

КОМПЬЮТЕРНАЯ ГРАФИКА И ВИЗУАЛИЗАЦИЯ

УДК 004.92

АДАПТИВНЫЕ ИНСТРУМЕНТЫ НАУЧНОЙ ВИЗУАЛИЗАЦИИ ДЛЯ УМНОГО ПАЛЕОНТОЛОГИЧЕСКОГО МУЗЕЯ

© 2019 г. К. В. Рябинин^{a,*}, М. А. Колесник^{b,**}

^a Федеральное государственное бюджетное образовательное учреждение высшего образования
“Пермский государственный национальный исследовательский университет”
614990 Пермь, ул. Букирева, д. 15, Россия

^b Государственное краевое бюджетное учреждение культуры “Пермский краеведческий музей”
614000 Пермь, ул. Монастырская, д. 11, Россия

* e-mail: kostya.ryabinin@gmail.com

** e-mail: kolesnik.ma@outlook.com

Поступила в редакцию 16.02.2019 г.

После доработки 16.02.2019 г.

Принята к публикации 18.03.2019 г.

Статья посвящена разработке инструментов, позволяющих музейным сотрудникам создавать интерактивные экспонаты с применением технологий Интернета вещей. Предлагается унифицированное программно-аппаратное решение, позволяющее пользователям, не обладающим глубокими познаниями в области электроники и программирования, на принципах конструктора собирать и связывать друг с другом различные устройства. Это, в свою очередь, обеспечивает возможность реализовать концепцию “умного музея” без привлечения сторонних ИТ-специалистов и крупных капиталовложений. Аппаратная часть предлагаемого решения базируется на программируемом микроконтроллере ESP8266 со встроенным WiFi-модулем. Периферические устройства могут быть подключены к этому микроконтроллеру на разъемных соединениях в виде плат расширения, т. н. “шилдов”. Программная часть решения основывается на адаптивной мультиплатформенной системе научной визуализации SciVi. Поведение этой системы полностью управляется онтологической базой знаний, а для пользователя предоставляется высокоуровневый графический интерфейс, позволяющий в виде диаграммы потока данных задавать необходимые алгоритмы визуализации. В рамках данной работы в состав системы SciVi включен механизм, позволяющий генерировать прошивки для устройств Интернета вещей. В его основе лежит онтология электронных компонентов, описывающая устройства и методы их программирования. Генератор кода прошивок, управляемый этой онтологией, автоматизирует создание легких встраиваемых копий системы SciVi, которые устанавливаются на IoT устройства и выступают в роли серверов сбора, обработки и визуализации данных. Предложенное решение было протестировано на практике при создании интерактивного экспоната раннепермского звероящера *Dimetrodon grandis*.

DOI: 10.1134/S013234741904006X

1. ВВЕДЕНИЕ

Предназначение современных музеев уходит далеко за пределы простого экспонирования предметов культурного, научного и природного наследия. На сегодняшний день музеи — это полноценные научно-просветительские площадки, оборудованные с использованием новейших достижений в области техники и дизайна. Но для поддержания такого статуса они должны идти в ногу с развитием тех научных или культурных областей, которым они посвящены, предоставляя своим посетителям актуальные сведения. Для этого у сотрудников музеев должны быть в наличии удобные инструменты для быстрого обновле-

ния экспозиций с целью своевременного отражения научных и культурных новшеств.

Особенно важны эти требования для естественно-научных музеев, в частности — для палеонтологических. Современная палеонтология динамично развивается, позволяя не только составить представление о биосфере далекого прошлого, но и ответить на многие вопросы относительно нашего настоящего и потенциального будущего. Понимание того, как возникла и развивалась жизнь на Земле, каковы механизмы эволюции и каковы основные вехи развития биосферы на пути к современному биоразнообразию — одни из ключевых аспектов формирования научного мировоззрения, важного не только для профессиональных ученых, но и для обывателей. В кинофильмах

и видеоиграх существует тенденция изображать доисторическую эпоху как мир непрерывной физической борьбы за выживание. Медийный образ оказывается более ярким, чем реальность, что вызывает желание подменить объективные факты эмоционально окрашенным, но недостоверным, вымыслом. Однобокие и во многом ложные представления об эволюции и биоразнообразии добавляют изрядную долю мифичности в научную картину мира. Это, в свою очередь, приводит к серьезным мировоззренческим проблемам, когда обыватели, несмотря на свой уровень образования, оказываются неспособны отличить доказательную науку от вымысла, оказываясь в плену у предрассудков и спекуляций и, как следствие, становясь уязвимыми для разнообразных мошенников и мифотворцев.

