

УДК 004.946:338.48-32

Веретеннікова Н.В.

Національний університет «Львівська політехніка»

Ржеуський А.В.

Національний університет «Львівська політехніка»

Кунанець Н.Е.

Національний університет «Львівська політехніка»

Лаба Р.Ю.

Національний університет «Львівська політехніка»

ФОРМУВАННЯ ВІРТУАЛЬНИХ ТУРІВ КОРПУСАМИ НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

У публікації представлено технології створення інформаційної системи, що забезпечує інтерактивний сервіс віртуального туру. Першим об'єктом, приміщеннями якого здійснюватиметься віртуальна подорож, обрано головний корпус Національного університету «Львівська політехніка». Функціональністю інформаційної системи передбачено можливість подальшого розширення туру на інші корпуси університету. Для створення віртуального туру було використано низку засобів: HTML, сервер баз даних MySQL, мова JavaScript, бібліотека Three.js, WebGL, JSON. У статті детальніше проаналізовано можливості використання бібліотеки Three.js. під час створення інтерактивних екскурсій.

Ключові слова: віртуальний тур, HTML, сервер баз даних MySQL, JavaScript, бібліотека Three.js, WebGL, JSON.

Постановка проблеми. Використання сучасних інформаційних технологій, що дозволяють здійснювати віртуальні подорожі різними установами та місцевостями, стає незамінним у багатьох сферах діяльності людини. Галуззю, що найдинамічніше розвивається в Україні, є туризм; під час підбору поїздки кожен турист хоче бути ознайомленим з усіма умовами й прагне отримати вичерпну інформацію про місце відпочинку. Візуалізацію інформації дозволяє забезпечити технологія «віртуальний тур». Такий підхід сприяє поданню зображення об'єкта в різних ракурсах і ознайомленню з куточком суспільного простору, який викликав зацікавлення. Як зручний інструмент візуалізації технологія «віртуальний тур» ефективно використовується у сфері маркетингу, що дозволяє показати потенційному споживачу його товар або послугу, створюючи ілюзію присутності в місці перегляду, викликаючи яскраві зорові образи. Цей інструмент дозволяє отримати більш повну інформацію про об'єкт перегляду [1].

Проблему формування іміджу міста засобами 3D досліджував А. Горохів [2]. Віртуальний тур як засіб документування об'єктів культурної спадщини розглядали З. Кулик і С. Іващенко [3]. Етапи розроблення віртуальних турів, а також переваги та недо-

ліки такого мультимедійного продукту для користувача розглядають О. Цимбаліст і О. Орлик [4].

Однак актуальним і найменш розробленим питанням є створення віртуальних турів чи систем інтерактивної навігації віртуальної прогулянки закладами вищої освіти України.

Постановка завдання. Мета дослідження – аналіз проекту інтелектуальної системи для створення інформаційного продукту забезпечення інтерактивного сервісу навігації корпусами Національного університету «Львівська політехніка».

Виклад основного матеріалу дослідження. В основу методологічного дослідження покладений системний аналіз. Об'єктом дослідження є проект інтелектуальної системи «Віртуальний тур головним корпусом Національного університету «Львівська політехніка».

Для розроблення інтерактивного сервісу навігації по університету використовувалися такі засоби, як мова розмітки гіпертексту HTML, сервер баз даних MySQL, мова JavaScript, бібліотека Three.js, WebGL, JSON, а також програмне забезпечення (Google Street View, Webstorm, Brackets).

У дослідженні пропонуються засади розроблення проекту інтелектуальної системи «Віртуальний тур» із можливістю розміщення на

веб-сайті. Проект передбачає створення зручної WEB-системи, яка надаватиме користувачу повну інформацію про об'єкти, можливість оглядати приміщення та переходити з локації на локацію в межах будівлі.

