fordított idő lecsökkentése, a lézersugár kibocsátási frekvenciájának növekedése, a beépített adattárolás és a könnyű, hordozható akkumulátor jelenti. Egyes szkennerek már integrált kamerával készülnek és akár GNSS vevővel is rendelkeznek, amely az abszolút helymeghatározást segíti [1].

A mérés

A földi lézerszkennelésnél általában a következő részfeladatokat szükséges elvégezni:

- méréstervezés,
- mérés,
- utófeldolgozás,
- végtermék előállítás.

Méréstervezés

A földi lézerszkenneléssel végzett felmérések előtt általában méréstervezést végzünk. Ebben a lépésben meghatározzuk a felmérés várható időigényét, az emberi erőforrás-szükségletet, valamint az álláspontok, és - amennyiben szükségesek - a kapcsoló objektumok helyeit is.

Mérés

A felmérések során ellenőrizzük, hogy a megtervezett helyeken lehetséges-e a mérés, valamint van-e valamilyen előre nem várt zavaró körülmény. A szkennert olyan pontokban állítjuk fel, hogy a felméréendő objektum teljes egészéről megfelelő pontsűrűségű adat álljon elő, valamint az álláspontok egy rendszerbe történő illesztése is megoldható legyen. A mérés során fontos feladat a manuálé készítése, mely a feldolgozás során, az álláspontok illesztésénél, a kapcsoló- és illesztőpontok azonosításában nyújt fontos segítséget.

Utófeldolgozás

A földi lézerszkenneléssel nyert pontfelhő helyszínen történő feldolgozása az ahhoz szükséges számítási erőforrás és időigény miatt nem lehetséges. Az utófeldolgozás lépései:

- álláspontok egy rendszerbe történő illesztése,
- terület lehatárolása és a pontfelhő igény szerinti újramintavételezése,
- zajszűrés és tisztítás,
- végtermékek előállítása.

Az álláspontok illesztését különböző módszerekkel lehet megtenni. Amennyiben a mérés során dedikált illesztő objektumokat helyeztünk el, azok detektálásával, azonos pontok alapján határozzuk meg a transzformációs paramétereket. Azok hiányában a valóságban fellelhető geometriai alakzatokat is alkalmazhatunk ilyen célokra (például fal síkja, lámpabúra, ablak sarokpontja). A gyakorlatban azonban az iteratív legközelebbi pontok módszere (Iterative Closest Point - ICP) terjedt el, amely egy jó kiinduló állapot és a pontfelhők kellő átfedése mellett automatikusan határozza meg a szükséges transzformációs paramétereket.

A feldolgozás következő lépése a terület lehatárolása és az újramintavételezés, amelyekre azért van szükség, hogy az adatokat kezelhető méretűvé alakítsuk.

A felméréskor óhatatlanul is felmérünk olyan pontokat, amelyek nem szükségesek a feldolgozás, kiértékelés szempontjából (például elhaladó autók, gyalogosok). A nagyobb porszemcsék, esőcseppek szintén zavarhatják a mérést, ezeket el kell távolítani a pontfelhőből. Erre több módszer áll rendelkezésre, a legkézenfekvőbb a manuális tisztítás, ekkor a

feldolgozást végző személy vizuálisan dönti el, hogy az adott pont vagy ponthalmaz részét képezi-e a vizsgált tárgynak (az általunk végzett méréskor ezt a módszert részesítettem előnyben, mert sok olyan pont volt, amit szemrevételezéssel jól el lehetett határolni a villanyoszloptól). Természetesen vannak automatikus módszerek is [1]:

- Nyolcasfa (octree)
- Izolált pontok szűrése
- 2,5D raszter
- Távolság határ (range gate)
- Intenzitás intervallum (intensity gate)
- Pontszűrő (point filter)

Gyakran nehéz élesen elhatárolni az utófeldolgozást és a végtermék előállítását, mert bizonyos alkalmazások esetén a pontok szűrésével és tisztításával előállt pontfelhőre már tekinthetünk végeredményként. Viszont ezután még felszínmodellt vagy vektoros modellt is alkothatunk, amely az objektum valós felületét közelíti.

2.2 Lézerszkenneres mérés

Az általam végzett méréshez Faro Focus 3D 120 S lézerszkennert alkalmaztam (3. ábra: Faro Focus 3D 120 S), ami egy fázisméréses (PB) lézerszkennere, hatótávolsága 0,6-70 m. Mint minden lézerszkennere (PB szkennerekre fokozottabban), erre is igaz, hogy rövid idő alatt nagy mennyiségű pontot tud rögzíteni (maximálisan 976 000 pont másodpercenként). A beépített kamerának köszönhetően a kapott pontfelhőhöz színadatok rendelhetők [2].



3. ábra: Faro Focus 3D 120 S

A feldolgozást nehezítette az a tény, hogy a mérés szabadban, nyílt területen történt, így sok felesleges pont keletkezett, amelyet később el kellett távolítanom a felhőből (bicikli, tábla, stb.), valamint az oszlopok magassága miatt a pontfelhő felső része hibás és hiányos lett, mindemellett nagyon zajos is volt. Ezeknek az orvoslására később kitérek.

2.3 Mélységkamerás mérés

A lézerszkennerekkel ellentétben a mélységkamerák (Flash LiDAR, Depth Camera) a pillanatnyi látómezőről készítenek egy távolságképet. A kép elkészítéséhez jellemzően infravörös fényimpulzust bocsájtanak ki, melynek visszavert képét rögzítik. A távolságadatok mérésére jellemzően két megoldás terjedt el: az időméréses (pl. Kinect Xbox One, 4. ábra) és az előmetszéses (pl. Kinect Xbox 360, 4. ábra) [3]. Az oszlopokról Kinect Xbox 360-al gyűjtöttem adatokat, ennek mérési pontossága az adott körülmények közt elmaradt a földi lézerszkennertől, valamint, mivel infravörös fényimpulzust használ, így érzékeny a napfényre. Az eszköz sikerének kulcsa a tömeggyártás, mivel az ára így alacsonyan tartható.



Kinect Xbox One



Kinect Xbox 360

4. ábra Jelenleg elérhető alacsony árú mélységkamerák

Felépítését tekintve két CMOS kamerából (1280×1024 px, de csak 640×480 px érhető el), egy infravörös sugárzóból, sztereó mikrofonból és egy gyorsulásmérőből áll.

2.4 Képpont alapú térrekonstrukció

A fotogrammetriában utóbbi években elterjedt megoldás a képpont alapú térrekonstrukció, mely lehetővé teszi, hogy felvételsorozatokból pontfelhőket, illetve akár textúrázott felületmodelleket alkossunk. A művelet több lépésben valósul meg. Először a felvételek jellemző pontjait kell kinyerni, majd azokat egymáshoz párosítani. Ezek alapján a kamerák helyzete kiszámítható, valamint egy ritka pontfelhő előállítható. A gyakorlatban az tapasztalható, hogy az ily módon előállított felhő további modellezésre nem alkalmas, így el kell végezni annak besűrítését. Ezen lépés után előáll egy olyan sűrű pontfelhő, amely segítségével a felmért objektumról felületmodellek is készíthetők.

3. Nyers adatok vizsgálata

Az általam használt szkener méréséből kapott nyers pontfelhőket a helyszínen nem lehet pontosan értékelni. A műszer kijelzője túl kicsi felbontású ahhoz, hogy azon jól lássuk a pontfelhőt, annak hibáit. Amennyiben az újramérés gondolata felmerül, célszerű a mérést megismételni. Bár a lézerszkenerhez lehet csatlakoztatni laptopot, a kiküldött kép felbontása korlátozott.

