

Analiza zastosowania okularów 3D w symulacji przestrzennej na przykładzie wizualizacji architektonicznej

Mateusz Karol Smaga*, Małgorzata Plechawska-Wójcik

Politechnika Lubelska, Instytut Informatyki, Nadbystrzycka 36B, 20-618 Lublin, Polska

Streszczenie. W artykule opisane zostało przygotowanie wizualizacji architektonicznej wirtualnego muzeum. Miało to na celu zbadanie możliwości płynących z wykorzystania okularów wirtualnej rzeczywistości, a także skanów 3D oraz aplikacji do projektowania. W czasie badań okulary VR pokazały duże możliwości w zakresie przedstawiania architektury. Wyniki testów wydajności potwierdziły duży potencjał płynący z wykorzystania w wizualizacjach skanów 3D i importu modeli z zewnętrznych aplikacji, mimo wymogu ich obróbki i uproszczenia siatki. Testy wykazały jednak niewystarczającą wydajność procesora graficznego w badanym smartfonie, co pokazuje potrzebę rozwoju rynku w tym kierunku. Stworzyłyby to możliwości tworzenia lepszych aplikacji VR.

Słowa kluczowe: rzeczywistość wirtualna; unity; grafika 3D; skany 3D

*Autor do korespondencji.

Adres/adresy e-mail: mateusz.smaga@gmail.com

Analysis of the 3D glasses application in a spatial simulation on the basis of the architectural visualization

Mateusz Karol Smaga*, Małgorzata Plechawska-Wójcik

Institute of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract. The article describes preparation of architectural visualization of the virtual museum. Its purpose was to explore possibilities of using VR glasses, 3D scans and design applications. During the study VR glasses showed great opportunities in the field of architectural visualisation. The results of the performance tests confirmed potential in 3D scans and models from design applications, although they require mesh processing and simplification. However, the tests showed insufficient performance of tested smartphone's GPU, which shows that market needs to advance in this direction. This would make it possible to create better VR applications.

Keywords: virtual reality; unity; 3D graphics; 3D scans

*Corresponding author.

E-mail address/addresses: mateusz.smaga@gmail.com

1. Wprowadzenie

Temat wirtualnej rzeczywistości (ang. *Virtual reality*, VR) nie jest nowym zagadnieniem. Poczynając od stereoskopów poprzez system CAVE (ang. *Cave Automatic Virtual Environment*) [1], czyli pokoju, na którego ściany wyświetlany jest obraz z rzutników, co tworzy efekt wirtualnej przestrzeni dla osób przebywających wewnątrz pokoju, aż do coraz powszechniej znanych dzisiaj HMD (ang. *Head Mounted Display*). HMD są okularami wirtualnej rzeczywistości składającymi się z ekranu lub ekranów, na których wyświetlane są 2 obrazy odwzorowujące obraz dla poszczególnych oczu. Tworzone jest w ten sposób złudzenie przebywania w trójwymiarowej przestrzeni.

Wirtualna rzeczywistość to zagadnienie wolno rozwijające się ze względu na ograniczenia technologiczne. Sytuacja zmieniła się jednak w ostatnich latach. Na rynku pojawiły się relatywnie tanie (w porównaniu z dostępnymi profesjonalnymi rozwiązaniami) okulary VR - Oculus Rift [2], HTC Vive, jak również Playstation VR dla użytkowników konsol Sony Playstation 4. Ich odpowiednikiem na rynku mobilnym jest między innymi Samsung Gear VR i Google Cardboard. Rozwiązania te zamiast wbudowanego ekranu wykorzystują telefon użytkownika.

Mimo częstego powiązania wirtualnej rzeczywistości z grami wideo wykorzystanie okularów VR znacząco wykracza poza aplikacje typowo rozrywkowe. Jednym z takich rozwiązań jest wizualizacja architektoniczna. Zarówno architektury nam znanej, odtworzonej na podstawie realnego świata, jak i światów stworzonych zupełnie od podstaw. Jednym z wczesnych przykładów jest gra *PlastiCity* [3]. Przedstawiała ona projekt przebudowy centrum Bradford w Wielkiej Brytanii. Każdy z mieszkańców mógł zwiedzić projekt i na własne oczy zobaczyć przewidywane zmiany.

Inne przykłady to między innymi odtwarzanie stanowisk archeologicznych grobowców [4], projekt odtworzenia rzymskiego teatru w Byblos [5], rekonstrukcja pałacu Ypres Salient w Belgii na podstawie danych historycznych [6], czy sale operacyjne w Walter Reed National Military Medical Center utworzone przy pomocy wielokrotnych wirtualnych spacerów po projektowanej sali [6], gdzie po każdym teście zmieniano projekt sali według sugestii testujących, by uzyskać projekt dopasowany idealnie do potrzeb załogi szpitala.