Возможным решением такой проблемы может быть разработка высококачественного научно-просветительского контента для палеонтологических музеев, который сделал бы их экспонаты привлекательными и запоминающимися, но при этом сохранил их соответствие актуальным научным данным. Для этой цели, наряду с традиционными методами создания экспонатов (реконструкции, анатомические отливки, иллюстрации), предлагается использовать мультимедийные возможности современных технических средств, а именно — методы и средства научной визуализации, объединенные с возможностями Интернета вещей (англ. *Internet of Things*, IoT) [1]. Технологии Интернета вещей способны обеспечить высокую интерактивность экспонатов, научная визуализация — высококачественное и научно-точное наглядное представление сопутствующих данных.

Бурное развитие технологий Интернета вещей, связанное, в частности, с удешевлением программируемых микроконтроллеров, сенсоров, исполнительных устройств и мобильных источников питания, закономерно привело к возникновению концепции “умного музея” (англ. *Smart Museum*) [2], предполагающей использование этих технологий для целей мониторинга активности посетителей [3], навигации внутри музея [4] и создания интерактивных экспонатов. Однако из-за того, что большинство музейных сотрудников, ответственных за экспозиции, не являются профессиональными программистами или инженерами, возникает задача создания эргономичных высокоуровневых программно-аппаратных решений в данной области.

2. КОНЦЕПЦИЯ ИНТЕРАКТИВНЫХ ПАЛЕОНТОЛОГИЧЕСКИХ ЭКСПОНАТОВ НА ОСНОВЕ ТЕХНОЛОГИЙ ИНТЕРНЕТА ВЕЩЕЙ

В результате предыдущих исследований была спроектирована и реализована адаптивная мультимедийная система научной визуализации SciVi [5], функционирующая на принципах онтологического инжиниринга. Ее поведение полностью управляется онтологической базой знаний, в которой описаны поддерживаемые типы входных данных, способы их обработки и рендеринга. Благодаря этому SciVi может быть адаптирована к решению новых задач визуализации в произвольной предметной области и к работе с произвольными входными данными исключительно путем пополнения соответствующих онтологий, без переписывания исходного кода ядра.

Онтологии хранятся в базе в стандартном представлении с использованием языка OWL [6]. Для их изменения и пополнения может быть использован любой поддерживающий этот язык редактор; в наших исследованиях мы используем редактор ОНТОЛИС [7], так как он представляется нам наиболее эргономичным среди доступных на сегодняшний день.

Настройка системы SciVi на решение конкретной задачи научной визуализации осуществляется при помощи графического редактора диаграмм потока данных [8]. Этот редактор позволяет пользователю описать алгоритм обработки и визуализации входных данных путем создания диаграммы, вершинами в которой выступают доступные в системе операторы преобразования и рендеринга данных, а связями — пути передачи информации между операторами. Набор доступных в редакторе операторов автоматически формируется на основе обработки онтологий из базы знаний SciVi и выдается пользователю в виде палитры инструментов. Такой подход к настройке графического конвейера обеспечивает высокую гибкость описания требуемых алгоритмов, высокую наглядность и легкость в освоении.

Настройка системы SciVi на решение конкретной задачи научной визуализации осуществляется при помощи графического редактора диаграмм потока данных [8]. Этот редактор позволяет пользователю описать алгоритм обработки и визуализации входных данных путем создания диаграммы, вершинами в которой выступают доступные в системе операторы преобразования и рендеринга данных, а связями — пути передачи информации между операторами. Набор доступных в редакторе операторов автоматически формируется на основе обработки онтологий из базы знаний SciVi и выдается пользователю в виде палитры инструментов. Такой подход к настройке графического конвейера обеспечивает высокую гибкость описания требуемых алгоритмов, высокую наглядность и легкость в освоении.