На першому етапі дослідження, після проведення аналізу наявних у мережі Інтернет проектів зі створення віртуальних турів, нами визначено схему розроблюваної web-системи. Запропонована система служить для віртуальної подорожі будівлями університету. Для цього визначені основні модулі, з яких буде складатися інтелектуальна система, а саме адміністративна частина, перелік об'єктів і головна сторінка.

Головна сторінка містить лінки для переходу до модулів «Адміністративна частина» і «Перелік об'єктів».

Модуль «Адміністративна частина» забезпечує процеси додавання адміністратором нових об'єктів і редагування інформації, поданої на сайті. На сайті розміщується зображення об'єктів – корпусу Національного університету «Львівська політехніка», віртуальна подорож яким відбувається за допомогою програмного продукту.

Перелік об'єктів забезпечує можливість переходу користувача до обраного об'єкта, яким відбуватиметься подорож.

Тривимірна графіка в інтелектуальній системі реалізується за допомогою технології WebGL. WebGL – стандарт, специфікація якого постійно вдосконалюється і призначена для відображення 3D-графіки у веб-браузерах. Ця технологія дозволяє впроваджувати апаратно прискорену 3D-графіку у веб-сторінки без необхідності використовувати спеціальні плагіни веб-браузера на будь-якій платформі, що підтримує OpenGL або OpenGL ES. Таким чином здійснюється прив'язка JavaScript-скриптів до функцій, визначених у бібліотеках OpenGL ES 2.0 і реалізованих на рівні браузера. Технологія забезпечує WebGL-виконання необхідних математичних операцій (із матрицями, кватерніонами й т. д.) для створення тривимірної графіки. У якості робочого фреймворку вибрана бібліотека Three.js, яка надає можливість написання 3D-частини на JavaScript. Three.js скрипти використовуються спільно з елементом HTML5, CANVAS, SVG або WebGL.

Для коректного відображення тривимірної графіки в Three.js необхідні три речі: сцена, камера й рендерер (для візуалізації перших двох пунктів). Об'єкт THREE.Camera додається автоматично бібліотекою Three.js, коли відбувається візуалізація сцени

```
(var scene = new THREE.Scene();
var camera = new THREE.PerspectiveCamera(
90, window.innerWidth / window.innerHeight, 0.1,
1000 );
var renderer = new THREE.WebGLRenderer(,);
```

проте його можна додавати і вручну.

Перше зображення, що з'являється в 3D-просторі, – це рендерер. Рендерер – це сутність, яка відображає на canvas обрану сцену з обраної камери. Під поняттям «сцена» розумітимемо сформований нами 3D-простір, у якому розташовуються необхідні нам об'єкти: елементи (меші). Рендерер для відображення потрібного нам об'єкта на canvas завжди подає (із використанням методу подання об'єкта WebGL) сцену й камеру. Саме ці дві ключові речі формують відображення простору на тег canvas.

Зовнішній вигляд елементів залежить саме від положення камери – тієї точки, з якої відбувається знімання сцени. Представлення 3D-об'єктів на екрані – це фактично двовимірні проекції, які подаються в тривимірному просторі. Сцена може бути відображена по-різному, залежно від модифікації обраної камери. Для тривимірного подання зображення використовується три види камер:

- CubeCamera;
- OrthographicCamera;
- PerspectiveCamera.

Камера PerspectiveCamera формує ракурс перспективи, відображає сцену так, що об'єкти, які знаходяться далі, виглядають меншими. Саме такий підхід потрібен для зйомки тривимірної сцени (Рис. 1. PerspectiveCamera).