A szkener feldolgozó szoftverében előzetesen már feldolgozott (illesztett, lehatárolt) pontfelhőt először két ingyenesen letölthető, nyílt forráskódú (open source - OS) programmal vizsgáltam meg, a CloudCompare-rel és a Meshlabbal (5. ábra). Ezekből látszott, hogy a pontfelhő alsó része közel hibátlan, míg a felső része hibás (zajos és hiányos) volt. Ezért utófeldolgozással próbáltam javítani az adatokat az előbb említett programokkal, de a végeredmény nem lett ideális, ezért további javításokra volt szükség.

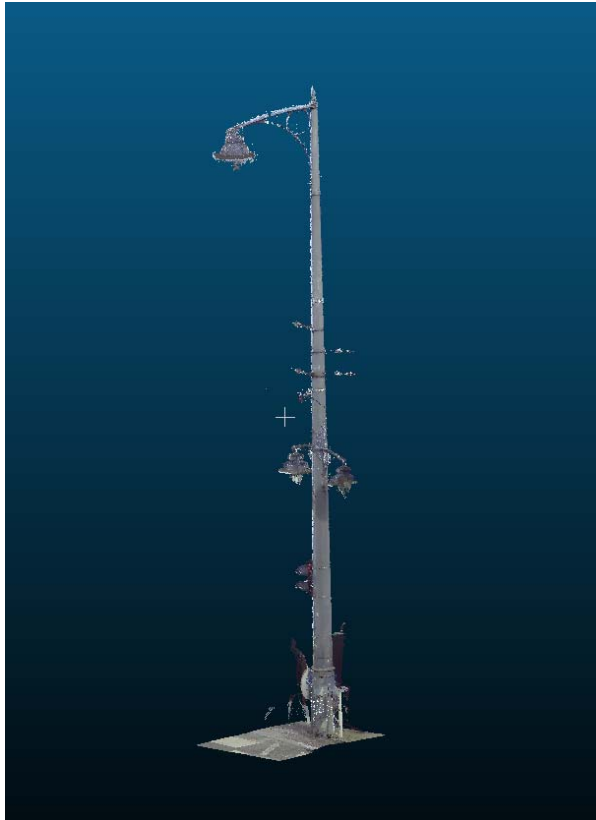
Kereskedelmi programokkal is megvizsgáltam a pontfelhőket; Geomagic Studioban is ugyanazt a következtetést kellett levonnom, mint az előzőekben, viszont a helyreállítás ebben a környezetben sokkal kényelmesebb volt számomra, köszönhetően a felhasználóbarát felületnek.

A pontfelhők utófeldolgozása több lépésből áll (amiket az előző fejezetben már említettem), mivel ezek közül az első kettő már megvalósult, így a továbbiakkal foglalkoztam. Szűrésnél az izolált pontok módszerét és a távolság határ módszert alkalmaztam, de ezeken kívül még kézzel történő tisztításra is szükség volt, mivel az oszlop mellett voltak egyéb, akaratlanul felmért objektumok is (pl. reklámtábla, kerékpár, növény), ezeket lasszó- vagy négyzetkijelölővel könnyedén kijelöltem és eltávolítottam. A felmérés során a szkenneléssel egybekötve fényképezés is történt, így színes pontfelhő ált rendelkezésemre, ami megkönnyítette az irreleváns pontok eltávolítását.

A végtermék előállításához elkészítettem a villanyoszlopok felszínmodelljét, ezeken még kisebb-nagyobb lyukak voltak megfigyelhetők. Ezeket a programba épített modulokkal manuálisan és automatikusan tudtam javítani. Előbbi precízebb megvalósítást tesz lehetővé, de lassú és időigényes, még egy ilyen villanyoszlop esetén is több száz befoltozandó lyukról beszélhetünk.

Az elkészített modellek közül néhány megfelelőnek bizonyult a további felhasználáshoz (ezek a lézerszkennelvel végzett mérésből származó pontfelhők voltak), viszont voltak olyanok,

amelyek túlságosan hibásnak bizonyultak (ezek pedig a mélységkamerás mérésből származtak, mivel a Kinect kamerája a közvetlen napsütés miatt nem tudott jó minőségű pontfelhőt előállítani). Szükségessé vált tehát, hogy elkészítsem a villanyoszlopok vektoros modelljét, amik helyenként eltérnek a valóságtól, viszont így van egy idealisztikus modell, ami közelítően megfelel a valóságnak.



CloudCompare



Meshlab

5. ábra Pontfelhők nyílt forráskódú szoftverekben

4. Modellezés

Az előző fejezetben megnéztük, hogy a nyers pontfelhőkből hogyan állítottam elő a felszínmodelljüket, amik egyes mérnöki feladatokra (főleg vizualizációs) célokra alkalmasak lehetnek, azonban sok más mérnöki feladathoz szükség van vektoros modellekre. A vektoros modelleknek számos előnyük van, pl. könnyebb szerkeszteni, javítani, méretezni, kevesebb helyet foglalhat (az objektum komplexitásától függ), jobban lehet használni animációkhoz, prezentációkhoz, valamint a vektoros modelleken pontosabban lehet méréseket végezni [4].

Említettem, hogy a vektoros modelleket jobban lehet használni prezentációkhoz. Ennek az az oka, hogy ezeket könnyebben lehet színezni, textúrázni, fényképet „ráhúzni” és animálni, renderelni (a renderelés az a folyamat, amikor egy 2D-s vagy 3D-s modellből képet készítünk számítógépes algoritmusok segítségével [5]).

A következőkben szó lesz arról, hogyan készítettem el a pontfelhőn mért méretek segítségével a villanyoszlopok vektoros modelljét, valamint arról, ezeket hogyan helyeztem el virtuális környezetben. Ezen műveletekhez egyaránt felhasználtam nyílt forráskódú (OS – open source), ingyenesen letölthető és kereskedelmi programokat is.

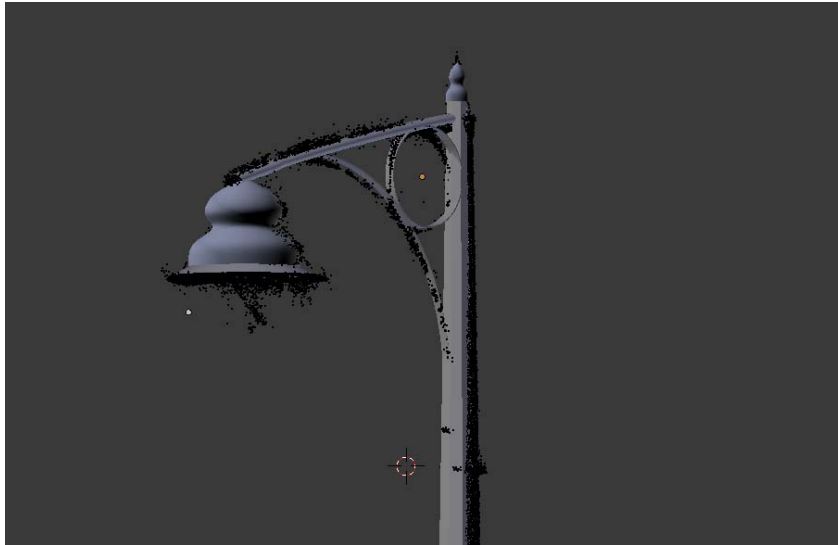
4.1 Nyílt forráskódú programokkal történő modellalkotás

A nyílt forráskódú (OS) programok lényege, hogy bárki hozzáférhet a program forráskódjához, optimalizálhatja, alakíthatja, valamint kiegészítheti azt, amelyeket bizonyos megszorítások mellett megoszthat másokkal is. Ezzel a program fejlesztése dinamikus, a felhasználók közvetlenül beleszólhatnak a fejlesztésbe.