В ходе дальнейших исследований база знаний системы SciVi была пополнена онтологией электронных компонентов и фрагментов прошивок для них, а в ядро SciVi добавлен управляемый этой онтологией модуль генерации программного кода для микроконтроллеров. В результате система SciVi приобрела возможность в автоматизированном режиме создавать свои легковесные копии, пригодные для установки на IoT-устройства [9]. Такие копии, называемые SciViES (англ. *SciVi Embedded System* — встраиваемая система SciVi), как правило включают следующие компоненты:

1. Код устройства-хозяина (англ. *Device Host Code*): часть прошивки устройства, отвечающая за основную логику его функционирования, такую как опрос сенсоров, управление актуаторами и сопутствующие вычисления.

2. Код сервера визуализации (англ. *Visualization Server Code*): часть прошивки устройства, реализующая сервер (обычно, HTTP- и WebSocket-сервер, но возможно использование и других комму-

никационных протоколов), который отвечает за передачу кода рендеринга и сопутствующих данных клиентам.

3. Код клиента визуализации (англ. *Visualization Client Code*): код, реализующий логику рендеринга данных и отображения результата (как правило, используется HTML5 и JavaScript). В случае, когда в роли аппаратной части сервера выступает IoT-устройство, клиент визуализации как правило оказывается более производительным (даже если это смартфон), и рендеринг целиком происходит на его стороне.

Для создания интерактивных палеонтологических экспонатов предлагается использовать микроконтроллеры ESP8266¹ со встроенной поддержкой коммуникации по WiFi. Эти микроконтроллеры завоевали большую популярность в контексте сборки легких IoT-устройств, так как продаются по низкой цене (около \$4 при заказе из Китая), но при этом обладают достаточно высокой производительностью (тактовая частота 80 МГц, объем ОЗУ 80 Кб) и малым энергопотреблением (работают при напряжении 3.3 В и силе тока 80 мА). Помимо ESP8266 могут быть использованы и другие микроконтроллеры, например, Arduino — при условии внесения соответствующих знаний о них в онтологию электронных компонентов системы SciVi.

Для удобства сборки и прошивки итоговых устройств предлагается использовать плату Wemos D1 mini², включающую в себя ESP8266 и необходимые для его работы электрические цепи (например, контур питания и сброса), а также интерфейс USB. Для присоединения дополнительных устройств, например разнообразных датчиков, предлагается использовать идею т.н. *шилдов* (англ. *Shields*) [10] — плат расширения, подключающихся при помощи разъемных соединений. Шилды можно рассматривать как своего рода кирпичики электронного конструктора, комбинациями которых можно быстро и без дополнительного оборудования (паяльников, проводов, крепежа) собирать различные устройства.

Логическое подключение шилдов к контроллеру удобно организовывать при помощи шинного интерфейса, такого как I²C, SPI, OneWire и т.п. В контексте использования микроконтроллера ESP8266 предпочтительнее оказывается использовать I²C. Во-первых, для коммуникации ведущего устройства (англ. *Master*) с несколькими ведомыми (англ. *Slave*) в нем используется ровно 2 сигнальных линии, тогда как SPI требует наличия линий-селекторов. При условии большого числа периферических устройств, на селекторные

линии может физически не хватить выводов микроконтроллера, а включение сдвиговых регистров для мультиплексирования сделает общую схему устройства более громоздкой и трудоемкой в сборке. Во-вторых, I²C, в отличие от OneWire, обеспечивает достаточно высокую скорость передачи данных и поддерживает полудуплексный режим. В-третьих, I²C поддерживают многие фабричные датчики, на базе которых можно создавать шилды.

Недостатком ESP8266 является наличие лишь одного канала аналого-цифрового преобразователя (АЦП). При необходимости подключения более чем одного аналогового датчика должен быть использован либо внешний мультиплексор, либо “прокси-контроллер”, имеющий большее число каналов АЦП. В роли прокси-контроллеров предлагается использовать чипы семейства ATtiny³, отличающиеся низкой стоимостью, но при этом предоставляющие достаточно много (в зависимости от модели — до 28) каналов АЦП. Прокси-контроллер может быть подключен к ESP8266 как ведомое устройство по шине I²C.

Идея шилдов, как набора готовых модулей, хорошо ложится и на онтологическую организацию системы SciVi: знания о их назначении, особенностях реализации и программирования могут быть заранее описаны в онтологии электронных компонентов, и пользователь сможет составлять алгоритмы соответствующих прошивок, пользуясь готовыми блоками, соединяя их в диаграмме потока данных. За счет этого музейные сотрудники, даже не обладая глубокими знаниями в области электроники и программирования, смогут на принципах конструктора создавать прототипы собственных интерактивных стендов.

