

# РАЗРАБОТКА КОНЦЕПЦИИ ИНТЕРНЕТ-ПРИЛОЖЕНИЯ ДЛЯ СОЗДАНИЯ 3D-ВИЗУАЛИЗАЦИИ ИНДИВИДУАЛЬНОГО УКРАШЕНИЯ С ПОМОЩЬЮ WEBGL И BLENDER

## DEVELOPMENT OF THE CONCEPT OF AN INTERNET APPLICATION FOR CREATING 3D VISUALIZATION OF AN INDIVIDUAL DECORATION USING WebGL AND BLENDER

**A. Murtazina  
E. Smirnov**

*Summary.* The rapid development of network technologies and three-dimensional graphics leads to continuous improvement in the provision of services. The article shows a 3D network visualization technology that does not require downloading components, WebGL, which is more important for overcoming difficulties that arise when transferring and downloading data in a network environment, as well as with visualization technology. A brief overview of the virtual three-dimensional graphics engine based on the integration of WebGL and Blender is given, the concept of using these technologies to create an individual decoration according to the user's design is proposed. The stages of plugin development are shown, the application interface is described, and Python code scripts are provided.

*Keywords:* three-dimensional graphics, PBR (physically based rendering), plugin, network, web resource, Python, VR.

**Муртазина Альфия Рустямовна**

*К.т.н., доцент, ФГБОУ ВО «Российский  
государственный университет имени А.Н. Косыгина  
(Технологии. Дизайн. Искусство)», Москва  
aly1029@yandex.ru*

**Смирнов Евгений Евгеньевич**

*К.т.н., доцент, ФГБОУ ВО «Российский  
государственный университет имени А.Н. Косыгина  
(Технологии. Дизайн. Искусство)», Москва  
evg7162@mail.ru*

*Аннотация.* Быстрое развитие сетевых технологий и трехмерной графики приводит к постоянному совершенствованию оказания услуг. В статье показана технология сетевой 3D-визуализации, не требующая загрузки компонентов, WebGL, что имеет большее значение для преодоления трудностей, появляющихся при передаче и загрузки данных в сетевой среде, а также с технологией визуализации. Кратко приведен обзор движка виртуальной трехмерной графики на основе интеграции WebGL и Blender, предложена концепция использования этих технологий для создания индивидуального украшения по дизайну пользователя. Показаны этапы разработки плагина, описан интерфейс приложения, приведены скрипты кода на языке Python.

*Ключевые слова:* трехмерная графика, PBR (physically based rendering), плагин, сеть, веб-ресурс, Python, VR.

## Введение

Традиционные технологии Web3D, используемые для реализации трехмерной визуализации, включают VRML, X3D, Java3D и т.д., которые требуют установки подключаемых модулей и компонентов загрузки. Кроме того, перечисленные технологии имеют ряд недостатков: сложная работоспособность и плохая совместимость. Таким образом, реализация среды Web3D, свободной от подключаемых модулей, остается перспективным направлением в развитии сетевого 3D. Появление спецификации WebGL обеспечивает

ускорение 3D-графики без подключаемых модулей для веб-браузеров, с хорошей совместимостью и приемлемыми эффектами рендеринга, что завоевало поддержку многих производителей браузеров. В статье приведена концепция реализации 3D-визуализации украшения по дизайну пользователя в Интернете.

## Литературный обзор

Технология 3D-визуализации может обеспечить более интуитивно понятные визуальные эффекты, но традиционная сетевая информационная система (WebGIS)

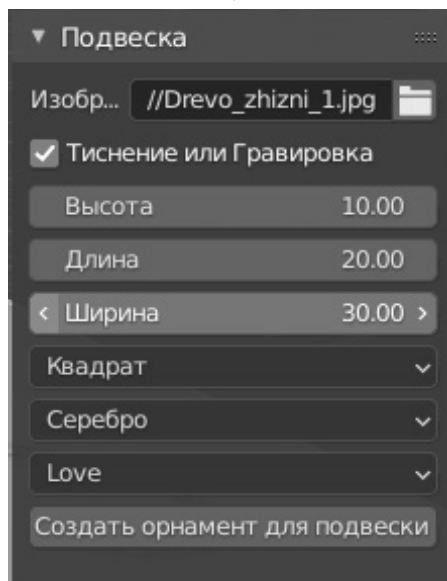


Рис. 1. Интерфейс для плагина

предназначена для разработки двумерных растровых и векторных данных. Поэтому для трехмерной графики из-за ограничений Web3D часто необходимо загружать компоненты или предварительно устанавливать плагины. На этом фоне WebGL имеет три очевидных преимущества:

- ◆ реализует создание 3D-контента с помощью сценариев HTML, при этом не требуется подключение каких-либо модулей;
- ◆ использует оборудование компьютера, обращаясь через стандартный кроссплатформенный интерфейс OpenGL[1], что позволяет ему работать на разных устройствах: персональный компьютер (ПК), смартфон, планшет;
- ◆ может быстро выполнять большое количество вычислений и выполнять рендеринг в режиме реального времени.