Az általam használt OS programok közül ki kell emelnem a Blender programot, mivel ezzel dolgoztam a legtöbbet és ez bizonyult a legjobbnak a munkához. A Blender nem csak modellalkotásra alkalmas, hanem más vizuális effektek, animációk, akár egyszerűbb fizikai szimulációk készítésére is.

A modellek elkészítéséhez támpontot a pontfelhő és az azokból elkészített felületmodell adott (6. ábra). Azok alapján igen jó, valóság-hű modelleket sikerült elkészíteni. A pontfelhők importálása nem ment egy lépésben, mivel azok kiterjesztését (.xyz) a Blender nem tudta értelmezni, így előtte egy másik programmal (pl. az előzőekben említett CloudCompare-rel vagy MeshLabbal) át kellett konvertálni őket object (.obj) vagy stanford (.ply) kiterjesztésre, ezeket a program már tudta kezelni. Természetesen a generalizálás következtében a modell

helyenként eltér a valóságtól, pl. a virágtartó fémgyűrű a valóságban már meghajolt, ferde, míg a modellelmen egyenesként szerepel.



6. ábra: Blenderben készített modell a pontfelhő alapján

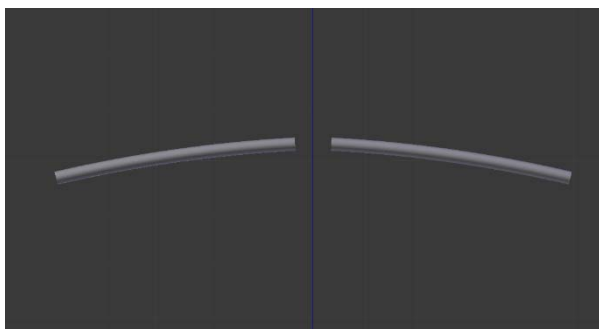
A modellalkotás folyamatát egy konkrét példán mutatom be:

- Indításkor a program mindig egy előre paraméterezett környezettel indul el. A villanyoszlopok geometriai alakja nyolcszögalapú csonka gúla. Ennek elkészítéséhez kiindulásként egy hasábot hoztam létre, amelynél megadható az oldalak száma és a belső (vagy külső) átmérő nagysága, valamint a magasság. A következő lépésben a hasábot gúlavá alakítottam, ehhez beépített transzformációt használtam. Így elkészült az oszlop (7. ábra).



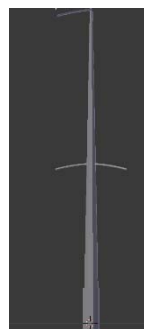
7. ábra: Az oszlop törzse (elforgatva)

- Ezután elkészítettem az oszlop három karját (8. ábra), amelyek a lámpaburákat tartják. Ezeknek a létrehozásához hengermodellt használtam, viszont itt több transzformáció is szükséges volt: kicsinyítés, nyújtás, moztatás, valamint az ívekhez deformálni kellett a testet. Mindezt szerencsére csak egyszer kell megcsinálni, ugyanis utána az első lemásolásával és áthelyezésével nem kell előlről kezdeni a műveletet.



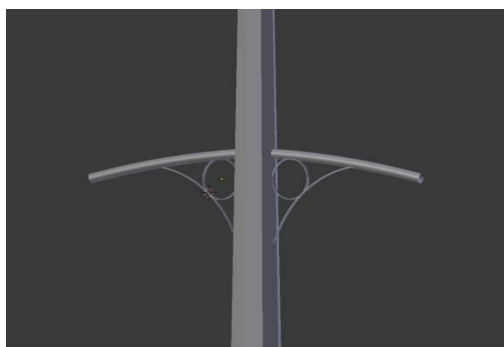
A lámpatestekkarjai külön

8. ábra



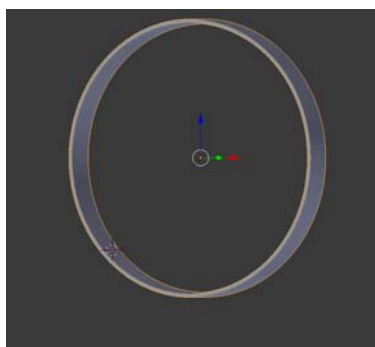
A karjok a törzsre helyezve

- A villanyoszlop karjai alatt vékonylemezből van egy díszítőelem. Ezeket a legegyszerűbb úgy elkészíteni, hogy létrehozunk egy vastagsággal és felső-alsó lappal nem rendelkező hengert, aminek aztán transzformálással adunk vastagságot és (a karokkal megegyező) magasságot. Ezzel megkapjuk a teljes elemet (9. ábra). A nagyobb sugarú körhöz (ívhez) lemásoljuk az előbb elkészített objektumot és felnagyítjuk.



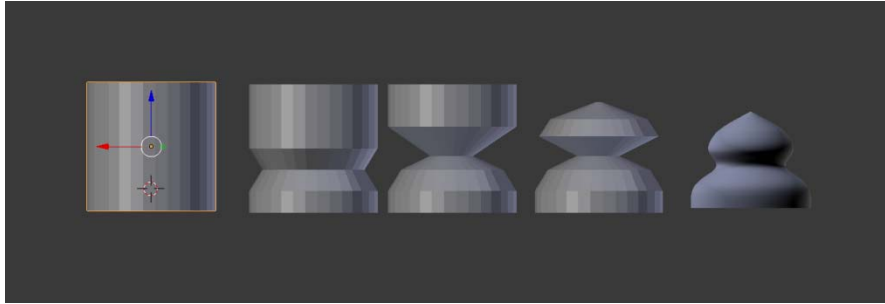
A karika és a vékonylemez az oszlopra helyezve

9. ábra



A díszítőkarika külön

- Az utolsó előtti lépés a bűrák elkészítése. Az elem összetett alakjának következtében ez egy bonyolult feladat, ugyanis a beépített hengertestet kézzel kell kialakítani, nem lehet pár egyszerű transzformálással előállítani. Az egész formázást szerkesztő módban végeztem; először felvágtam a testet, így új pontokat és oldalakat kaptam. Minél több szeletre vágjuk a hengert, annál pontosabban tudjuk formázni, viszont ezzel együtt annál időigényesebb is a művelet (10. ábra). Ezután még az aljára helyeztem egy alapfelület nélküli kúpot.



10. ábra: A lámpaburák készítési folyamata

- Utolsó lépésként egybeillesztettem az elkészített testeket, és elkészült a villanyoszlop. (11. ábra)



11. ábra: A kész modell

4.2 Kereskedelmi programokkal történő modellalkotás

Több olyan kereskedelmi program elérhető, amelyekkel végre lehet hajtani egy modellezési feladatot (pl. Autodesk AutoCAD, Rhinoceros, SolidWorks). Ezen termékek közül munkám során az AutoCAD-del foglalkoztam.