W latach 2014-2015 L. Berg oraz J. Vance przeprowadzili szerokie badania dotyczące wykorzystania VR w przemyśle [7]. Skontaktowano i odwiedzano 18 firm zajmujących się między innymi motoryzacją, energetyką i rolnictwem.

Na miejscu okazało się, że firmy wykorzystują zarówno CAVE, HMD jak i połączenia tych technologii pozwalające na korzystanie z nich więcej niż jednej osobie na raz. Wirtualna rzeczywistość była wykorzystywana do procesu projektowania, a także rozwiązywania problemów locie. Najprzydatniejszym wykorzystaniem było badanie zakresu widzenia osób korzystających z projektowanego urządzenia w przypadkach takich jak zakres widzenia kierowcy z wnętrza samochodu, a także przeprojektowywania położenia urządzeń wewnątrz pojazdu. Przykładowo ułożenia klamek tak, by były one łatwo dostępne. Z pomocą VR prezentowane są również scenariusze. Przedstawienie zdarzeń w trójwymiarowej scenie działa na użytkowników znacznie lepiej niż pokaz slajdów. Łatwiej odnajdują się w sytuacji i szybciej pojmują prezentowane rozwiązania. Firmy stwierdziły również, że wykorzystanie wirtualnej rzeczywistości zmniejszyło koszty produkcji. Autorzy badania twierdzą, że VR nie jest rewolucją, a jednym z wielu przydatnych narzędzi w repertuarze firmy.

Nieustający rozwój technologii pozwala na zwiększenia rozdzielczości, co pozwoliłoby na zmniejszenie screen-door effect. Jest to efekt wynikający z odległości między oczami użytkownika a ekranem. Z tego względu widoczna jest siatka odstępów pomiędzy poszczególnymi pikselami. Drugą zaletą postępu technologicznego jest zmniejszenie wagi urządzeń, co znacząco wpływa na komfort ich użytkowania. Dodatkowo pojawiły się rozwiązania room-scale, czyli możliwość śledzenia ruchów użytkownika przy pomocy dodatkowych czujników [8], co umożliwia odwzorowywanie poruszania się użytkownika w świecie wirtualnym.

2. Cel i obiekt badań

Celem badania jest ocena możliwości i ograniczeń wirtualnej rzeczywistości w porównaniu z dostępnymi metodami prezentacji architektury, a także ocena przydatności tego rozwiązania na platformach mobilnych. Przed rozpoczęciem badania przedstawione zostały dwie hipotezy:

- Możliwe jest wykorzystanie skanów 3D i/lub projektów z aplikacji do projektowania do wizualizacji architektury na smartfonach.
- Wydajność smartfonu pozwala na wykorzystanie w takiej wizualizacji oświetlenia w czasie rzeczywistym.

W celu sprawdzenia tych hipotez przygotowana została wizualizacja w Unity 3D pozwalająca na poruszanie się po wirtualnym muzeum. Użytkownik może dowolnie obracać eksponaty i korzystać z interaktywnych tablic, by czytać ich opisy. Dodatkowo jeden z eksponatów można rozbić na mniejsze elementy.

Poza aplikacją wymagane są okulary. Hełm wirtualnej rzeczywistości w przypadku technologii mobilnych składa się najczęściej jedynie z obudowy, do której wkładany jest smartfon i soczewek skupiających [9]. W trakcie testów aplikacji wykorzystano kilka par okularów wirtualnej rzeczywistości.

Pierwsza para to Google Cardboard v1, czyli najprostsze i najtańsze dostępne na rynku rozwiązanie. Okulary te składają się z kartonu, przycisku magnetycznego i dwóch soczewek. Jest to minimum umożliwiające doświadczenie VR. Okulary te doczekały się następcy - Google Cardboard

v2 (Rys. 1.). Jest to ulepszona wersja wcześniej opisanego modelu. Jest łatwiejsza do złożenia, posiada większe soczewki, współpracuje z większymi telefonami. Usprawniono również działanie przycisku. W tej edycji przycisk fizycznie naciska na ekran telefonu. Zapewnia to działanie przycisku za każdym razem, co nie było prawdą w przypadku v1.



Rys. 1. Okulary Google Cardboard v2. Srebrny przycisk widoczny po lewej stronie dociska srebrny pasek pomiędzy soczewkami do ekranu. Emuluje to dotknięcie ekranu palcem i umożliwia interakcję z aplikacją.