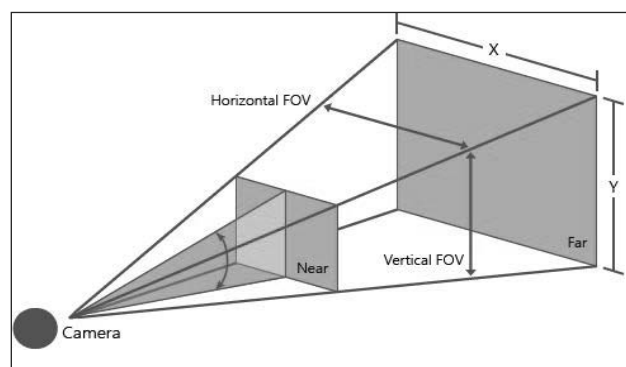


Рис. 1. PerspectiveCamera

1. Перший параметр (90) визначає вертикальне поле сцени в градусах (від нижнього до верхнього краю). Це частина сцени, яку видно на екрані в заданий момент. Горизонтальне поле огляду обчислюється з використанням вертикального поля огляду.

2. Другий параметр (`window.innerWidth/window.innerHeight`) визначає пропорції камери. Ми обрали відношення ширини вікна перегляду до його висоти для максимально точного відображення панорами на екрані.

3. Третій параметр (0.1) визначає передню площину видимого простору камери. У двовимірній системі позитивні значення осі X відкладаються справа, а осі Y – зверху вниз. Однак у тривимірній системі координат початок розташовується в центрі досліджуваної області. Позитивні значення осі X відкладаються вправо, осі Y – знизу вверху, а осі Z – із центра до спостерігача. У такому разі передня площина видимого простору практично збігається з площиною XY (тобто з екраном).

4. Останній параметр (1000) визначає задню площину видимого простору. Таким чином, коли об'єкт зміщується більш ніж на ± 1000 одиниць, він виходить за межі видимого простору Three.js і відсікається від зображення.

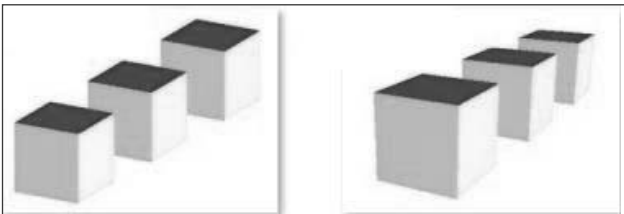


Рис. 2. OrthographicCamera й PerspectiveCamera

Геометрія простору – це набір вершин, які під час генерації з'єднуються між собою графічними примітивами. Якщо ми вважатимемо найпростішу площину геометрією простору, то в рамках цієї площини на зображенні точки з'єднуються між собою прямими лініями. Якщо є сфера, то в ній ключові точки в сегментах з'єднуються прямими лініями.

Бібліотека Three.js включає в себе 4 види геометрії площини:

- BoxGeometry;
- SphereGeometry;
- CylinderGeometry;
- TorusGeometry.

Ми обрали другий вид геометрії простору, а саме кубічну, адже для згладженого ефекту панорамної 3D-сцени вона є найбільш оптимальною для сприйняття людським оком.

Матеріали в Three.js описують зовнішній вигляд об'єктів (у нашому випадку – 3D-сцену). Матеріал – це не просто колір об'єкта, це те, як він себе поводить на сцені. Матеріалом нашої сцени є панорамні фотографії (Рис. 3), зроблені за допомогою програми Microsoft Photosynth. Усі матеріали містяться в JSON-файлі, в якому також зберігаються всі переходи між кімнатами.



Рис. 3. Панорамна фотографія

На рис. 4 можна побачити схему взаємодії основних програмних сутностей у межах нашого фреймворку.