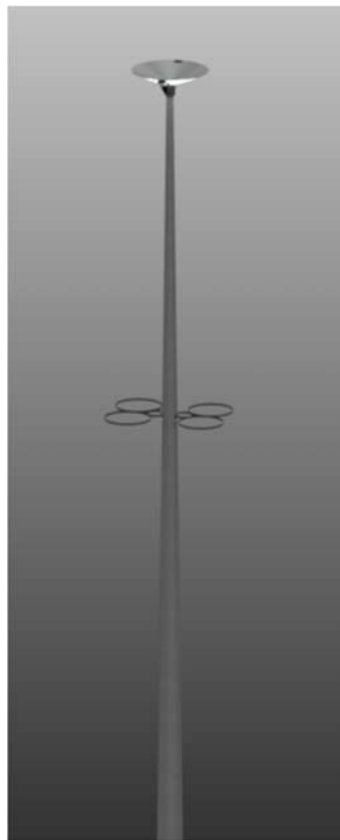
A modellezést hasonló módon végeztem, mint az előző fejezetben leírtam, egyszerű, beépített testekből indultam ki, azokat transzformáltam és alakítottam teljes modellé. Ebben az esetben is felhasználtam a mérési eredményekből a pontfelhőket, így a modellek a valóságnak egy jó közelítései lettek. A pontfelhőket ebben az esetben Blenderből vittem át, mivel így ki tudtam próbálni, ami az egyik nagy előnye az OS programoknak; átjárhatóság számos környezet

között. Alapból a Blender nem rendelkezik olyan funkcióval, hogy az AutoCAD által használt kiterjesztésben (.dxf) exportáljon ki objektumokat, így le kellett hozzá tölteni egy modult (másnéven plugint), ami ezt lehetővé teszi, így megoldódott a két program közötti átjárhatóság, át tudtam vinni a pontfelhőket (de akár a már Blenderben elkészített modellek is átvihetők).

Az AutoCAD egy hasznos funkciója a metszet készítés, amit gyakorta használnak az építőiparban. A Blenderből importált modellekről így akár készíthetünk metszeteket is, amiken jelölhetjük a gépészeti vezetékeket vagy a gyártáshoz adhatunk meg hasznos információkat, méreteket.

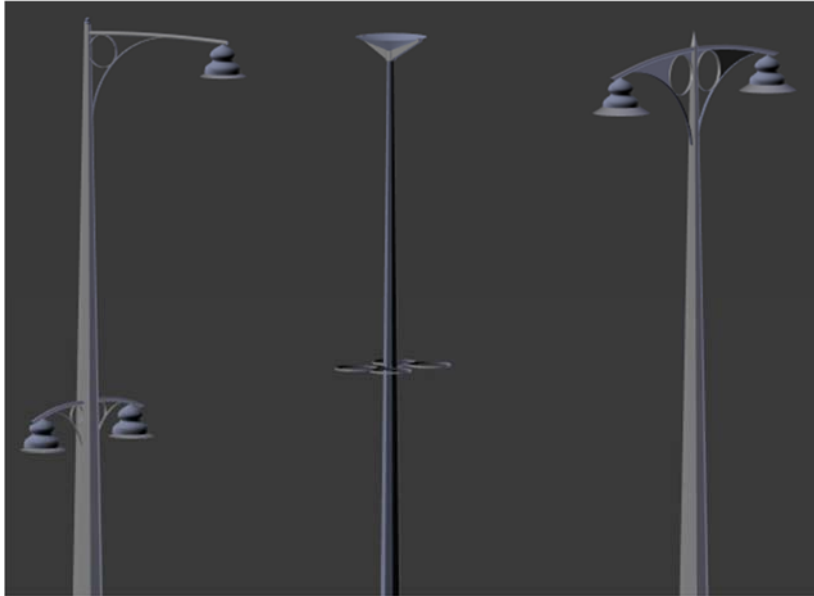
4.3 Modellek értékelése

A modellek (13. ábra) elkészítése gyorsan ment és ezzel sok időt spóroltam meg, mivel a pontfelhők (már amelyiket lehetett) javítása úgy, hogy azok teljesek (hiánytól és egyéb hibáktól mentesek) legyenek, nagyon időigényes és fárasztó munka lett volna. Persze az elkészített modellek csak közelítik a valóságot, helyenként eltérnek tőle (pl. a 12. ábra látható villanyoszlop virágtartó karikája a valóságban nem egyenes, mivel az idő folyamán lehajlott kicsit), de annak egy nagyon jó mása lett.



12. ábra: Villanyoszlop modell AutoCAD környezetben

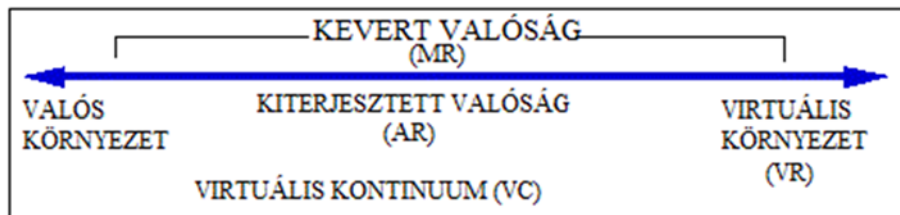
A Blenderben elkészített villanyoszlopok jól kezelhetőek más programokkal, mivel könnyedén lehet őket exportálni más formátumokba is (pl.: .ply, .obj, .stl, .dxf), így ezek a modellek kiváló alapként szolgálnak további munkákhoz (pl. a már említett metszetek elkészítéséhez AutoCAD-ben vagy VR/AR alkalmazáshoz, amit a következő fejezetben ismertetek).



13. ábra: A három villanyoszlop modellje Blender környezetben

5. VR/AR technológia

A Virtuális Valóság (VR – Virtual Reality) egy olyan mesterségesen létrehozott környezet, amiben a megfigyelő interakcióba is léphet környezetével, alakíthatja azt. A VR nem feltétlenül másolja le teljesen a fizikai világot, átalakíthatja azt (pl. létezhet a fizika törvényei nélkül is). Ezt a két világot, a virtuális- és a valós világot foglalja magába a Virtuális Kontinuum valamint a kettő keverékéből születik a Kevert Valóság (MR – Mixed Reality). A két világ között helyezkedik el a Kiterjesztett Valóság (AR – Augmented Reality), ami tartalmaz objektumokat a valós világból és behelyezi azt egy virtuális környezetbe (14. ábra). [6]



14. ábra: Virtuális Kontinuum ábrája [6]

Napjainkban egyre több okostelefon és tablet képes a VR tartalom megjelenítésére, továbbá különböző VR headsetek is elérhetőek, ilyen pl. a HTC vive és az Oculus Rift (15. ábra).



HTC Vive



Oculus Rift

15. ábra

5.1 Modellek beillesztése VR környezetbe

A Blenderben és AutoCADben elkészített modellek alkalmasak arra, hogy azokat beillesszük VR környezetbe.

Régen erre a legkézenfekvőbb és legszemléletesebb mód a modellek Google Earth-be történő beillesztése volt, amit akár a Sketchup programban [7] is elkészíthettük, ám ez már nem opció, ugyanis a Google nem engedi meg a modellek feltöltését magánfelhasználóknak (ennek az oka valószínűleg az, hogy nagyon sok volt a felhasználók által feltöltött rossz minőségű modell).

Emiatt egy másik módszert kellett keresnem. Egy virtuális környezetet Blenderben is létre lehet hozni további modellezéssel és textúrázással. Viszont (Weszelovits Gergő MSc-s hallgató segítségével) sikerült egy jobb, látványosabb megoldást kitalálni, melynek megvalósításához a Unity programot használtuk.