3. РЕАЛИЗАЦИЯ ИНТЕРАКТИВНОГО ЭКСПОНАТА *DIMETRODON GRANDIS*

Новые палеонтологические находки и новые технологии изучения образцов (например, магнитно-резонансная томография), использование методов сравнительно-генетического анализа обуславливают быстрое развитие знаний об облике и особенностях доисторической жизни. Так, например, в 1960-х годах новые интерпретации палеонтологических фактов и окаменелостей позволили сформировать совершенно новый взгляд на динозавров (т.н. “ренессанс динозавров” [11]): удалось установить, что некоторые из них были теплокровными, покрытыми перьями животными со сложными паттернами поведения.

Однако медийный образ ископаемых животных сильно отстает от актуальных научных знаний. В то время как профессиональные палеонтологи следят за развитием области и произво-

¹ <https://www.espressif.com/en/products/hardware/esp8266ex/overview>

² https://wiki.wemos.cc/products:d1:d1_mini

³ <https://www.microchip.com/design-centers/8-bit/avr-mcus>



Рис. 1. Результат визуализации 3D-модели звероящера *Dimetrodon grandis*.

дят научно-точный визуальный контент [12], объем устаревших изображений все еще очень велик и, что хуже всего, продолжает активно использоваться как в массовой культуре, так и в научно-популярных источниках.

Новые интерпретации касаются не только динозавров, но и других представителей доисторической фауны, например, звероящеров, ранних представителей группы Synapsida. Однако информации об этих животных значительно меньше, а в музеях и популярной литературе зачастую все еще представлены устаревшие реконструкции. Традиционно звероящеры считались более близкими к рептилиям, однако современные изыскания позволяют приблизить их к млекопитающим.

Так, например, работа С. Хартмана [13] описывает новую точку зрения на анатомическое устройство *Dimetrodon grandis* Romer, Price, 1940, раннепермского звероящера, жившего более 290 млн лет назад. Отличительные особенности реконструкции диметродона Хартмана состоят в новой форме спинного паруса, учете изгибов позвоночника и выпрямленных при ходьбе лапах. При высокой правдоподобности реконструкции и доказательности исследования Хартмана, визуальных представлений диметродона, сделанных на основе этой реконструкции, очень мало. В особенности это касается трехмерных цифровых моделей: практически единственный образец научно-достоверной 3D-модели *Dimetrodon grandis* доступен в сети Интернет на коммерческой основе⁴, однако даже эта модель не представляет диметродона как прото-млекопитающее: текстура и рельеф кожи сделаны по образу рептилий, распределение кожи на парусе не соответствует данным, полученным при исследовании музейных образцов. Более того, воспроизведенная в модели походка не соответствует известному эволюционному тренду звероящеров: задние лапы сделаны не прямыми.

⁴ <https://sketchfab.com/models/21c7948d4d1e4c219ff6af19f71c8088>

Чтобы по возможности восполнить пробел в визуальном контенте, в рамках данной работы было решено создать отвечающую наиболее современным палеонтологическим данным трехмерную модель *Dimetrodon grandis* и сделать на ее основе интерактивный музейный экспонат. В экспонате задействованы IoT-устройство, отвечающее за организацию интерактивности и выполняющее функцию сервера визуализации, а также клиент визуализации, отвечающий за отображение визуального контента.

3.1. Цифровая реконструкция

Трехмерная цифровая реконструкция звероящера *Dimetrodon grandis* выполнена в соответствии с анатомическими особенностями, отмеченными в исследовании С. Хартмана. Для воссоздания текстуры использовались сведения, приведенные в книге “Все минувшие дни” Дж. Конвея и С.М. Косемена [14], а также изображение Дж. Конвея⁵. Для уточнения анатомических деталей, не отмеченных в исследовании Хартмана, использовалась высокоточная анатомическая отливка из музея Пермских древностей (г. Пермь, филиал Пермского краеведческого музея). К таким деталям относится, например, структура пальцев на лапах звероящера.

Модель создана в редакторе Blender, содержит около 3000 вершин и 5000 треугольников (рис. 1). Для передачи движений используется скелетная анимация на основе 33 костей. Материал поверхности включает в себя текстуру цвета и карту нормалей. Для использования в условиях интерактивного стенда модель экспортирована в стандартный формат COLLADA.

