

KONSPEKT AKVIZÍCIE GEOPRIESTOROVÝCH DÁT PRE VYTVORENIE 3D MODELU REFERENČNÉHO LETISKA

CONSPECT OF GEOSPATIAL DATA ACQUISITION FOR CREATING 3D MODEL OF REFERENCE AIRPORT

Filip Škultety
Air Transport Department
University of Žilina
Univerzitná 8215/1
010 26, Žilina
skultety@fpedas.uniza.sk

Michal Janovec
Air Transport Department
University of Žilina
Univerzitná 8215/1
010 26, Žilina
Michal.Janovec@fpedas.uniza.sk

Abstract

When rendering large scenarios for flight simulators, we encounter similar problems as in the case of scenes for cinematography or computer games. In the case of this work, we solve the rendering of the earth's surface for any camera configuration, and we try to maximize the quality of the rendered images without a visible decrease in the rendering frequency. The article is devoted to the creation of a reference model of the Piešťany airport (LZPP) as a part of the scenery for a certified flight simulator. The presented solution maintains a stable 60 frames per second when rendering the airport surface from any distance and direction.

Keywords

Image generator, flight simulator, geospatial data, reference model, airport, Trian3D

1. Úvod

Pri vykresľovaní rozsiahlych scenérii pre letecké simulátory sa stretávame s podobnými problémami ako v prípade scén pre kinematografiu alebo počítačové hry. V prípade tejto práce riešime vykresľovanie povrchu zeme pre ľubovoľnú konfiguráciu kamery, a snažíme sa maximalizovať kvalitu vykreslených snímok bez viditeľného poklesu vo vykresľovacej frekvencii. Článok je venovaný tvorbe referenčného modelu letiska Piešťany LZPP ako súčasťou scenérie pre certifikovaný letecký simulátor. Predstavené riešenie udržiava stabilných 60 snímok za sekundu pri vykresľovaní povrchu letiska z ktorejkoľvek vzdialenosti a smeru.

2. Vykresľovací reťazec

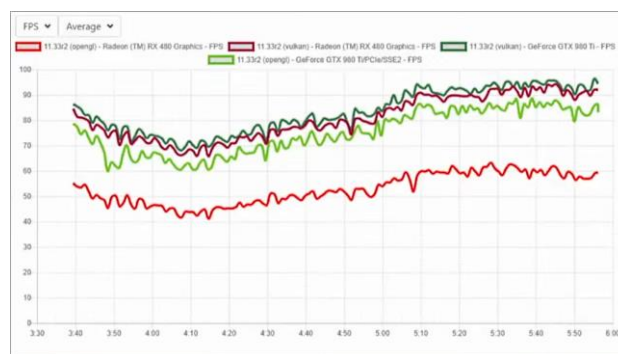
2.1. Definícia

Cieľom vykresľovacieho reťazca je transformovanie trojrozmernej scény na dvojrozmerný obraz založený na polohe a vlastnostiach virtuálnej kamery, ktorý je možné vykresliť na displej. (Brezoňáková et al., 2019) Vykresľovací reťazec sa medzi rôznymi grafickými prostrediami mierne líši, princíp fungovania je však veľmi podobný, nasledujúca časť sa teda bude hlavne venovať všeobecnému opisu generického vykresľovacieho reťazca.

2.2. Počítačová grafika

V súčasnosti medzi najpopulárnejšie grafické API (programovacie prostredie aplikácií) patria OpenGL a Vulkan od spoločnosti Khronos, DirectX od spoločnosti Microsoft a Metal od spoločnosti Apple. Letecký simulátor Prepar3D, ktorý je založený na platforme ESP od spoločnosti Microsoft teda využíva grafické API DirectX, najnovšie verzie využívajú verziu DirectX 11. Letecký simulátor X-Plane od spoločnosti Laminar Research využíva OpenGL. Hlavnou výhodou grafického API OpenGL oproti

DirectX je kompatibilita z viacerými operačnými systémami, na rozdiel od simulátora Prepar3D, ktorý je obmedzený na operačný systém Windows, X-Plane podporuje aj operačné systémy Linux a MacOS. V súčasnosti je snaha prejsť na takzvané "Low-overhead" grafické API, ako sú DirectX 12, Vulkan a Metal. Tieto grafické API dokážu pracovať z hardvérom na nižšej úrovni čo umožní efektívnejšie využitie CPU aj GPU. Na obrázku 1 je vidno porovnanie snímok za sekundu v simulátore X-Plane 11 za použitia OpenGL a Vulkan. Využitie grafického API Vulkan nezlepšilo výkon len z pohľadu snímok za sekundu, ale aj z konzistencie tejto hodnoty.