Bardziej złożonym rozwiązaniem są na przykład okulary Fiit VR 2n (Rys. 2.). Oferują one gąbkę umożliwiającą wygodniejsze ułożenie okularów na twarzy, miejsce pozwalające na włożenie okularów VR razem z okularami korekcyjnymi, a także suwaki umożliwiające zablokowanie telefonu w jednej pozycji. Co najważniejsze dostępna jest regulacja rozstawu soczewek, co pozwala na dopasowanie ich do konkretnej i wyeliminowanie problemów ze zniekształconym lub niewyraźnym obrazem. Możliwa jest również regulacja odległości soczewek od oczu, co pozwala na zrezygnowanie z okularów korekcyjnych na rzecz korekcji w okularach. Model ten nie posiada jednak przycisków.



Rys. 2. Okulary Fiit VR.

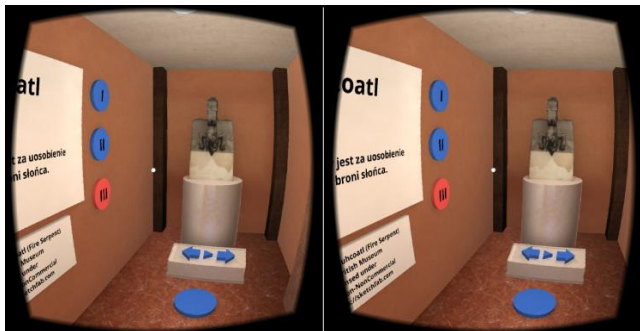
3. Przygotowanie wizualizacji

Wizualizacja została przygotowana na urządzenia mobilne z systemem operacyjnym Android. Został on wybrany ze względu na wsparcie okularów wirtualnej rzeczywistości oraz udostępnione przez Google VR SDK dla środowiska Unity. Do testów użyto smartfonu Xiaomi Redmi Note 3 (specyfikacja w Tabeli 1).

Tabela 1. Specyfikacja telefonu Xiaomi Redmi Note 3.

Procesor	Snapdragon 650 6x 1.8 GHz
Pamięć RAM	2 GB
Rozdzielczość ekranu	1920x1080 px
Wersja systemu Android	6.0.1

Google VR SDK (w wersji 1.0.3.) jest zestawem służących do integracji aplikacji przygotowanej w Unity z urządzeniami mobilnymi. Najważniejszym z nich jest gotowy prefab obsługi kamery GvrViewerMain. Generuje kamery dla lewego i prawego oka oraz dodaje obsługę ruchów głowy użytkownika (Rys. 3.).

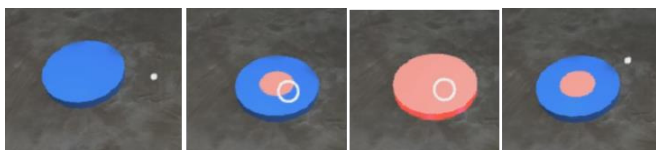


Rys. 3. Działanie skryptu generującego oddzielny obraz dla każdego oka.

Aby ułatwić interakcję przed oczami użytkownika znajduje się wskaźnik (jest on częścią Google VR SDK). Pomaga on w skupieniu wzroku na wybranym przez niego obiekcie i ułatwia orientację w przestrzeni. Gdy wzrok nie jest skierowany na obiekt, z którym użytkownik może wejść w interakcję, wskaźnik pozostaje jednobarwnym okręgiem. Jeżeli przejdzie on na obiekt interaktywny zamienia się on w obręcz. Ze względu na charakterystykę wirtualnej rzeczywistości wszystkie elementy graficznego interfejsu użytkownika odwzorowane są jako trójwymiarowe obiekty w scenie.

W ten sam sposób rozwiązana została interakcja z punktami, między którymi może poruszać się gracz. Mają one postać niebieskich, płaskich dysków. Gracz może je wskazać i, po wypełnieniu się dysku czerwonym kolorem, przemieścić się w tę pozycję. Rysunek 4 obrazuje zachowanie analogiczne do przycisków menu. Od lewej strony jest to kolejno:

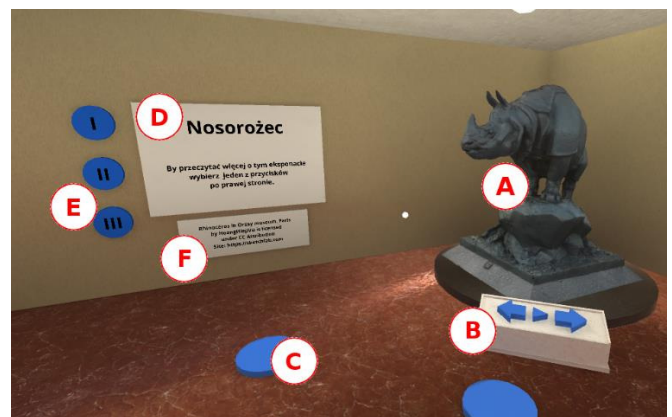
- dysk nie jest wskazywany przez gracza, zachowuje on niebieski kolor,
- gracz wskazał dysk wskaźnikiem, wskaźnik zamienił się w okrąg, nastąpiło rozpoczęcie odliczania zobrazowane rozszerzającym się, czerwonym okręgiem,
- dysk wypełnił się czerwonym kolorem, gracz został przemieszczony na jego pozycję,
- w sytuacji, gdy gracz przestał wskazywać na dysk, wypełnienie zaczyna się zmniejszać. Użytkownik ma możliwość wznowienia ładowania wskazując ponownie na dysk.



Rys. 4. Zachowanie punktów pod wpływem wskaźnika.

W każdej scenie umieszczony został licznik klatek - jego zadaniem będzie sprawdzenie wydajności aplikacji na urządzeniu mobilnym.

Do przygotowania planu muzeum i rozkładu rzeźb użyte zostały proste bryły udostępniane przez Unity. Z ich pomocą w krótkim czasie utworzony został mockup pomieszczeń. Głównym jego zadaniem było zaprojektowanie układu eksponatów pozostawiającego dużą przestrzeń na elementy sterujące oraz punkty, między którymi będzie mógł poruszać się gracz. Ich liczba powinna być jak największa, aby umożliwić graczowi jak największą swobodę poruszania się po pomieszczeniu. Po uzyskaniu satysfakcjonujących wyników układ pomieszczeń został wymodelowany w Blenderze, zaimportowany do Unity i wypełniony eksponatami.



Rys. 5. Elementy sceny.

Na scenę składają się elementy przedstawione na Rys. 5. Są to kolejno:

- **A - eksponat** - uproszczony model skanu 3D umieszczony na okrągłej podstawie,
- **B - panel** - pozwala na interakcję z rzeźbą - lewa i prawa strzałka obraca rzeźbę, dopóki wzrok pozostaje na przycisku. Przycisk w kształcie trójkąta pozwala na włączenie lub wyłączenie ciągłego obrotu,
- **C - teleport** - punkt pozwalający na przemieszczanie się użytkownika po scenie,
- **D - opis eksponatu** - tablica, na której wyświetlany jest tekst opisujący umieszczony obok eksponat, domyślnie nazwa eksponatu oraz tekst zachęcający do wybrania jednego z przycisków (oznaczonych na rysunku znakiem E),
- **E - przyciski** - pozwalają na zmianę tekstu wyświetlanego na elemencie D - użytkownik może się między nimi dowolnie przełączać przy pomocy wskaźnika,
- **F - licencja** - zawiera wszystkie informacje zawarte w wymaganiach licencji - nazwę autora, rodzaj licencji oraz stronę, z której pobrano został model.

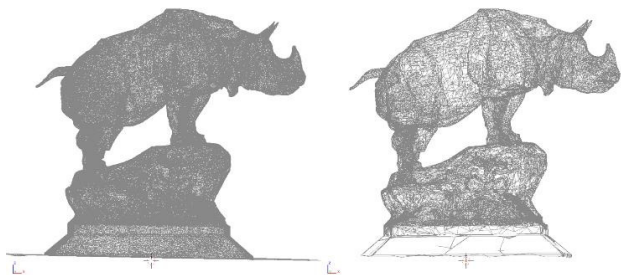
Wszystkie modele zeskanowanych rzeźb umieszczonych w scenie zostały pobrane ze strony Sketchfab. Udostępniane są one na licencji Creative Commons. Przed dodaniem ich do sceny wymagana była ich dodatkowa obróbka. Modele te charakteryzują się siatką złożoną z dużej liczby wierzchołków (od kilkudziesięciu do nawet ponad miliona). Ze względu na sprzęt obsługujący aplikację, smartfon, nie jest możliwe uzyskanie wrażenia płynności w aplikacji.

W związku z tym wymagane było znaczne uproszczenie siatki tych modeli.