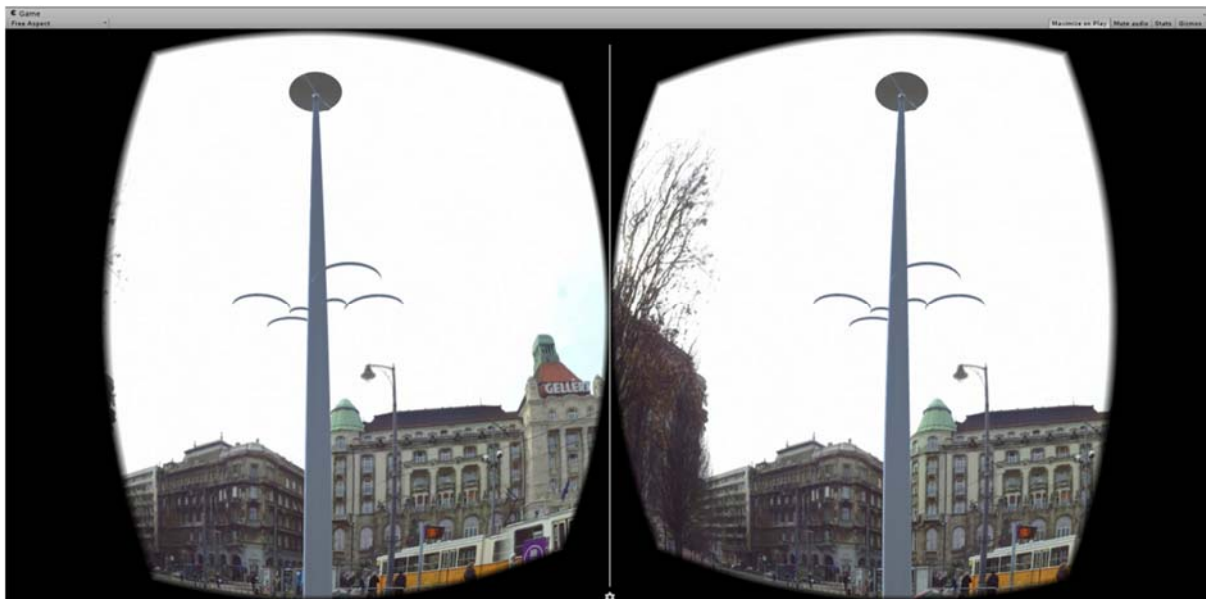
A Unity egy multiplatform motor, amit főleg telefonos applikációk és játékok létrehozásához használnak fejlesztők, de felhasználóbarát grafikus felülete révén amatőr felhasználók által is használható különféle appok és játékok készítésére. Ez az alkalmazás mindaddig teljesen ingyenes, amíg nonprofit célokra használjuk őket.

Unity felhasználásával létrehoztunk egy VR környezetet (16. ábra), amiben egy headsettel szétnézhetünk. Az elkészített alkalmazást felhasználhatnánk Oculus Rifthez, HTC Vive-hoz vagy egyéb hasonló eszközökhöz, mi most egy telefonos applikációt készítettünk (17. ábra).

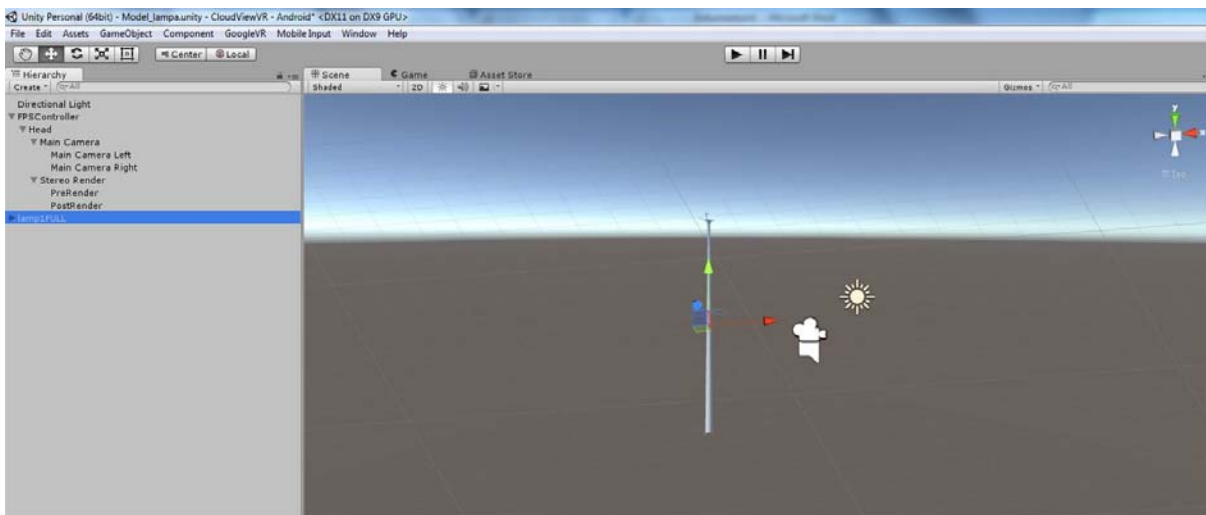
A virtuális tér előállításához a szkennerből kinyert panorámaképet állítottam be háttérként. Ehhez a képet betöltöttem Unitybe, majd Cubemap textúrává alakítottam át (ez a Unity-ben egy beépített eszköz, ami a textúrát gömbbé formálja, így a VR headsettel úgy látjuk, mintha egy valós, végtelen térben állnánk, de valójában csak egy gömbben vagyunk). Ezt alkalmaztam a meglévő virtuális tér háttérére, így már a panorámaképben nézhetünk körbe.

Ahhoz, hogy Androidos applikáció formájában álljon elő, a Unity jelenet struktúráját Android rendszerre alakítottam át. Ezután az Asset store-ból (ami a Unity által létrehozott oldal, ahol a fejlesztők megvásárolhatnak különféle, előre elkészített könyvtárakat, jeleneteket [8]) letöltött androidos modulokat alkalmaztam, mint pl. a GVRMain assetet. Ebben rendelkezésre áll egy sztereo kamera és renderer, a kamera térben történő elhelyezése után külön bal és jobb képre rendereltként látható a gömbpanoráma (16. ábra).

Ezután az .obj kiterjesztésű modellt textúráztam, melyet úgy helyeztem el a virtuális térben, hogy a GVRMain kamerája előtt legyen. A fényeket ezután hozzárendeltem az objektumhoz, illetve a jelenet fő szírt fényét is beállítottam, a modell jó megvilágítása érdekében (17. ábra).



16. ábra: Villanyoszlop a virtuális valóságban



17. ábra: A villanyoszlop modellje Unity környezetben

5.2 AR kártya létrehozása

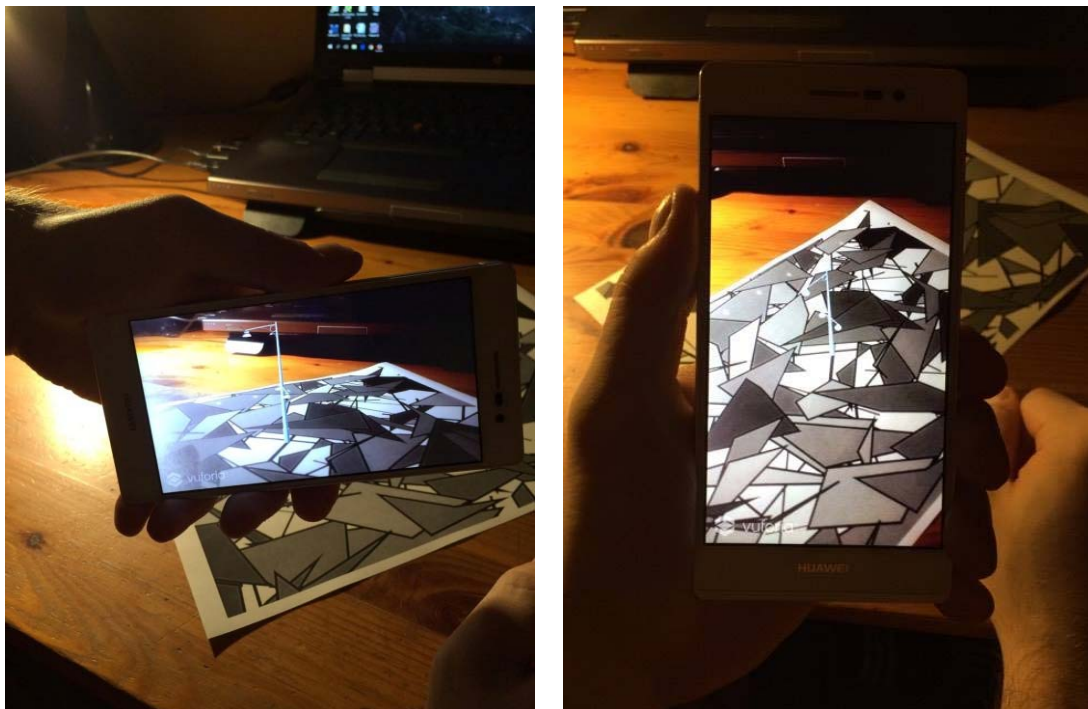
Az AR kártya egy olyan kártya, amin egy referencia kép van, amelye ha egy megfelelő alkalmazás segítségével ráfókuszáljuk a telefonunk kameráját (vagy más eszközünk kameráját), akkor a kijelzőn azt tapasztaljuk, hogy a kép fölött „lebeg” a tárgy (18. ábra), amit előzetesen odahelyeztünk. Ez jól szemlélteti a kiterjesztett valóság fogalmát is, ugyanis a kijelzőn bár a környezet valós, a tárgy nem.