Obrázok 1: Porovnanie výkonu grafických API OpenGL a Vulkan v simulátore X-Plane 11 (Zdroj:Laminar Research, 2019)

2.3. Spracovanie geometrie

Prvou časťou spracovania geometrie je „input assembly“, tzv. skladanie vstupov. Úlohou tejto časti je čítať vstupné dáta a zložiť tieto dáta do rôznych „primitive“ typov, ktoré môžu byť spracované ďalšími časťami vykresľovacieho reťazca. Medzi základné „primitive“ typy patria napríklad body, čiary a trojuholníky. Všetky tieto „primitive“ typy sú definované vrcholmi—vertices. Každý vertex je definovaný pomocou

karteziánskej súradnicovej sústavy, teda vzdialenosťou od stredu súradnicovej sústavy v osi x, y, z. Trojuholník je zadefinovaný pomocou troch vertexov, môžeme však ušetriť počet vertexov použitím pásu trojuholníkov kde ten istý vertex je použitý aj pre susedný trojuholník. Výstupne dáta z „input assembly“ sú ďalej spracovávané vertex „shaderom“. Hlavnou úlohou vertex „shaderu“ je vykonávať transformačné operácie na jednotlivých vertexoch. Na zmenu polohy jednotlivých vertexov sa používajú transformačné matice.

$$\begin{bmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ w \end{bmatrix} = \begin{bmatrix} x_0 + \Delta x \\ y_0 + \Delta y \\ z_0 + \Delta z \\ w \end{bmatrix}$$

V tomto príklade sa jedná o jednoduchú zmenu pozície vertexu. Hodnoty Δx , Δy , Δz reprezentujú o koľko bude vertex zo súradnicami x_0 , y_0 , z_0 posunutý v osiach x, y, z. Písmeno w označuje či sa jedná o pozičný vektor, označovaný 1, alebo smerový vektor, označovaný 0. (Supnik, 2019)

Vertex „shader“ pomocou týchto transformačných operácií pripraví scénu rozmiestnením jednotlivých objektov. Ďalším krokom je transformácia kamery, čiže zadefinovanie jej polohy a orientácie. Kamera skladá z dvoch bodov a vektoru, pre zadefinovanie jej polohy je potrebné zadefinovať polohu samotnej kamery, referenčnú pozíciu – bod v strede zorného poľa kamery a vektor, ktorý smeruje kolmo hore od bodu pozície smeru kamery. Po transformácii kamery nasleduje transformácia projekcie. V tomto kroku sa vytvorí zrezaný kužeľ – „view frustum“, ktorý bude predstavovať naše zorné pole. Parametre pre definíciu tohto zrezaného kužeľa sú štyri: vertikálny uhol zorného poľa, pomer šírky a výšky zorného poľa a hodnoty zNear a zFar. Tieto posledné dve hodnoty vyjadrujú vzdialenosť dvoch rovín ktorými je kužeľ zrezaný. Po zadefinovaní „view frustumu“ je možné odstrániť všetky vertexy ktoré sa v ňom nenachádzajú. Tento proces sa nazýva „frustum culling“ a zvyšuje efektivitu vykresľovacieho reťazca tým že sa vertexy nachádzajúce sa mimo zorného poľa ďalej nespracovávajú. Podobným procesom je „back-face culling“, čo je proces ktorí zabraňuje vykresľovaniu zadnej strany polygónov. Back-face culling testuje či sa jednotlivé vertexy nachádzajú v smere hodinových ručičiek, dokáže tak určiť či sa jedná o prednú, alebo zadnú stranu polygónu. Keďže vo väčšine prípadov potrebujeme zobrazíť len prednú stranu polygónu tento proces taktiež zvýši efektivitu vykresľovacieho reťazca. Pri potrebe vykresľovania oboch strán polygónu je jedným riešením vytvorenie identického polygónu z opačným normálovým vektorom, keďže tieto polygóny zdieľajú vertexy vplyv na výkon je malý. V niektorých prípadoch môže byť užitočné vykresliť veľké množstvo obojstranných polygónov, napríklad pri vykresľovaní stromov zložených z dvoch kolmých billboardov v leteckom simulátore. V tomto prípade je možné zapnúť obojstranné vykresľovanie len na vykreslenie týchto objektov, toto riešenie je využité pri vykresľovaní stromov v simulátore X-Plane. Ďalším podobným procesom je „occlusion culling“, tento proces zabraňuje vykresľovaniu objektov ktoré sú zakryté inými objektami. V leteckom simulátore môže byť tento proces obzvlášť efektívny pretože kokpit a časti lietadla ako napríklad krídla môžu zakrývať značnú časť scenérie z veľkým počtom objektov.