Operacje związane z optymalizacją eksponatów wykonane zostały w programie graficznym Blender. Funkcja Decimate>Collapse pozwala ustalić procentową zawartość wielokątów w uproszczonym modelu względem modelu oryginalnego. Zastosowane zostało wielokrotne użycie tej funkcji z ustawieniami wartości w granicach 60-80% aż do uzyskania siatki składającej się z 20-30 tysięcy wierzchołków. Liczba wierzchołków została ustalona w trakcie badania opisanego w dalszej części artykułu. Niech za przykład posłuży statua nosorożca z Muzeum Orsay. Różnice w liczbie wierzchołków oraz ścian modelu przedstawiono w Tabeli 2, a podgląd siatki widoczny jest na Rys. 6.

Tabela 2. Porównanie modelu oryginalnego i uproszczonego.

	Model oryginalny	Model uproszczony
Liczba wierzchołków	175783	30703
Liczba trójkątów	351224	61607



Rys. 6. Porównanie siatki obu modeli. Model oryginalny (po lewej) oraz uproszczony (po prawej).

Testy wydajności przeprowadzone zostały przy pomocy aplikacji Trepp Profiler oraz przygotowanej wcześniej w Unity sceny wirtualnego muzeum. Badane miary to:

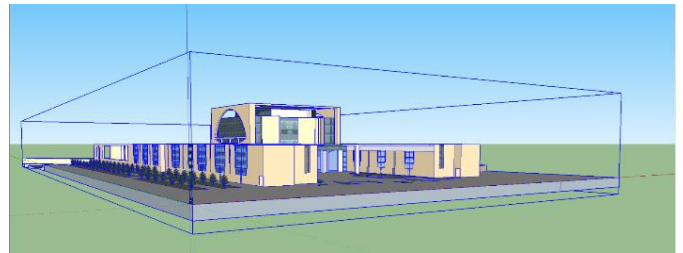
- maksymalna liczba generowanych klatek na sekundę,
- minimalna liczba generowanych klatek na sekundę,
- średnia liczba generowanych klatek na sekundę,
- średnie zużycie pamięci i zużycie pamięci w kolejnych sekundach testu,
- średnie obciążenie CPU i obciążenie CPU w kolejnych sekundach testu,
- średnie obciążenie GPU i obciążenie GPU w kolejnych sekundach testu.

Dodatkowo wykonano przegląd wybranych aplikacji umożliwiających eksport projektów jako modeli 3D. Przegląd ten miał za zadanie sprawdzenie możliwości prostego wykorzystywania przygotowanych wcześniej projektów w aplikacjach wirtualnej rzeczywistości. Umożliwia to przedstawienie modeli użytkownikom, którzy mogą zwiedzić budynek z perspektywy człowieka przechadzającego się po nim. Stwarza to niesamowite możliwości przedstawiania projektów jak i sprawdzania ich przydatności.

Do testów wykorzystane zostały dwie dostępne za darmo aplikacje do projektowania architektury: Sketchup oraz Sweet

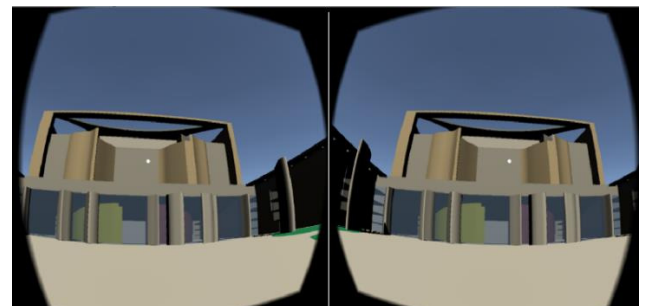
Home 3D. Poza nimi na rynku dostępne jest wiele innych rozwiązań, zarówno darmowych, jak i płatnych. Wybór aplikacji powyżej jest czysto subiektywny, mający na celu przedstawienie możliwości eksportu gotowych modeli do aplikacji wirtualnej rzeczywistości.

Pierwszy z programów, Sketchup, udostępnia dodatkowo bibliotekę udostępnionych za darmo modeli. W związku z tym do eksportu przygotowany został jeden z dostępnych w bibliotece modeli - urząd kanclerza federalnego Niemiec (Rys. 7).



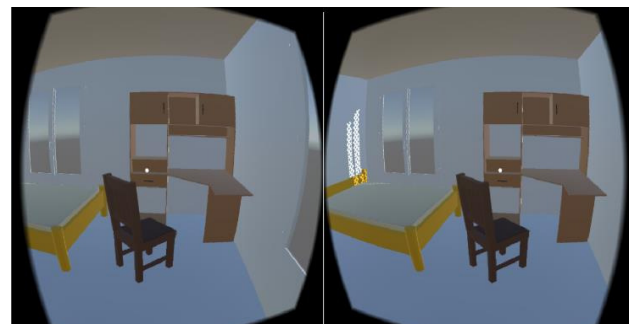
Rys. 7. Projekt w programie SketchUp.