Использование WebGL позволяет моделировать трехмерное пространство, но конечный результат должен отображаться на двумерном дисплее. Контекстные отношения, определяемые глубиной, увеличением и уменьшением масштаба в зависимости от рассеяния, должны быть рассчитаны заранее.

Таким образом описанная технология может быть использована для интернет-приложения, позволяющего создавать индивидуальное украшение (кулон/подвеска) на заказ по предложенным рисункам. В данном случае у пользователя появляется возможность посмотреть будущую модель кулона в режиме 3D. Реализовать подобную задачу можно с помощью Blender и WebGL. Первый пакет позволит создать заготовку для изделия и выполнить визуализацию, а второй — реали-

зовать взаимодействие через сайт. Далее опишем шаги по реализации задуманной идеи в предоставлении услуги изготовления украшения.

## Материалы и методы

Кратко разработку приложения можно разделить на следующие этапы: создание концепции и прототипа интерфейса, создание 3D-модели объекта и текстуры, переход к WebGL, внедрение на сайт.

На первом этапе разработки приложения необходимо продумать концепцию: описать действия, доступные пользователю; определить функции приложения, указать принимаемые и выдаваемые форматы файлов.

Поскольку планируется предоставить пользователю возможность изменения формы кулона, то необходимо написать дополнение, которое позволит выбирать основу и загружать изображение, являющееся внутренним узором подвески. В этом случае следует рассмотреть два варианта работы с файлом: векторный и растровый форматы. В первом случае наиболее приемлемым является SVG, поскольку он универсален для Интернета, не требует установки дополнительных программ для просмотра. Кроме того, доступно множество онлайн-ресурсов, позволяющих перевести изображение в SVG, в качестве альтернативы можно воспользоваться распространенным графическим редактором. Во втором варианте необходимо выполнить векторизацию растрового изображения. Можно использовать методику, изложенную в статье [2], для получения описания точек контуров будущего изделия.