Визуализация 3D-модели осуществляется специальным модулем, сгенерированным системой SciVi. Модуль реализован на языке JavaScript (представляет собой HTML-страницу) и функционирует на основе графического движка Three.js⁶, выполняющий рендеринг в Web-браузере при помощи API WebGL. HTTP-сервером, отдающим эту страницу, выступает IoT-устройство, разворачивающее сеть WiFi. Клиентом визуализации может быть как стационарный компьютер, установленный в музее, так и личное мобильное устройство посетителя.

3.2. Интерактивность

В интерактивном режиме представлен палеонтологический факт о том, что диметродон использовал свой спинной парус в качестве “солнечной батареи”, разворачивая его к солнечному свету и, тем самым, нагревая свое тело с целью повышения скорости обмена веществ.

⁵ <http://johnconway.co/high-walkin>

⁶ <https://threejs.org/>

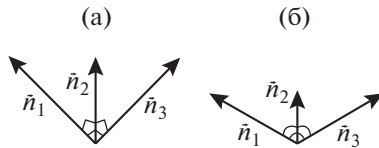


Рис. 2. Исходный (а) и масштабированный (б) базис устройства для разложения вектора направления к источнику света.

Для 3D-модели диметродона организована анимация поворота, а IoT-устройство, лежащее в основе экспоната, снабжено датчиком направления света [15]. Датчик позволяет синхронизировать реальное освещение стенда и освещение на виртуальной сцене. Диметродон автоматически разворачивается так, чтобы на его парус падало как можно больше света. Посетитель музея, освещая датчик фонариком, может наблюдать реакцию диметродона, отображаемого на мониторе компьютера, проекционном экране или дисплее мобильного устройства.

3.3. Датчик направления света

Датчик направления света функционирует на основе трех фоторезисторов VT90N2, подключенных к микроконтроллеру ESP8266 в виде шилда. Фоторезисторы расположены в вершинах равностороннего треугольника, их фотопринимающие пластины ориентированы под углом 45° к плоскости стола, на котором размещается датчик. Пик диаграммы направленности каждого фоторезистора совпадает с направлением нормали к его фотопринимающей пластине. Соответственно,

но, вектор направления к источнику света может быть выражен как $\vec{l} = a_1\vec{n}_1 + a_2\vec{n}_2 + a_3\vec{n}_3$, где $\vec{n}_1, \vec{n}_2, \vec{n}_3$ — нормали к фотопринимающим пластинам фоторезисторов, a_1, a_2, a_3 — коэффициенты, прямо пропорциональные напряжению на фоторезисторах.

Вектор направления к источнику света передается IoT-устройством клиенту визуализации посредством протокола WebSocket по WiFi. Данный протокол обеспечивает быструю и надежную доставку данных, наилучшим образом подходя для использования в ситуации, когда клиент визуализации реализован на языке JavaScript.

Базис, образованный нормальными к фотопринимающим поверхностям фоторезисторов датчика направления света, показан на рис. 2а (т.н. базис устройства). Проблема заключается в том, что коэффициенты a_1, a_2, a_3 всегда положительны по своей природе (вычисляются на основе величин потенциалов, измеренных АЦП на фоторезисторах). Это означает, что свет может быть определен только в первом квадранте базиса устройства. На соответствующей виртуальной сцене угол падения света на 3D-модель в таком случае может изменяться только в диапазоне от 45° до 90° к горизонту. С целью увеличения визуального качества результата этот диапазон искусственно расширяется путем сжатия базиса по вертикали (масштабирования в направлении $\vec{v} = \{1, 1, 1\}$), как показано на рис. 2б.

Для правильной работы датчик направления света требует калибровки трех параметров:

1. Матрица трансформации M , преобразующая систему координат IoT-устройства в систему ко-