2.4. Rasterizácia

Jedným zo spôsobov na premenenie trojrozmernej scény na dvojrozmerný obraz je rasterizácia (Ficová et al. 2016; Novák, 2018). Existujú alternatívne spôsoby na splnenie tejto úlohy ako napríklad „ray-tracing“, avšak obzvlášť pre využitie v generátoroch obrazu pre letecké simulátory v súčasnosti tieto spôsoby nie sú praktické, z tohto dôvodu sa budem v mojej práci venovať len rasterizácii.

Prvým krokom rasterizácie je vypočítanie súradníc vertexov v dvojrozmernom priestore nazývanom „window space“. Každá časť trojrozmernej scény je rozdelená na fragmenty ktoré majú parametre ako napríklad svoju pozíciu v dvojrozmernom priestore „window space“, hĺbku – vzdialenosť od kamery, farbu a súradnice na UV mape textúry. Tieto fragmenty sa dajú považovať za kandidátov na pixely ktoré budú vykreslené na obrazovke. V kroku „fragment shader“ sú zadefinované vlastnosti jednotlivých fragmentov akými sú napríklad farba, ktorá je načítaná z pixelu zo špecifickými súradnicami textúry na špecifický fragment alebo či na fragmente vznikajú odrazy. Ďalším dôležitým krokom procesu rasterizácie je „depth-test/z-test“. V tomto teste sa určuje či je fragment viditeľný alebo je zakrytý iným fragmentom. Počas tohto testu sa využíva „z-buffer“, je to podstate tabuľka ktorá pre každú súradnicu obrazu zapisuje vzdialenosť fragmentu od kamery. Pokiaľ má nový fragment z rovnakými súradnicami ako predchádzajúci nižšiu hodnotu vzdialenosti od kamery je táto hodnota nahradená novou, nižšou hodnotou. V prípade priehľadných fragmentov sa využíva krok z názvom „blending“, tento krok vypočíta farbu fragmentu na základe vlastností priehľadných fragmentov z rovnakými súradnicami. Výsledkom procesu rasterizácie je „frame/colour buffer“, v ňom sú uložené hodnoty výsledných farieb pixelov pre súradnice x, y.