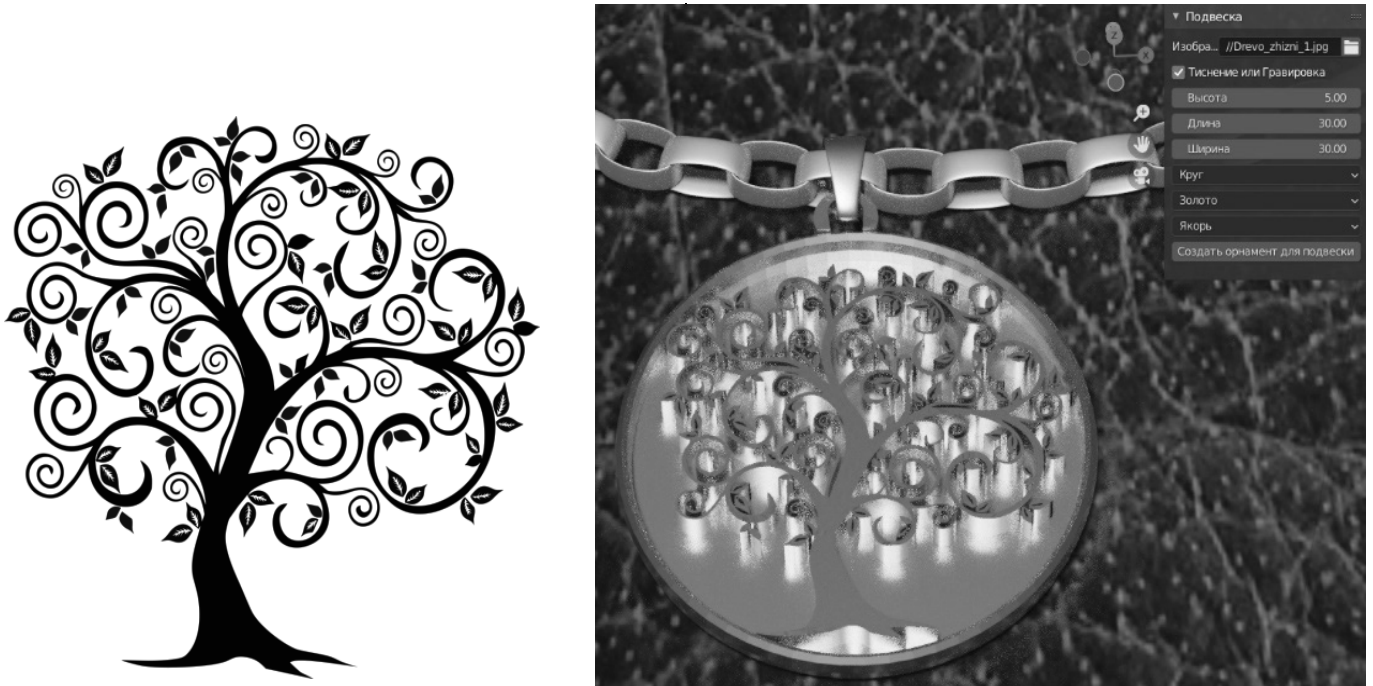


Рис. 2. Пример работы: а) исходное изображение б) визуализация

Необходимо предусмотреть два метода создания рельефа для заданного рисунка: тиснение и гравировка — оба метода используются для создания глубины в материале. Однако, в то время как стиль гравировки достигается за счет удаления незначительного количества материала, для тиснения используется штамп, который поднимает материал на заданный уровень с помощью 3D-оттиска.

Функция приложения заключается не только в демонстрации 3D-модели, но и её визуализации, наиболее приближенной к реальности. Следовательно, требуется создать библиотеку дополнительных элементов: крепление кулона, варианты типов цепочек из разных материалов. Поскольку вариаций текстур много, поэтому можно выбрать наиболее распространенные из них: серебро, золото. Пользователю необходимо предоставить выбор дополнительных элементов и материалов из имеющейся базы. Исходя из перечисленных требований и функций, был разработан интерфейс приложения, показанный на рисунке 1.

Трудоемкий этап — создание библиотеки необходимых моделей и текстур, а также написание плагина. При моделировании крепления кулона и цепочек использовались разные элементы (BezierCircle, Cube, Plane и др.) и модификаторы (Solidify, Subdivision, Array и др.). При создании текстур за основу была выбрана библиотека Free PBR Texture Library, которая использует метод физически корректной визуализации (physically

based rendering), основанный на трассировке лучей [3]. Он учитывает световые явления, такие как отражение и преломление, для создания реалистичных теней и отражений, а также значительно улучшает полупрозрачность и рассеяние, что позволяет достигнуть реалистичной визуализации. Карты текстур Free PBR Texture Library являются бесшовными, их можно настраивать в зависимости от требуемых задач, так же дополнительные плюсы — распространение под лицензией Creative Commons CC0 License (можно использовать в коммерческих целях) и регулярное обновление базы материалов.

Алгоритм работы скрипта состоит из следующих шагов: ввод информации, преобразование изображения, масштабирование, экструдирование рисунка, построение основы кулона по заданным характеристикам, применение текстур. Плагин содержит инструкции для выбора элементов, построения объектов, приведение к заданным размерам и применение материалов из библиотеки. В коде для меню приложения объявлены переменные, заданы значения по умолчанию и для габаритных размеров указаны ограничения на размер:

```
my_Height: FloatProperty(
    name = "Высота",
    description = "Укажите высоту",
    default = 20.0,
    min = 5.0,
    max = 35.0
)
```

Поскольку на размеры кулона накладываются ограничения, то необходимо выполнить проверку на возможность создания изделия по заданным характеристикам. Кроме того, следует привести к нужному размеру исходное изображение, применив операцию масштабирования.

Далее добавляют форму кулона на основе выбранных параметров, используя один из примитивов. Например, `bpy.ops.mesh.primitive_circle_add(radius=my_radius, enter_editmode=False, align='WORLD', location=(0, 0, 0), scale=(k, k, t))`. После выполняют операцию экструирования:

```
bpy.ops.mesh.extrude_region_move(MESH_OT_extrude_region={"use_normal_flip": False, "use_dissolve_ortho_edges": False, "mirror": False}, TRANSFORM_OT_translate={"value":(0, 0, my_Height), "orient_type":'GLOBAL', "orient_matrix":((1, 0, 0), (0, 1, 0), (0, 0, 1)), "orient_matrix_type":'GLOBAL', "constraint_axis":(False, False, True), "mirror": False, "use_proportional_edit": False, "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "use_proportional_connected": False, "use_proportional_projected": False, "snap": False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align": False, "snap_normal":(0, 0, 0), "gpencil_strokes": False, "cursor_transform": False, "texture_space": False, "remove_on_cancel": False, "release_confirm": False, "use_accurate": False, "use_automerge_and_split": False})
```

Для создания рельефа проверяют флаг «Тиснение или Гравировка» и, в зависимости от выбранного параметра, выполняют: смещение на указанную высоту или используют логический модификатор «Разность», вычитая из основной формы объемный рисунок, полученный путем экструирования.