Ennek a kártyának a létrehozásához a Unity és a Vuforia szolgáltatását vettem igénybe. A Unity-t az előző fejezetben már említettem; a Vuforia felel az alkalmazás AR részéért, segítségével tudjuk létrehozni a referenciaképet és az eszközöket (a Vuforia oldaláról letöltött

Assets könyvtár tartalmaz minden, az app elkészítéséhez nélkülözhetetlen fájlt, pl. az AR kamerát), hogy megvalósíthassuk a kártyát. Hasonlóan a Unity-hoz a Vuforia is mindaddig ingyenesen igénybe vehető, míg azt nonprofit célra használjuk.

A kártya elkészítésének lépései:

- Első lépésben letöltöttem az oldalról a Vuforia SDK-t, ahol felkínálta a Unity-be való integrálást.
- Következő lépésként létrehoztam egy új „jelenetet”, amelybe elhelyeztem egy AR kamerát.
- Ezt követően egy általam választott képet (ami a referencia kép lesz) feltöltöttem a Vuforia oldalára, ami készített belőle egy adatbázist, amelyet Unity-be importáltam.
- Végül a képet beillesztve arra van szükség, hogy egy tárgyat is elhelyezzünk, ami lényegében bármi lehet. Én a már elkészített villanyoszlopokat töltöttem be. Az elkészült AR alkalmazást egy Android alapú mobiltelefonra helyeztem át. Ezt Unity-n belül lehetett megtenni. [9]



18. ábra: Az elkészített AR program működése mobiltelefonon

5.3 A VR/AR technológia további felhasználási lehetőségei

Az előzőekben leírt VR headset és AR kártya felhasználása sokrétű lehet, de egyelőre ezeket főleg vizualizációs célra tudjuk felhasználni. Elképzelhető, hogy néhány év múlva egy építési projekt keretében az építészek és mérnökök számára lehetővé válik, hogy ArchiCAD,

AutoCAD, vagy Revit felhasználásával készült modelleket VR környezetben mutassák be a beruházók számára, vagy kommunikációs felületként használva ennek segítségével egyeztessenek az egyes szakágak. Az AR kártyára akár egy elmetszett modell is tehető, így a kártya forgatásakor láthatóvá válik az objektum külső fele, de akár a belső felépítése is, pl. a villanyoszlopban futó vezetékek és az oszlop anyagvastagsága.

A nagy költségvetésű filmeknél (és videojátékoknál) a munkafolyamatnak manapság alapvető részét képezi a térbeli adatgyűjtés folyamata és egyre többször használnak VR/AR eszközöket az egyes jelenetek elkészítéséhez. Erre a feladatra egy forgatáson külön csapat van, akik a VFX (Visual Effects) és CGI (Computer-Generated Imagery – Számítógép által létrehozott kép) részért felelnek. A produkcióban részt vevő összes helyszínt, színészt és kelléket szkennelik, majd ezt utómunkálatokkal tökéletesítik és készítik el a vizuális effektusokat az egyes jelenetekhez [10]. Egy film készítésénél nagyon nagy csapat dolgozik azon, hogy elkészüljenek az animációk és egyéb FX-es elemek. Egy ilyen munka nagyon sok részfeladatból áll, leegyszerűsítve ezek a következők [11]:

- Previzualizáció,
- Szkennelés,
- Tervezés (Layout TD – technical director),
- Modellezés,
- Textúrázás,
- Animálás,
- Vizuális effektusok hozzáadása,
- Renderelés (végtermék elkészítése),
- Elhelyezés.

Manapság, míg az utófeldolgozásra fordított idő jelentősen lecsökkent, addig az effektek minősége nőtt. Ennek köszönhetően például egyre többször nem élő állatokat használnak fel egy-egy jelenet forgatásához, hanem számítógéppel „keltik életre” azokat (19. ábra).



19. ábra: Számítógéppel készített virtuális medve A visszatérő (The Revenant) c. filmből

6. Gazdasági elemzés

Gazdasági elemzésem során megvizsgáltam, hogy az általam elvégzett munkához milyen hardver és szoftver szükséges, ezeknek milyen tartományban mozog az ára 2016-ban. Mindemellett kitérek arra is, hogy megérné-e befektetni felhőszolgáltatásba a feldolgozás során.

6.1 Hardver

Az előzőekben leírt munkafolyamatok elvégzéséhez kellőképpen erős számítógépre van szükség. Több szempont alapján is lehet keresni gépeket, amiket mindenképp figyelembe kell venni [12]:

1. Processzor (CPU - Central Processing Unit)

A processzorok esetében döntően két tulajdonságot kell figyelembe venni

- a. Az órajel alapján következtetni lehet a processzor teljesítményére.
- b. A magok és a szálak száma továbbá azt befolyásolja, hogy a CPU mennyi műveletet képes elvégezni párhuzamosan.

Azt, hogy melyik tulajdonságot érdemes előnyben részesíteni vásárláskor az dönti el, hogy az adott program, amihez használni akarjuk, mennyire használja ki azt, hogy több magja van egy CPU-nak. CAD környezetek esetében a szoftverek döntő többsége ún. egyszálú, csak egy magot használ [13]. Ezért érdemes nagy órajelű processzort választani.

2. Videokártya (GPU - Graphics Processing Unit)

A videokártya kezeli a 2D-s és 3D-s modellek megjelenítését a képernyőn. AutoCAD és más tervező, modellező szoftverek esetében a videokártyát általában a 3D-s modellek veszik igénybe, szóval, ha csak 2D-s tervezéshez akarjuk használni a számítógépet nem érdemes drága videokártyát venni. A 3D modellezéshez egy jó választás lehet az NVIDIA Quadro kártya család, ami kifejezetten ilyen jellegű munkához készült. Azonban számos feladatra a szélesebb körben elterjedt, olcsóbb GeForce kártyák is megfelelő választások lehetnek. Továbbá lehetőség van több kártya együttes használatára is például: Nvidia SLI (Scalable Link Interface).

3. Memória (RAM - Random Acces Memory)

A dolgozatban említett modellezési feladatok végzéséhez ajánlott 16 GB RAM-mal rendelkezni. Az AutoCAD minimum követelménye 8GB, de az optimális használathoz 16GB vagy több javasolt. Nagyobb munkákhoz és projektekhez

viszont ettől jóval több, 64-128 GB is szükséges lehet. A számítógépek memóriája megfelelő alaplap választása mellett egyszerűen bővíthető.