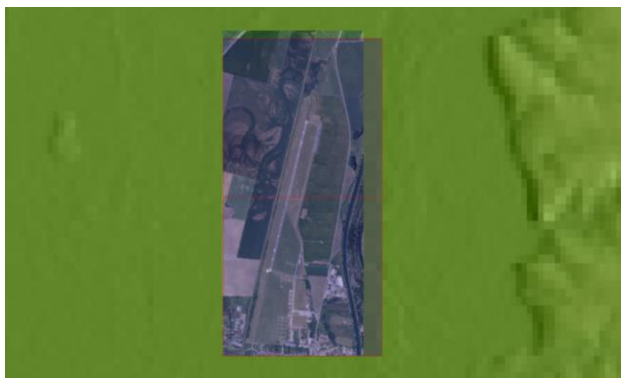
3. POSTUP PRÁCE TVORBY LETISKA

3.1. Konfigurácia projektu

Program Trian3DBuilder bude použitý na tvorbu základnej geometrie letiska, čiar a svetiel. Prvým krokom je importovanie elevačných dát cez menu „Data > Elevation Data“. Podobne sa importujú aj letecké snímky cez menu „Satellite Maps“. Po importovaní leteckej snímky nasleduje základná konfigurácia projektu. V menu „Tile Grids“ sa nastaví „Tile Size“ na rozmery 2048x2048 metrov a v menu Tile Grids, sekcii Level of Detail sa nastaví hodnoty „tex_res“ na 4096 a „mesh_res“ na 90 m. Hodnota tex_res nastaví rozlíšenie textúry, ktorá bude exportovaná. Pre čo najmenší počet textúr sa odporúča nastaviť rozlíšenie textúr 4096x4096 pri hustote pixelov 0.5m/px. Z tohto dôvodu boli zvolené rozmery Tile Size ako ½ rozlíšenia textúr. Mesh_res definuje rozlíšenie terénu v metroch, hodnota tohto údaju je maximálna vzdialenosť medzi dvoma bodmi „gridu“ terénu.

Po zadefinovaní hodnôt Tile Size sa pomocou nástroja „Draw Tiles“, na ktorej sa bude nachádzať letisko vrátane plochy okolo letiska ktorá bude súčasťou scenérie. V menu „Generator“ sa vyberie typ projekcie „Geographic (flat earth)“ a nastaví sa stred projektu pomocou „Set on Grid“. Tlačidlo „Show Position“ zobrazí na pár sekúnd v dvojrozmernom zobrazení stred projektu. V menu „Settings“, sekcii „Format Settings“ sa zapne možnosť triangle „strips“ a v sekcii „texture“ sa nastaví formát „dds“. V menu „Generator“ sa dá projekt vygenerovať tlačidlom

Build, po vygenerovaní sa dá prejsť do trojrozmerného zobrazenia projektu tlačidlom „3D view“.



Obrázok 2: „Project tiles placeholder“ (Zdroj: Autor)

3.2. Vektorizácia prvkov

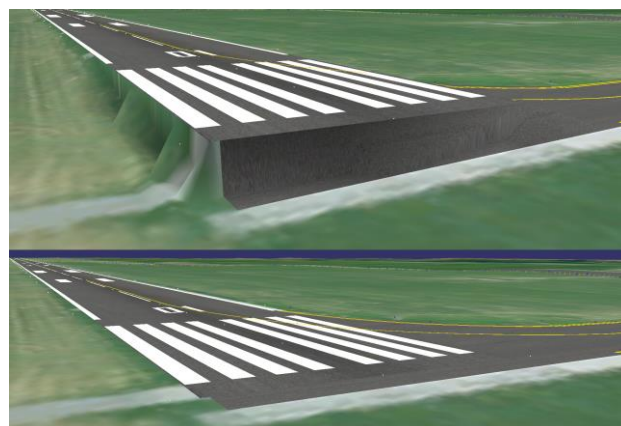
Jednotlivé prvky letiska sú zadané vektormi. Pre každý prvok letiska sa vytvorí „Vector ID“. Jednotlivé vlastnosti týchto vektorov sú zadané za pomoci „modifierov“.

Vzletová a pristávacia dráha je zadaná jedným vektorom zo začiatkovým a koncovým bodom. Presné umiestnenie dráhy je veľmi dôležité, preto sa na jej umiestnenie nepoužije referenčná letecká snímka, ale zadanú sa presné súradnice počiatkového a koncového bodu vektora a taktiež výška týchto bodov v poli „Position“ v sekcii „Line Edit“.

„Modifier Height Type“ ovláda výšku a kontúru vektorových dát, pre vzletovú a pristávaciu dráhu bol jeho hlavný parameter nastavený na „Vector Height“, výška dráhy bude teda nastavená na hodnotu „Z“ z poľa „Position“. Pri začiatku možnosti „Plane inside points“ budú všetky body v oblasti vyrovnané na výšku vektora. „Outline“ vygeneruje obrys čiar určitej šírky, v tomto prípade je to šírka dráhy, čiže 30 metrov. V parametri textúra sa vybrala textúra asfaltovej dráhy. Správny spôsob namapovania tejto textúry sa dá zabezpečiť modifierom „Mapping“, kde sa nastavením parametru „Mapping type“ na možnosť „Linear“ vygeneruje lineárne mapovanie pozdĺž vektora.