На заключительном этапе построения сцены добавляют цепочку, крепление и окружение и на основе выбранных материалов добавляют текстуру к объектам (рис. 2).

При переходе к WebGL следует использовать надстройку Blender4Web, которая позволяет осуществить экспорт в формат JSON, применяемый для создания сложных проектов, требующих программирования. Заключительный этап — внедрение созданного приложения на сайт.

## Результаты и обсуждение

Приложение позволяет выбирать из библиотеки основу и материал для будущего изделия и загружать рисунок в формате SVG, являющийся основой кулона. Написанный плагин преобразует загруженный пользо-

вателем файл в 3D-модель с указанными характеристиками: габаритные размеры и материал. Использование преимуществ Blender позволяет визуализировать полученный результат, приближенный к реальности, что поможет заранее оценить вид изделия.

Эксперименты показали, что задача векторизации решается неоднозначно: результат зависит от качества исходного файла и рисунка. Поэтому следует разработать отдельный сервис для создания, редактирования и векторизации изображения. Во-первых, у пользователя будет возможность доработать исходный эскиз. Во-вторых, подобная надстройка позволит избежать ошибочного ввода и построить объект заданной формы.

Со временем, проанализировав спрос, библиотеку объектов можно развить, что позволит расширять диапазон предоставляемых услуг, например, предложить гравировку текстовых надписей или другой тип изделия: брелок, ключницу.

С точки зрения решения функциональных возможностей следует обратить внимание на перспективное направления VR и AR, которые поддерживаются Blender4Web. По словам разработчиков для использования технологий виртуальной реальности в Blender4Web не требуется вносить никаких существенных изменений в само приложение, кроме того, поддерживаются разные методики рендеринга: HMD (head-mounted display; доступна в экспериментальном режиме), SSR (Side-to-side rendering) и анаглиф. Для создания приложений дополненной реальности требуются дополнительные настройки и соответствие определенным системным требованиям: устройство с веб-камерой для запуска приложения и веб-браузер с поддержкой AR [4].

## Заключение

В связи с постоянным развитием различных программных и аппаратных технологий, статья предлагает устранить проблему установки плагинов в процессе сетевой 3D-визуализации и использовать метод рендеринга без дополнений. Трёхмерные модели с использованием технологии WebGL можно изменять в режиме реального времени с применением средств браузера, что позволит решить громоздкие этапы загрузки подключаемых модулей, а также выполнение запросов клиентов при обработке сложных изображений или трёхмерных объектов. Эксперименты позволяют сделать вывод, что WebGL подходит в качестве метода реализации внешнего интерфейса сетевой технологии. Так же перспективным является использование Blender4Web, поскольку позволяет применять технологии дополненной и виртуальной реальности.

ЛИТЕРАТУРА

1. Коршунов С.А., Дородных Н.О. Программное средство визуализации на основе библиотеки WebGL. Информационные и математические технологии в науке и управлении. 2016. № 2. С. 129–136.
2. Муртазина А.Р., Гусев А.О., Костылева В.В., Разин И.Б. Векторизация чертежей конструкций верха обуви с использованием открытых библиотек. Дизайн и технологии. 2019. № 74 (116). С. 135–141.
3. Peddie J. (2019) Ray-Tracing Programs and Plug-ins. In: Ray Tracing: A Tool for All. Springer, Cham. DOI: 10.1007/978-3-030-17490-3\_8. ISBN978-3-030-17489-7.
4. Электронный ресурс: Blend4Web. Руководство пользователя v18.05.0. Триумф. Обновлено: 2019-04-16. — URL: [https://www.blend4web.com/doc/ru/vr\\_and\\_ar.html](https://www.blend4web.com/doc/ru/vr_and_ar.html). Дата обращения 14.11.2021.

© Муртазина Альфия Рустямовна (aly1029@yandex.ru), Смирнов Евгений Евгеньевич (evg7162@mail.ru).  
Журнал «Современная наука: актуальные проблемы теории и практики»