4. Háttértároló

Jelenleg két kézenfekvő változat érhető el: HDD és SSD

- a. A HDD 1956-ban jelent meg az IBM gondozásában. Az adatokat mágnesezhető réteggel bevont, forgó lemezekon tárolja. Jelenleg tárolókapacitást illetően lényegesen olcsóbb megoldás, mint az SSD.
- b. Az SSD nem tartalmaz mozgó alkatrészt. Emellett nagy előnye a gyors adatátviteli sebesség, alacsony elérési idő és energia felhasználás.

Az SSD előnyei miatt nagyobb projektek kapcsán mára megkerülhetetlen technológiák, az adatok archiválására azonban a HDD technológia továbbra is jó megoldás.

5. Hordozhatóság

A mobilitás lényeges szempont, a dolgozatban említett felméréseket sokszor célszerű még a mérés helyszínén ellenőrizni, esetleges pótmérésről dönteni. Ugyan a PC nagy előnye lehet, hogy az összes komponense akár külön-külön is beszerezhetőek és cserélhetőek, azonban laptop használata esetén a felhasználó nincs helyhez kötve. A laptopok körében is vannak workstation (ws - munkaállomás) kategóriájú gépek, amelyek kifejezetten ilyen munkákra vannak készítve.

6. Ár

Az ára egy számítógépnek, ami alkalmas az általam végzett munka elvégzésére széles skálán mozog. Mai árakkal kb. 200 000 Ft-tól (650€) indulnak az olyan gépek, amikkel a modellezést és a renderelést el lehet végezni, de ha hosszútávra tervezünk, akkor érdemes lehet befektetni egy nagyobb teljesítményű gépbe, viszont ezek árai 400 – 500 000 Ft-tól (1300 - 1600€) indulnak.

6.2 Szoftverek

Figyelembe kell venni azt is, hogy a programok egy része, amit a munkához használtam, fizetős; ingyenesen csak próba vagy diákverziót lehet használni, de ez profitszerzés céljára nem használható. A kereskedelmi programok, amiket felhasználtam:

- AutoCAD Civil 3D 2016: Ez a program az Autodesk család tagja és kifejezetten építőmérnöki alkalmazásra van kifejlesztve, sok beépített modul segíti az építőmérnökök munkáját (pl. cső-, úthálózat kialakításhoz, lakótömbök kialakításához

eszközök). Jelenleg a legfrissebb verzió a 2017-es, ami egy éves előfizetéssel 2290 € [14], ezzel szemben a „sima” AutoCAD előfizetése egy évre 1400 €. [15]

- Geomagic Wrap és Geomagic Design X: Ezek a programok a mérésekből előállított pontfelhőt tudják szerkeszteni, javítani, valamint az utófeldolgozáshoz szükséges eszközökkel is rendelkeznek, felületmodell készítésére és vektoros modellezésre is alkalmasak (utóbbira főként a Design X).
- Unity: A Unity egy multiplatform motor, ami alkalmazások készítéséhez kiváló választás. Én egy ingyenes csomagot töltöttem le, de vannak fizetős (Plus - \$35/hó és Pro - \$125/hó) csomagok, amik több lehetőséget rejtenek magukban, persze ez csak fejlesztőknek ajánlott, magánfelhasználóknak elég az ingyenes. [16]

Használhatunk ingyenesen letölthető, nyílt forráskódú programokat is, ahogy már az előzőekben említettem, ám azokkal a munka nem minden esetben hatékony, viszont az OS programok nagy előnye, hogy folyamatosan frissítik, sorra kerülnek ki hozzájuk újabb és újabb kiegészítők (pluginok). Emellett ha rendelkezünk kellő ismeretekkel az adott program programozási nyelvében (a Blender nyelve pl. a python), akkor mi magunk is formálhatjuk azt, készíthetünk hozzá modulokat (amiket meg is oszthatunk másokkal), hogy hatékonyabb legyen.

6.3 Felhő alapú számítástechnika - Cloud Computing

A felhőalapú számítástechnika (Cloud Computing, CC) az informatika egy mára széles körben elterjedté ága. Számos felhőalapú szolgáltatás vált elérhetővé az évek során, ezek közös vonása, hogy a szolgáltatásokat nem egy dedikált hardvereszközön üzemeltetik, hanem a szolgáltató eszközein elosztva, a szolgáltatás üzemeltetési részleteit a felhasználotól elrejtve. Az alapján, hogy a szolgáltatáshoz milyen módon és a felhasználók milyen köre férhet hozzá négy típus különíthető el: privát, hibrid, közösségi, illetve publikus felhő. Általánosságban a cloud computingnak számos előnyös tulajdonsága van magánfelhasználóknak és cégeknek egyaránt [17]:

1. Költséghatékony
 - a. A hardvereszközök megvásárlásának költségét a szolgáltatás használatának díja váltja fel – ez például lehet a bérelt számítási kapacitás, hálózati forgalom, vagy a felhasználók száma alapján kiszámolt összeg.
 - b. Az üzemeltetési, sokszor nehezen tervezhető feladatok nem a felhasználókat terhelik.

2. Bárhonnan elérhető
 - a. Egy felhő-alapú megoldás (főleg publikus felhő) szolgáltatás esetében a szolgáltatás bárhonnan könnyen elérhető lehet.
 - b. A szolgáltatások jelentős része platformtól független
3. Rugalmas, méretezhető
 - a. A számítási kapacitás a felmerülő igényeknek megfelelően skálázható
 - b. Amennyiben nincs rá szükség, használata felfüggeszthető
4. Megbízható, biztonságos
 - a. Az szolgáltatásokért felelős szerverparkok védett helyeken található
 - b. A szolgáltatások nem egy dedikált helyről, hanem több osztott szerverfarmról származnak

A cloud computing típusai [18]:

- Szoftver-alapú számítási felhő (Software as a Service, SAAS): a szoftvert nyújtja szolgáltatásként jellemzően egy vékony kliensen keresztül, egy böngészővel lehet elérni, pl. Google docs. A felhasználó nem fér hozzá a mögöttes tartalomhoz, pl. az operációs rendszerhez és hardver infrastruktúrához
- Platform-alapú számítási felhő (Platform as a service, PAAS): esetenként skálázható operációs rendszerszolgáltatás (Pl. Microsoft Azure, Google App Engine)
- Infrastrukturális számítási felhő (Infrastructure as a Service, IAAS): Virtuális hardverszolgáltatás (pl. Amazon EC2)

Esetemben a platform-alapú számítási felhőszolgáltatást lenne célszerű megvizsgálni, hogy az elvégzendő munkához szükséges paraméterekkel rendelkező számítógépet mennyibe kerülne bérelni felhőben. Ennek az ára nagyon széles skálán mozog és több szolgáltató is rendelkezésre áll. A szolgáltatóknál be lehet állítani, hogy mi legyen a fő szempont, mire akarjuk használni a számítógépet, valamint milyen időintervallumban (egy nap hány órát üzemeljen és egy héten mennyi nap, pl. heti öt nap, napi tíz órában). A Microsoft-nál az „N” széria a legalkalmasabb, mivel ez főleg a GPU-ra (Graphics Processing Unit – Grafikus számítási egység) fókuszál, ennek előfizetése 389,00 €-tól indul havonta [19].

Egy hasonló paraméterekkel rendelkező gép az Amazonon (bár itt valamennyivel kevesebb opció közül lehet választani) havi szinten 313,87 €-tól [20].