Pomocou „modifieru Insert“ zo zapnutým parametrom „Cut hole“ sa vyreže v teréne diera okolo obrysu vygenerovaného modifierom „Outline“. „Modifier Border“ je potom použitý na vytvorenie hranice medzi plochou dráhy a okolitým terénom, zaškrtnutím parametru „Insert“ sa zapne možnosť využitia tohto modifieru na úpravu terénu. „Parameter Width“ nastavuje šírku hranice prechodu medzi plochou dráhy a okolitým terénom. Druh vyhladenia tohto prechodu sa nastavuje pomocou parametru „Smoothing type“. Priority type určuje, kedy sa modifier použije počas procesu generovania terénu. „Lower own insert“ vygeneruje hranicu hneď po vyrezaní diery pomocou svojho modifieru „Insert“.

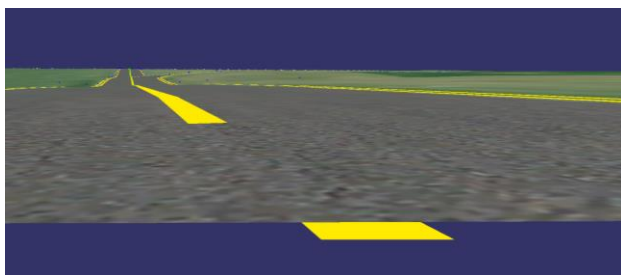
„Collision modifier“ ovláda spôsob správy kolízií. Parameter „Bounding Box“ vytvorí ohraničenie okolo dráhy na výpočet kolízií, parametre „Test upwards“ a „Test downwards“ sú vybrané pre aktivovanie kolízií zo skupinami vektorov vyššej aj nižšej priority. „Modifier Geometry“ vygeneruje geometriu plochy. Rolovacie dráhy môžu byť vytvorené dvoma spôsobmi, buď využitím čiar podobne ako u vzletovej a pristávacej dráhy, alebo pomocou plôch.



Obrázok 3: Efekt modifieru „Border“ na terén okolo vzletovej a pristávacej dráhy (Zdroj: Autor)

Pri využití čiar je potrebné zmeniť parametre niektorých modifierov v porovnaní so vzletovou a pristávacou dráhou. V modifiery „Height Type“ je hlavný parameter nastavený na „Relief“. Geometria rolovacej dráhy bude teda kopírovať terén. Výška terénu je určená na každom vertexe, teda čím viac vertexov bude použitých, tým presnejšie bude geometria rolovacej dráhy kopírovať terén. V modifiery „Insert“ sa zapne parameter „Neutralize Inserts“, tento parameter dokáže pomocou „boolean“ operácií vyzerať diery cez iné diery vyrezané modifierom „Insert“. Táto metóda umožní relatívne rýchle zadefinovanie rolovacích dráh, je však potrebné v miestach, kde sa rolovacia dráha križuje s inou rolovacou alebo vzletovou a pristávacou dráhou pridať zaoblené plochy medzi tieto križujúce sa dráhy. Na vytvorenie zákrut je možné použiť nástroj „Create Bezier Curve“, avšak je po vytvorení potrebné tieto krivky premeniť na obyčajné čiar. Tento nástroj sa dá využiť aj na rýchle vytvorenie rovnej čiar pravidelne rozdelených vertexami. Počet vertexov na čiare premenenej z krivky sa dá nastaviť pomocou poľa „Subdiv Steps“. Pri vytváraní plôch sa využívajú rovnaké modifiery ako pri čiarach, v modifiery „Mapping“ sa však nastaví parameter „Mapping Type“ na „Area“. Pri vytváraní plôch je potreba aby boli jednotlivé vertexy vkladané proti smeru hodinových ručičiek. Kvôli spresneniu kopírovania terénu je potrebné dlhšie hrany rozdeliť viacerými vertexami podobne ako tomu bolo u dráh vytvorených čiarami.

Pre vytvorenie čiar sa dá použiť modifier „Outline“ alebo „Marking“. Pri použití „Outline“ je potrebné nastaviť parameter „Mapping type“ v modifiery „Mapping na Linear“. Použitie modifieru „Marking“ umožňuje jednoduchšie vytvorenie rôznych typov značení, ako napríklad dvojité alebo prerušované čiar. Obrázok 4 zobrazuje nedostatočný počet vertexov stredovej čiar vzletovej a pristávacej dráhy.