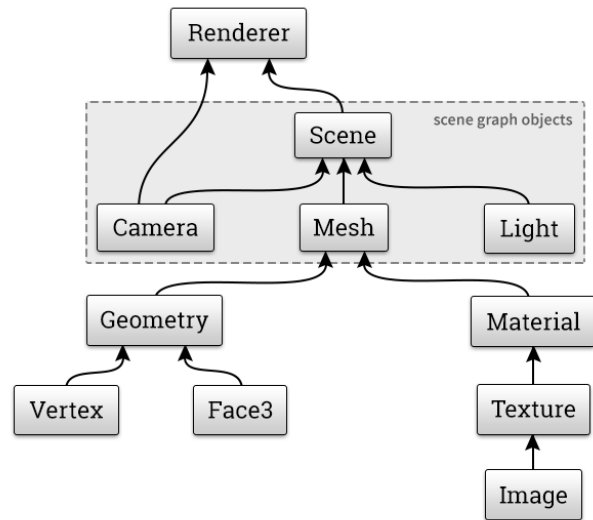


Рис. 4. Node-map

Меш – це елемент сцени, який складається з геометрії простору й матеріалу об'єкта. Отже, геометрію простору й матеріали об'єкта ми отримали в тривимірному просторі, що призвело до заповнення сітки (mesh). Після такого налаштування камера також готова для відображення інтерактивної карти. Це дозволяє додати об'єкти на сцену й отримати 3D-карту. Для кращого розуміння реалізації пропонується розглянути таку концепцію: «Ми знаходимося в центрі сфери, на внутрішню поверхню якої нанесена текстура панорами. Ми – це камера, яка може повертатися й дивитися вгору чи вниз, вліво чи вправо».

Можливість повороту камери була надана за допомогою спеціальних формул:

```
lat = Math.max(- 85, Math.min(85, lat));
phi = THREE.Math.degToRad(90 - lat);
theta = THREE.Math.degToRad(lon);
camera.target.x = Math.sin(phi) * Math.cos(theta);
camera.target.y = Math.cos(phi);
camera.target.z = Math.sin(phi) * Math.sin(theta).
```

Також надана можливість наближення та віддалення сцени (рис. 5):

```
function onDocumentMouseWheelStop() {
    if(event.wheelDeltaY && camera.fov == 30){
        camera.fov -= event.wheelDeltaY * -0.05;
    }else if(event.wheelDeltaY && camera.fov == 100){
        camera.fov -= event.wheelDeltaY * -0.05;
    }
}

function onDocumentMouseWheel(event) {
    // WebKit
    if (event.wheelDeltaY && camera.fov > 30 && camera.fov < 100) {
        camera.fov -= event.wheelDeltaY * 0.05;
        onDocumentMouseWheelStop();
    }
    // Opera / Explorer 9
    } else if (event.wheelDelta && camera.fov > 30 && camera.fov < 100) {
        camera.fov -= event.wheelDelta * 0.05;
        onDocumentMouseWheelStop();
    }
    // Firefox
    } else if (event.detail && camera.fov > 30 && camera.fov < 100) {
        camera.fov += event.detail * 0.05;
        onDocumentMouseWheelStop();
    }
    camera.updateProjectionMatrix();
}
```

Рис. 5. Функції MouseWheel

Ця робота може містити рухомі елементи всередині сцени. Тому на внутрішній поверхні сфери знаходяться спрайти. Спрайт – це рухомий елемент зображення в комп’ютерній графіці, найчастіше в растровому зображенні. Спрайт передбачає наявність декількох картинок, які поєднані в одну. Спрайти сприяють користувачу в перетягуванні сцен. Отримані рухомі елементи сцени роблять зображення зручнішим для користування.

Райкастер відповідає за проектування променя на сцену, визначення результату його зіткнення з елементами, перевірку місця курсору. Переважно він є невидимим для користувача, але його використання в поєднанні з бібліотекою three.js допомагає в такому:

- ідентифікувати об’єкт, на який навели мишку;
- дізнатися положення елемента;
- визначити, у який бік «дивиться» той чи інший об’єкт.

Освоєння цього класу скриптів відкриває величезні можливості для створення інтерактивності й отримання повної картини щодо об’єктів на створеній нами сцені.