Następnie został on wyeksportowany z programu SketchUp do formatu Collada (rozszerzenie .dae). Po bezpośrednim imporcie do Unity dostępny jest model z gotowymi już materiałami, które można modyfikować bezpośrednio wewnątrz środowiska. Po dokonaniu kilku poprawek materiałów dodano kamerę i elementy Google SDK. W ten sposób otrzymana została scena widoczna na Rys. 8.



Rys. 8. Projekt zaimportowany z programu SketchUp do sceny w Unity.

Drugi z programów, Sweet Home 3D, pozwala na planowanie pokoi przy pomocy prostych narzędzi i metody przeciągnij i upuść. Plan pokoju generowany jest jako podgląd 3D w czasie rzeczywistym. Program oferuje wyeksportowanie tak przygotowanego mieszkania do formatu .obj i następnie zaimportowanie go do Unity (Rys. 9).



Rys. 9. Projekt zaimportowany z programu Sweet Home 3D do sceny w Unity.

Analogiczny proces odbywa się w innych aplikacjach do projektowania. W ten sam sposób możliwe jest również importowanie modeli z programów takich jak AutoCAD, czy FreeCAD. Każdy z programów umożliwiających eksport do formatów takich jak .obj, czy .fbx umożliwia umieszczenie tych obiektów wewnątrz wirtualnej rzeczywistości i obejrzenie ich w trójwymiarze.

4. Przebieg testów

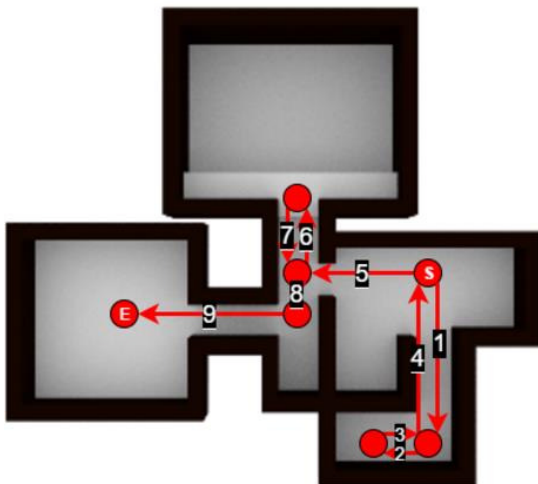
W trakcie testów wykorzystane zostały dwie metody badawcze:

- eksperyment na bazie jednego z eksponatów mający na celu dobór maksymalnej liczby wierzchołków umożliwiających płynne działanie aplikacji,
- metoda porównawcza, której zadaniem było porównanie wydajności sceny z oświetleniem obliczanym w czasie rzeczywistym ze sceną wykorzystującą lightmapy.

Pierwszy eksperyment opracowano na bazie modelu jednego z eksponatów. Przygotowana w Unity scena składała się z kilku kopii tego modelu oświetlonego czterema lampami punktowymi. Następnie model ten był stopniowo upraszczany aż do uzyskania zadowalającej wydajności (minimum 24 klatki na sekundę).

Zadaniem metody porównawczej było sprawdzenie wydajności dla dwóch wersji gotowej już aplikacji. Pierwsza, oświetlenie w czasie rzeczywistym, pozwala na uzyskanie dynamicznego światła i cieni reagujących na zmianę położenia obiektów w scenie. Druga korzystająca z lightmap, które można wygenerować w trakcie tworzenia aplikacji, co pozwala na zwiększenie wydajności aplikacji.

Kolejne kroki wykonywane podczas porównywania dwóch scen to:



Rys. 10. Kolejne kroki podczas przeprowadzania testu w muzeum.

- wczytanie aplikacji,
- wybranie muzeum z listy scen,
- odwrócenie się w prawo o 180 stopni (S),
- przemieszczenie się w kierunku interaktywnej rzeźby (1, 2),
- rzucenie rzeźbą i obserwowanie jej rozbicia,
- powrót do punktu startowego (3, 4),

- przejście do pokoju z pomnikiem, obrót rzeźby (5, 6),
- przejście do pokoju z czterema rzeźbami (7, 8, 9),
- obrót o 360 stopni,
- zakończenie badania ze wzrokiem skierowanym na tablicę wyświetlającą liczbę wyświetlanych klatek na sekundę (E).

Numery w nawiasach obrazują przemieszczanie się wzdłuż ponumerowanych strzałek zaprezentowanych na Rys. 10.