Obrázok 4: Nedostatočný počet vertexov stredovej čiary na komplexnom teréne (Zdroj: Autor)

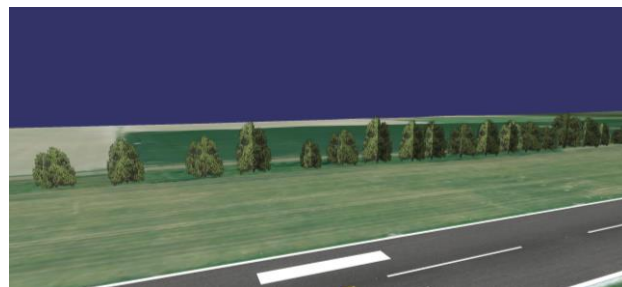
Druhy čiar sa dajú nastaviť parametrom Type. Parametre „Marking Length“ a „Gap Length“ nastavujú dĺžku čiar a dĺžku medzery pri prerušovanej čiare, „Marking Width“ nastavuje šírku značenia. Parameter „Side Offset“ umožňuje posunutie vykreslenej čiary od vektora ktorým je daná čiara definovaná. Tento parameter je užitočný pri vytváraní značenia na vzletovej a pristávacej dráhe, kde je možné pomocou tohto parametra jednoducho vložiť značenia ktoré sú paralelné z dráhou. Pre nastavenie priority vykresľovania sa použije „Layer modifier“, čím je vyššia hodnota „Layer ID“, tým je nastavený väčší polygon offset, čiže vyššia priorita vykresľovania. Pokiaľ sa čiara nachádza na nerovnom teréne je potrebné nastaviť „Height type“ na „Relief“, aby čiara nasledovala terén a vertikálny ofset, aby sa čiara nachádzala niekoľko centimetrov nad terénom. Ako pri rolovacích dráhach čiary je podľa komplexnosti terénu treba rozdeliť.

3.3. Vkládanie objektov

Objekty môžu byť vkladane buď v bode, na čiary alebo v oblasti. Vkládanie v bode za pomoci modifieru „Point Object Placer“ je vhodné pre vkladanie jedinečných objektov ale aj pre objekty ktoré chceme umiestniť v presnej polohe alebo zo špecifickým smerom. Tento modifier však dokáže zdefinovať len objekt, ktorý bude do projektu vložený, rotáciu tohto objektu je potrebné nastaviť modifierom „Point Offset“.

Pri potrebe vkladania objektov v rade z presným rozstupom, alebo kde rozstup nie je dôležitý je výhodné použiť „Contour Object Placer“. Tento modifier je možné použiť v kombinácii s inými modifiermi, napríklad na automatické vkladanie svetiel na rolovacej dráhe. Pomocou parametru „Scale factor“ je možné zmeniť veľkosť objektu a pomocou „Scale variance“ je možné túto veľkosť meniť náhodne v určenom rozhraní. 0% znamená konštantnú veľkosť, 100% znamená, že objekt môže mať veľkosť ktorá je rovná 50% až 150% pôvodnej veľkosti. Pri hodnote 200% je veľkosť od 0% až po 200% pôvodnej veľkosti. Vzdialenosť objektov sa nastavuje parametrom „Distance“ a je taktiež možné túto vzdialenosť náhodne meniť v určenom rozhraní za pomoci parametru „Distance variance“. 0% znamená konštantnú vzdialenosť medzi vloženými objektami, 100% znamená rozdiel vzdialenosti od 0 po hodnotu vzdialenosti v poli „Distance“. Smer otočenia objektov je zdefinovaný v parametri „Standard alignment“ a môže byť parametrom „Rotation type“ nastavený ako „fixed“ kedy je rotácia rovnaká pre každý vložený objekt „random“, kedy je rotácia objektov náhodná, alebo „aligned“, kedy rotácia objektov závisí od smeru čiary. Pri nastavení „random“ je taktiež možné nastaviť parameter „Rotation steps“, ktorý nastavuje v ako veľkých krokoch bude objekt otáčaný. „Parameter Offset“ umožňuje umiestniť objekty určitú vzdialenosť od čiary a pomocou „Offset side“ sa dá vybrať