Отже, при наведенні курсору на спрайт нами досягається певна дія, яка визначає поведінку об’єкта (принаймні зміна кольору спрайту). Ми розраховуємо перетин заданого променя з обраною геометрією простору. У такому разі ми направляємо промінь униз і знаходимо найближчий перетин курсора з платформою. Під час переведення курсору з платформи вона набуває стандартного кольору. Цей метод поданий на рис. 6.

```
var intersects = raycaster.intersectObjects(sprites);
if (intersects.length > 0) {
    if (INTERSECTED != intersects[0].object) {
        INTERSECTED = intersects[0].object;
        if (INTERSECTED) {
            INTERSECTED.material.color.setRGB(3.0, 142.0, 0.0);
        }
    }
    else {
        if (INTERSECTED) { INTERSECTED.material.color.setRGB(0, 0, 0); }
        INTERSECTED = null;
    }
}
renderer.render(scene, camera);
```

Рис. 6. Метод райкастеру

Висновки. У ході нашого дослідження розроблений проект віртуального туру корпусами Національного університету «Львівська політехніка» з використанням методу системного аналізу. Переваги, які ми отримуємо під час розроблення цього мультимедійного інформаційного продукту, такі:

- можливість інтеграції на сайт, оскільки тур буде представлений на веб-ресурсі;
- адаптивність до операційних середовищ мобільних пристроїв;
- можливість розширення функціоналу;
- можливість аудіосупроводу відеоряду під час проведення віртуального туру.

Віртуальний тур буде корисним для абітурієнтів, студентів, туристів, які зможуть ознайомитися зі структурою й архітектурними особливостями Національного університету «Львівська політехніка» в дистанційному режимі.

Список літератури:

1. Робота екскурсійного бюро. URL: https://studwood.ru/1271843/informatika/roboata_ekskursiynogo_byuro (дата звернення 26.08.2018).
2. Горохів А. Методичні положення щодо формування бренду туристичної дестинації. Проблеми сучасної економіки. 2007. № 3 (19). С. 15–25.
3. Кузик З., Іващенко С. Віртуальний тур як засіб документування об’єктів культурної спадщини. Сучасні досягнення геодезичної науки та виробництва. 2015. Вип. II (30). С. 82–86.
4. Цимбаліст О., Орлик О. Віртуальні екскурсії по містах світу. Інформатика та інформаційні технології: матеріали студ. наук. конф. (20 квітня 2015 р.): матер. конф. Одеса: ОНЕУ, 2015. С. 52–55.

ФОРМИРОВАНИЕ ВИРТУАЛЬНЫХ ТУРОВ КОРПУСАМИ НАЦИОНАЛЬНОГО УНИВЕРСИТЕТА «ЛЬВОВСКАЯ ПОЛИТЕХНИКА»

В публикации представлены технологии создания информационной системы, обеспечивающей интерактивный сервис «виртуальный тур». Первым объектом, по помещениям которого будет осуществляться виртуальное путешествие, избран главный корпус Национального университета «Львовская политехника». Функциональностью информационной системы предусмотрена возможность дальнейшего расширения тура на другие корпуса университета. Для создания виртуального тура был использован ряд средств, таких, как HTML, сервер баз данных MySQL, язык JavaScript, библиотека Three.js, WebGL, JSON. В статье подробно проанализированы возможности использования библиотеки Three.js. при создании интерактивных экскурсий.

Ключевые слова: виртуальный тур, HTML, сервер баз данных MySQL, JavaScript, библиотека Three.js, WebGL, JSON.

FORMING VIRTUAL TOURS IN THE BUILDINGS OF LVIV POLYTECHNIC NATIONAL UNIVERSITY

This publication presents technologies for creating an information system that provides an interactive virtual tour service. The first object, where the virtual tour will be done, is selected to be the main building of Lviv Polytechnic National University. The functionality of the information system provides the possibility of further expansion of the tour to other university buildings. To create a virtual tour, a number of tools were used such as: HTML, MySQL database server, JavaScript language, Three.js library, WebGL, JSON. This article analyses in more details the possibilities of using the Three.js library. when creating interactive excursions.

Key words: virtual tour, HTML, MySQL Database Server, JavaScript, Three.js Library, WebGL, JSON.