5. Wyniki testów wydajności

Wyniki wydajności dla poszczególnych szczegółowości modeli zaprezentowane zostały w Tabeli 3.

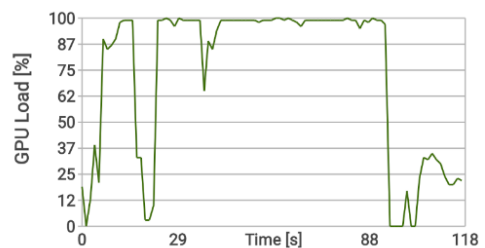
Tabela 3. Wydajność dla poszczególnych wersji modelu.

	Liczba wierzchołków	Liczba klatek na sekundę
Wersja A	190691	15
Wersja B	101420	17
Wersja C	62877	21
Wersja D	36524	24
Wersja E	22000	27
Wersja F	11807	30

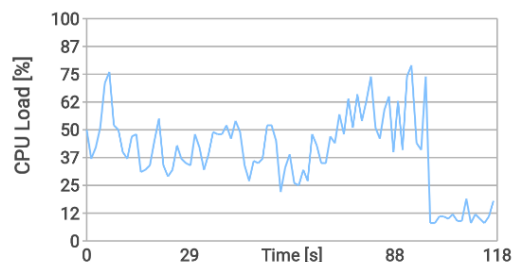
Wyniki drugiego testu przedstawione zostały w Tabeli 4. Kolumna Real-time opisuje wyniki dla sceny z oświetleniem w czasie rzeczywistym, a kolumna Lightmaps scenę z oświetleniem generowanym z użyciem lightmap. Dla drugiej sceny dodatkowo zaprezentowane zostało procentowe obciążenie GPU (Rys. 11.) i CPU (Rys. 12.).

Tabela 4. Wydajność dla wizualizacji z oświetleniem w czasie rzeczywistym oraz oświetleniem przy pomocy lightmap.

	Real-time	Lightmaps
Max FPS	28	48
Min FPS	4	28
Średnie FPS	16	40
Średnie obciążenie CPU [%]	41,28	39,04
Średnie obciążenie GPU [%]	77,31	81,92
Średnie zużycie pamięci [MB]	1855,8	1831,33



Rys. 11. Obciążenie GPU w trakcie działania aplikacji (lightmapy).



Rys. 12. Obciążenie CPU w trakcie działania aplikacji (lightmapy).

6. Opis wyników i wnioski

Wstępne testy przedstawione w Tabeli 3 pozwoliły na ocenę możliwości urządzenia i ustalenie zakresu liczby wierzchołków eksponatów. Do płynnego działania liczba wierzchołków nie powinna przekraczać 36 tysięcy dla testowanego modelu telefonu.

Następny test miał na celu porównanie wydajności aplikacji z oświetleniem generowanym w czasie rzeczywistym i aplikacji z wypalonymi lightmapami. Jak widać w Tabeli 4 oświetlenie w czasie rzeczywistym nie pozwala na osiągnięcie minimalnej liczby klatek wymaganej do płynnego wyświetlania obrazu - jest to średnio 16 klatek na sekundę. Dopiero przygotowanie lightmap pozwoliło na osiągnięcie zadowalających wyników. Średnio 40 klatek z minimalną liczbą klatek na poziomie 28. Jest to zadowalający wynik pozwalający na bezproblemową interakcję z aplikacją. W przypadku CPU obciążenie nigdy nie przekroczyło wartości 75% (Rys. 12.). Widocznym słabym punktem było jednak GPU. Często osiągało obciążenie na poziomie 100% (Rys. 11.). W związku z tym druga z hipotez postawiona przed rozpoczęciem badań jest nieprawdziwa. Wydajność testowanego urządzenia była niewystarczająca do wykorzystania w scenie z oświetleniem w czasie rzeczywistym.

Zarówno wykorzystanie skanów 3D jak i aplikacji do projektowania znacząco skraca proces tworzenia wizualizacji architektonicznych, nawet biorąc pod uwagę wymaganą wstępną obróbkę skanów. Potwierdza to pierwszą hipotezę postawioną przed rozpoczęciem badania. Dostępne na rynku rozwiązania pozwalają na relatywnie prostą obróbkę modeli do wykorzystania w aplikacji mobilnych. Pozwala to na przygotowania wizualizacji architektury bez posiadania umiejętności w zakresie modelowania 3D. Uzyskane w ten sposób obiekty pozwalają na odwzorowanie rzeczywistości w najdrobniejszych szczegółach. Największą przewagą modeli przygotowanych ręcznie jest ich optymalizacja. W przypadku skanów 3D zaobserwowana dużą liczbę wierzchołków nie wpływających znacząco na szczegółowość modelu, które mogą stanowić problem w trakcie procesu automatycznego upraszczania siatki modelu.