smer od čiary. „Offset variance“ umožňuje meniť veľkosť náhodne v určitom rozhraní. 0% znamená konštantnú vzdialenosť, 100% znamená, že objekt môže mať vzdialenosť ktorá je rovná 50% až 150% pôvodnej vzdialenosti. Pri hodnote 200% je vzdialenosť od 0% až po 200% pôvodnej vzdialenosti. Parameter „Place on Corners“ umiestňuje objekty na každom vertexe, miesta medzi vertexami sú vyplnené objektami podľa nastavenia ostatných parametrov modifieru. Umiestniť objekty bez medzery je možné za pomoci modifieru „Touching Objects“, toto umožňuje napríklad vytvorenie stien alebo plotov. Obrázok 5 znázorňuje príklad využitia parametrov „variance“.



Obrázok 5: Príklad využitia parametrov „variance“ (Zdroj: Autor)

Na vkladanie objektov v oblasti sa používa modifier „Area Object Placer“, parametre tohto modifieru fungujú obdobne ako parametre modifieru „Contour Object Placer“. Hlavné využitie tohto modifieru je vkladanie lesov.

3.4. Airport Module X-Plane

„Airport Module“ umožňuje vygenerovanie letiska na základe dát importovaných z databázy letísk apt.dat zo simulátoru X-Plane. Táto databáza obsahuje letiská zo „Scenery Gateway“ ktorá vznikla ako metóda pre používateľov simulátoru X-Plane na nahrať letísk vytvorených programom WED ktorých kvalita je potom posudzovaná moderátorom ktorý rozhodne o ich prijatí a neskoršom vydaní v simulátore X-Plane alebo o ich zamietnutí. V prípade ak bolo moderátorom prijatých viac scenérií rovnakého letiska, bude jedno letisko označené ako doporučené, toto letisko považuje moderátor za najkvalitnejšie a bude vydané v simulátore X-Plane. Za moderáciu je v súčasnosti zodpovedný Julian Lockwood ktorý pôvodne prišiel s nápadom pre „Scenery Gateway“ ako zo spôsobom na relatívne jednoduchú a rýchlu implementáciu 3D letísk do simulátoru. Tento systém sa stretol s veľkým úspechom a v čase písania tejto práce obsahuje 38232 letísk z čoho je 32148 vytvorených 8068 registrovanými používateľmi a 29668 letísk obsahuje 3D objekty. (Han-Wei, 2018)

Pri posudzovaní kvality letísk moderátor vo všeobecnosti zvažuje chyby zaznamenané v log súbore, presnosť číselných označení vzletových a pristávacích dráh, prekážky na vzletovej a pristávacej dráhe, rolovacej dráhe alebo na stojisku, nezablokované umiestňovanie objektov „autogenu“ ako sú stromy, budovy, cesty, navigačné pomôcky a podobne v miestach kde je ich umiestňovanie nežiadúce, prezenciu 3D objektov, prezenciu zdefinovaných dráh na pohyb lietadiel a vozidiel pozemnej obsluhy, logiku týchto dráh, vyrovnanie terénu v okolí letiska a subjektívny vzhľad scenérie letiska. Scenéria je taktiež automaticky kontrolovaná programom WED pred jej nahraťím do systému „Scenery Gateway“. Pri varovaní

sa chyba zapíše do log súboru, pri väčších problémoch program neumožní nahratie scenérie do „Scenery Gateway“. Letiská už nahrané v „Scenery Gateway“ môžu byť automaticky upravované Python skriptom z názvom „WEDBot“, tento skript pôvodne vznikol na nahrávanie „legacy“ letísk z X-Plane a v súčasnosti sa používa na zmenu číslovania vzletových a pristávacích dráh v prípadoch kde bol magnetický smerník dráhy zmenený vplyvom magnetickej deklinácie.