Ha a 6.1-es fejezetben lévő szempontokat figyelembe véve magunk állítunk össze egy számítógépet, akkor az ~650 €-tól indulna (természetesen egy workstation kategóriájú gép ettől sokkal többbe kerülne), viszont ezt csak egyszer kell kifizetni és bármikor cserélhetjük benne a processzort, videokártyát, bővíthetjük a memóriáját. Viszont, ha meghibásodás történik, akkor nekünk kell elvinni szervizbe, a munka a javítás ideje alatt nem folytatható, ezzel szemben a felhő-alapú szolgáltatásnál ezzel a szolgáltató foglalkozik, valamint a felhőben vásárolt számítógépet bárhol elérhetjük, nem kell magunkkal vinni, ami hordozhatóság szempontjából nagyon nagy előny.

7. Összefoglalás

Dolgozatom célja, hogy bemutassam, különböző módokon felmért objektumokból előállított pontfelhőket hogyan tudunk felhasználni VR/AR környezetekben, valamint, hogy az építőiparban és egyéb területeken ennek milyen alkalmazási lehetőségei vannak.

Ennek kutatása közben kitértem a földi lézerszkennelésre, annak milyen lépései vannak, és hogyan kell egy objektumot felmérni, a végterméket előállítani. Több eszközt is bemutattam a hagyományos lézerszkenneléstől, megvizsgáltam, hogy az azokkal előállított pontfelhők mennyire hasznosíthatók és milyen a minőségük a lézerszkenneléstől előállított pontfelhőkhöz viszonyítva.

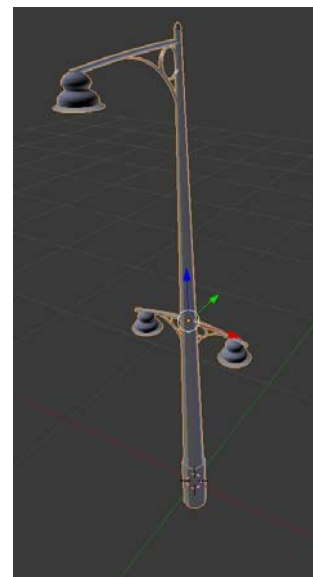
A nyers adatok megvizsgálásakor látszott, hogy a pontfelhők helyenként hibákat tartalmaznak, amiket ki kellett javítanom. Ennek elemzésekor kitértem az ehhez alkalmazható módszerekre és az általam preferált eszközökre is. Miután a nyers adatokból előállítottam a felületmodellt, előállítottam az objektumok vektoros modelljét is (20. ábra). A modellezés lépéseit részletesen leírtam; kitértem az ingyenes és kereskedelmi programokra is.



A villanyoszlop a valóságban



A villanyoszlopról készített pontfelhő



A pontfelhő alapján készített vektoros modell

20. ábra

Bemutattam a virtuális és kiterjesztett valóság fogalmakat, majd leírtam a modellek VR és AR környezetbe történő beillesztését, leírtam azoknak elkészítési menetét.

Kitértem a technológia felhasználási területeire. Példákat hoztam arra, hogy az építőiparban milyen alkalmazási lehetőségek vannak és bővebben kitértem a filmiparban használatos CGI és VFX használatára.

Végül elkészítettem a gazdasági elemzést, amiben kitértem a hardver és szoftver részre is. Bemutattam, hogy melyek voltak az általam használt fizetős és melyek voltak az ingyenes programok, ugyanis a pontfelhő utófeldolgozásánál és a vektormodellek készítésénél egyaránt felhasználtam ingyenes és kereskedelmi programokat is. Megvizsgáltam annak a lehetőségét is, hogy megérné-e beruházni felhőszolgáltatásba, az milyen költségekkel járna, ismertettem annak pozitív és negatív oldalát is.

Mindezekből azt a következtetést vontam le, hogy a technológiát fel lehet használni építőipari célokra, de egyelőre a mérnöki pontosságú modellek levezetése nem ideális környezetben (amilyen az általam, szabadban felmért villanyoszlopok esetén adódott) még számos problémába ütközhet. A VR/AR alkalmazások a térbeli megjelenítésen keresztül azonban már ma is hatékonyan segíthetik az irodai és terepi építőmérnöki munkákat, melyeknek a dolgozatban tárgyalt térbeli adatnyerési eljárások hasznos segítői lehetnek.

Köszönetnyilvánítás

Szeretném megköszönni a dolgozat elkészítéséhez nyújtott hatalmas segítségét és fáradhatatlan munkáját két konzulensemnek, Dr. Lovas Tamásnak és Somogyi József Árpádnak, nélkülük ez a dolgozat nem jöhetett volna létre. Emellett még köszönettel tartozok Weszelovits Gergely MSc hallgatónak, aki a VR technológiában nyújtott hatalmas segített.

Irodalomjegyzék

- [1]: Lovas Tamás, Berényi Attila, Barsi Árpád: Lézerszkennelés, Monográfia (TERC Kft. 2012)
- [2]: Faro Focus 3D 120S - <http://www.laserscanning-europe.com/en/glossary/faro-focus-3d-120>
- [3]: How does the Xbox Kinect work (Robert Cong és Ryan Winters)
<http://www.jameco.com/jameco/workshop/howitworks/xboxkinect.html>
- [4]: Vector graphics advantages - <http://naldzgraphics.net/design-2/vector-graphics-advantages/>
- [5]: Rendering - [https://en.m.wikipedia.org/wiki/Rendering_\(computer_graphics\)](https://en.m.wikipedia.org/wiki/Rendering_(computer_graphics))
- [6]: A taxonomy of Mixed Reality visual displays (Paul Milgram, Fumio Kishino); IEICE Transactions on Information Systems, Vol E77-D, No.12 December 1994.
http://etclab.mie.utoronto.ca/people/paul_dir/IEICE94/ieice.html
- [7]: Sketchup hivatalos oldala - <http://www.sketchup.hu/>
- [8]: Unity Asset Store - <https://www.assetstore.unity3d.com/en/>
- [9]: How to make an AR app in 5 minutes with Unity and Vuforia (Danny Goodayle)
<http://www.justapixel.co.uk/how-to-make-an-ar-app-in-5-minutes-with-unity-and-vuforia/>
- [10]: Bödő Gábor: Virtuális tartalomgyártás – mintavételezés a valós világból, 2015. (diplomamunka)
- [11]: VFX jobs and what they do - <http://blog.digitaltutors.com/vfx-jobs/>
- [12]: Recommended Hardware for AutoCAD -
<https://www.pugetsystems.com/recommended/Recommended-Systems-for-Autodesk-AutoCAD-134/Hardware-Recommendations>
- [13]: Recommended Hardware for AutoCAD -
<https://www.pugetsystems.com/recommended/Recommended-Systems-for-Autodesk-AutoCAD-134/Hardware-Recommendations>

- [14]: AutoCAD Civil 3D előfizetés - <http://www.autodesk.hu/products/autocad-civil-3d/subscribe>
- [15]: AutoCAD előfizetés - <http://www.autodesk.hu/products/autocad/subscribe>
- [16]: Unity előfizetés - https://store.unity.com/?_ga=1.171234345.863383023.1475081216
- [17]: Cloud Computing - <http://searchcloudcomputing.techtarget.com/definition/cloud-computing>
- [18]: Felhő alapú számítástechnika - https://hu.wikipedia.org/wiki/Felh%C5%91_alap%C3%BA_sz%C3%A1m%C3%ADt%C3%A1stechnika
- [19]: Microsoft Azure N széria ár és leírás - <https://azure.microsoft.com/en-gb/pricing/details/virtual-machines/series/#n-series>
- [20]: Amazon EC2 - <http://calculator.s3.amazonaws.com/index.html>