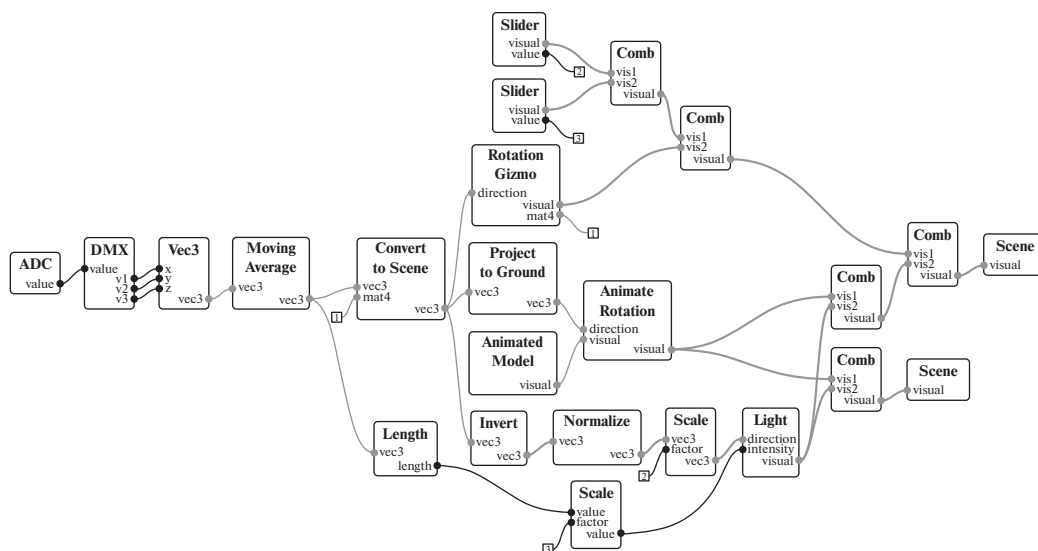


Рис. 3. Диаграмма потока данных, по которой осуществляется генерация прошивки для интерактивного экспоната *Dimetrodon grandis*.

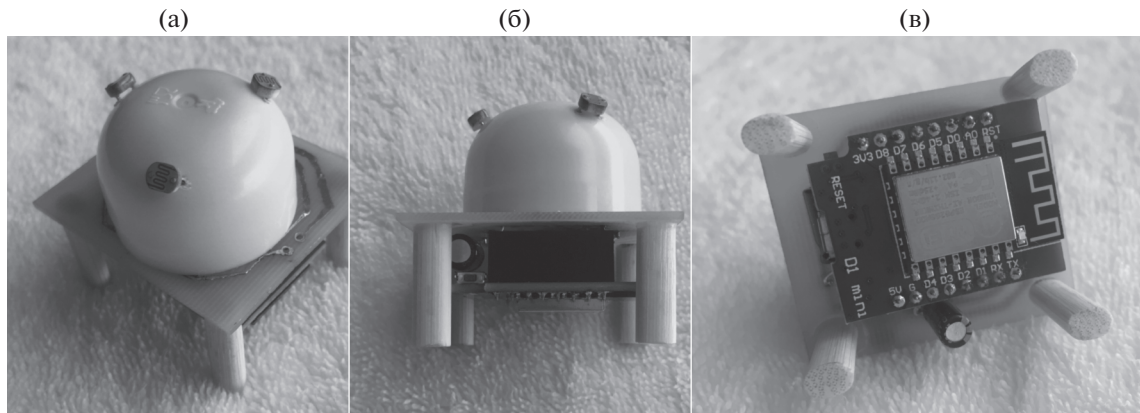


Рис. 4. Фотографии датчика направления света: вид сверху (а), вид сбоку (б) и вид снизу (в).

ординат виртуальной сцены. Эта матрица нужна для учета ориентации (поворота) устройства относительно монитора, на котором отображается 3D-модель.

2. Коэффициент вертикального масштабирования базиса s_v IoT-устройства, служащий для искусственного заострения детектируемых углов падения света, как отмечалось выше.

3. Коэффициент усиления датчика b , служащий для корректировки интенсивности света. Настройка этого коэффициента необходима для учета окружающей засветки и адаптации работы прибора к разным условиям фоновой освещенности.

3.4. Прошивка IoT-устройства

Прошивка IoT-устройства, сгенерированная по высокоуровневому описанию системой научной визуализации SciVi, включает в себя код устройства-хозяина, сервера и клиента визуализации. Алгоритмы, реализованные в этом коде, рассмотрены ниже.

Концептуальный алгоритм устройства-хозяина

1. Инициализировать калибровочные параметры:

- (a) M = единичная матрица;
- (b) $s_v = 1$;
- (c) $b = 1$.

2. Получить и сохранить параметры M , s_v и b от клиента (если предоставлены) через WebSocket.