4. Záver

Jedným z problémom opísaného systému je konzistencia kvality letísk, ktorá bola riešená postupným spríšňovaním moderácie ale v prípade dôležitejších častí scenérie, ako napríklad pozícia vzletovej a pristávacej dráhy, boli použité aj iné spôsoby riešenia tohto problému. Jednou z chýb ktorej sa autor môže dopustiť je nesprávne umiestnenie posunutého prahu dráhy a ochrannej plochy proti náporu vzduchu za prúdovým motorom. Toto môže ovplyvniť prístrojové priblíženia, napríklad pri nie presných priblíženiach, kde je často „missed approach point“ v rovnakom mieste ako prah dráhy, bude tento bod v simulátore umiestnený na základe pozície prahu dráhy. Od verzie X-Plane 11.10 sú preto súradnice tohto bodu pre GPS a FMS premiestnené tak, aby boli v súlade z navigačnými dátami. Toto riešenie je však účinné len ak sa jedná o relatívne malú nepresnosť v pozícii vzhľadom na hlavnú os dráhy. Pokiaľ má používateľ aktivované anonymné odosielanie dát a zároveň aktuálne navigačné dáta bude informácia o tejto chybe zaslaná do analytického serveru „Laminar Research“ a vytvorí sa tak databáza letísk ktoré potrebujú aktualizáciu.

Aby sa zabránilo takejto chybe vôbec vzniknúť bola do programu WED pridaná validácia pozície prahu dráhy a posunutého prahu dráhy pri ktorej je pozícia porovnávaná z CIPF dátami od spoločnosti Jeppesen. Pokiaľ sa prah dráhy nenachádza v správnom mieste WED nedovolí autorovi scenériu nahráť na „Scenery Gateway“.

Pre profesionálnych používateľov, kde je presné umiestnenie dráhy absolútne nevyhnutné bol pridaný príkaz – „accurate_runways“, ktorý dynamicky prepíše scenériu po načítaní priblíženia. Dráha bude premiestnená na jej presné súradnice a taktiež sa prepíše číselné označenie dráhy. Toto riešenie funguje len pre letiská využívajúce štandardné dráhy z procedurálne generovanou textúrou.

Pod'akovanie

Článok je publikovaný ako jeden z výstupov projektu **KEGA 011ŽU-4/2018** s názvom „Nové technológie vo vzdelávaní v študijnom programe Letecká doprava a Profesionálny pilot“.

5. Referencie

- Aharrah, R., 1965. Flight simulation, past, present, and future. San Francisco, CA.
- Brezoňáková, A., Škvareková, I., Pecho, P., Davies, R., Bugaj, M., Kandra, B. 2019. The effects of back lit aircraft instrument displays on pilots fatigue and performance. In: Transportation Research Procedia 40, pp. 1273-1280.
- Cambridge University Press., 1987. Flight Simulation. Cambridge : Cambridge University Press, ISBN: 78-0521306492.

Ficová, D. Badánik, B. Novák, A. Novák Sedláčková, A. Turiak, M. 2016. Effects of automation and electronic devices on board aircraft on pilot skills, training requirements and flight safety: Zeszyty naukowe Wyższej szkoły finansów i prawa w Bielsku-Białej, ISSN 2084-1809.

Han-Wei, S., 2018. Airports Scenery Gateway. [online] Laminar Research. Available at: <<https://gateway.x-plane.com/airports/page>> [Accessed 30 November 2019].

Novák, A. a Mrázová, M. 2015. Research of physiological factors affecting pilot performance in flight simulation training device. In: Communications : scientific letters of the University of Žilina. - ISSN 1335-4205. - Vol. 17, no. 3 s. 103-107.

Novák, A., 2018. Measurement of GNSS signal interference by a flight laboratory. Elsevier, 2018. – (Transportation Research Procedia, ISSN 2352-1465 ; 35). – ISSN 2352-1465, s. 271-278 [online]

Petríková, M. Kandra, B. 2019. Vplyv záťaže pilotov na nehodovosť: Práce a štúdie, Katedra leteckej dopravy, Fakulta prevádzky a ekonomiky dopravy a spojov, Žilinská univerzita v Žiline Vol. 5. - 1. ISBN 978-80-554-1563-5

Supnik, B., 2019. X-Plane Developer Site. [online] Two-Sided Drawing and Up-Normals. Available at: <<https://developer.x-plane.com>> [Accessed 28 November 2019].