Wykorzystywane w badaniu okulary Cardboard umożliwiły komfortowy odbiór, ale brak konfiguracji rozstawu soczewek, dostępny w okularach FiiT VR, znacząco utrudniał uzyskanie wyraźnego obrazu jeżeli z okularów korzystała więcej niż jedna osoba. Udostępnione przez Google narzędzia do konfiguracji pozwalają na przygotowanie profilu dedykowanego dla poszczególnych użytkowników. W przypadku testowanego smartfonu o rozdzielczości 1920x1080 pikseli nadal widoczny jest jednak screen-door effect. Rozwiązaniem tego problemu jest wykorzystanie smartfonu z większą rozdzielczością.

Okulary wpływają również na decyzje podejmowane podczas projektowania aplikacji. Kwestie takie jak sposób poruszania się oraz interakcja z otoczeniem, to problemy, które trzeba rozwiązać przed rozpoczęciem tworzenia wizualizacji. HMD dostępne na komputerach stacjonarnych posiadają kontrolery odwzorowujące ruchy rąk. W przypadku rozwiązań mobilnych niektóre modele okularów nie posiadają fizycznych przycisków, lub jest to tylko jeden przycisk.

Możliwe jest użycie bezprzewodowych kontrolerów bluetooth, co może jednak ograniczyć liczbę użytkowników posiadających możliwość interakcji z aplikacją. Dlatego też w przypadku testowanej aplikacji postawiono na rozwiązania umożliwiających korzystanie z aplikacji na jak najszerzej liczbie urządzeń. Kontrola przy pomocy skupiania wzroku pozwala na korzystanie z aplikacji bez użycia fizycznych kontrolerów. Wybrany sposób poruszania się to najpopularniejsza obecnie metoda, czyli poruszanie się pomiędzy punktami. Jest to jednak rozwiązanie mało komfortowe i badane są obecnie inne rozwiązania.

Podsumowując, odwzorowanie architektury wirtualnej rzeczywistości na urządzeniach mobilnych jest jak najbardziej możliwe i ma bardzo duży potencjał. Wykorzystanie w niej skomplikowanych modeli wymaga jednak większej wydajności procesorów graficznych w smartfonach, szczególnie tych z niższej półki, które zostały wykorzystywane do testów przedstawionych w artykule. Obserwując ciągły rozwój zarówno smartfonów jak i okularów wirtualnej rzeczywistości uzyskanie satysfakcjonującej jakości obrazu jest tylko kwestią czasu.

Literatura

- [1] „The CAVE: audio visual experience automatic virtual environment,” *Communications of the ACM*, pp. 64-72., Nr 6/35.
- [2] „A review Paper on Oculus Rift – A Virtual Reality Headset,” *International Journal of Engineering Trends and Technology*, pp. 175-179., 2014.
- [3] M. Fuchs, „PlastiCity A Multiplayer Urban Planning Game,” *Space Time Play Computer games, architecture and urbanism: the next level*, pp. 370-371, 2007.
- [4] B. J. Fernández-Palacios, D. Morabito i F. Remondino, „Access to complex reality-based 3D models using virtual reality solutions,” *Journal of Cultural Heritage*, pp. 40-48, 23/2017.
- [5] „Virtual and Augmented Reality for Rich Interaction with Cultural Heritage Sites: A Case Study from the Roman Theater at Byblos,” *Digital Applications in Archaeology and Cultural Heritage*, 2017.
- [6] J. Konopacki, „The technology of augmented reality - virtual reconstructions of landscape architecture design,” *Technical Transactions*, nr 5-A, pp. 98-112, 2014.
- [7] A. Ibrahim, J. Dimick i A. Joseph, „Building a Better Operating Room,” *Annals of Surgery*, p. 35, 1/2017.
- [8] L. Berg i J. Vance, „Industry use of virtual reality in product design and manufacturing: a survey,” *Virtual Reality*, pp. 1-17, 21/2017.
- [9] S. Greenwald, Z. Wang, M. Funk i P. Maes, „Investigating Social Presence and Communication with Embodied Avatars in Room-Scale Virtual Reality,” w *Immersive Learning Network Conference*, 2017.
- [10] Z. Deng, „Design and fabrication of concave-convex lens for head mounted virtual reality 3D glasses,” w *International Conference on Optical Instruments and Technology*, 2015.