3. Измерить потенциалы на 3 фоторезисторах, используя АЦП (встроенный в ESP8266 на аппаратном уровне).

4. Собрать вектор $\vec{l}$ с использованием значений, полученных из измерений, объединить его с калибровочными параметрами и отправить результирующее сообщение клиенту по протоколу WebSocket.

Концептуальный алгоритм сервера визуализации

1. Запустить HTTP-сервер для предоставления клиенту по запросу HTML-страниц и соответствующего мультимедийного контента (3D-моделей и текстур).

2. Запустить WebSocket-сервер для передачи клиенту данных от устройства-хозяина и получения от клиента калибровочных параметров.

Концептуальный алгоритм клиента визуализации

1. Получить по протоколу WebSocket от устройства-хозяина направление $\vec{l}$ к источнику света.

2. Получить калибровочные параметры M , s_v и b .

3. Предоставить пользовательский интерфейс для изменения параметров M , s_v и b . Матрица M изменяется посредством вращения сцены специальным элементом управления⁷, остальные параметры изменяются слайдерами.

4. Отправить параметры M , s_v и b по WebSocket обратно устройству-хозяину, если они были изменены.

5. Вычислить направление к источнику света $\vec{d}_L$ скользящим средним для вектора $\vec{l}$. Скользящее среднее позволяет устранить случайный шум, неизбежно присутствующий в измерениях.

6. Вычислить интенсивность света $e = b|\vec{d}_L|$.

7. Трансформировать сцену при помощи матрицы M .

8. Построить кватернион q_L на основе кратчайшей дуги между векторами $\vec{v} = \{1, 1, 1\}$ и $\vec{d}_L$. Этот кватернион выражает наклон лучей света к горизонту.

9. Вычислить вектор $\vec{p}_L$ как результат поворота вертикального вектора $\vec{v}' = \{0, 1, 0\}$ кватернионом

⁷ <https://knowledge.autodesk.com/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/AutoCAD-Core/files/GUID-7BD066C9-31BA-4D47-8064-2F9CF268-FA15-htm.html>

q_L . Полученный вектор выражает направление к источнику света в системе координат сцены.

10. Масштабировать координату Y вектора $\vec{p}_L$ с коэффициентом s_v , чтобы сделать угол падения света более острым (что увеличивает визуальное качество результата рендеринга).

11. Нормировать вектор $\vec{p}_L$ и умножить его на $3r$, где r — радиус отображаемой 3D-модели (радиус сферы, описанной вокруг модели). Этот шаг нужен для размещения источника света на константном расстоянии от модели, что позволяет избавиться от потенциального дрожания тени и прочих артефактов освещения. Константа 3 подобрана экспериментально и не требует модификации, поэтому в набор калибровочных параметров не включена.

12. Разместить виртуальный источник света в точке, координаты которой задает вектор $\vec{p}_L$, и выставить ему интенсивность e .

13. Вычислить вектор $\vec{g}_L$ занулением координаты Y вектора $\vec{p}_L$ и нормированием результата. Полученный вектор выражает проекцию вектора направления к источнику света на горизонтальную плоскость. Последующие шаги реализуют следующую идею: диметродон должен развернуть свой парус к свету, то есть 3D-модель должна с анимацией развернуться так, чтобы плоскость паруса стала перпендикулярна вектору $\vec{g}_L$.

14. 3D-модель диметродона выровнена по осям координат так, что плоскость паруса совпадает с плоскостью YZ . Эта плоскость имеет два перпендикуляра: $\vec{p}_1 = \{1, 0, 0\}$ и $\vec{p}_2 = \{-1, 0, 0\}$. Вычислить векторы $\vec{r}_1$ и $\vec{r}_2$ путем поворота векторов $\vec{p}_1$ и $\vec{p}_2$ кватернионом q_m , выражающим текущую ориентацию модели.

15. Вычислить два угла: $a_1 = \widehat{\vec{r}_1 \vec{g}_L}$ и $a_2 = \widehat{\vec{r}_2 \vec{g}_L}$.

16. Найти минимальный угол $a = \min\{a_1, a_2\}$, сохранить соответствующий ему повернутый перпендикуляр как $\vec{r}$ и исходный перпендикуляр как $\vec{p}$.

17. Найти ось поворота $A = \vec{r} \times \vec{g}_L$. Вокруг этой оси должна быть повернута 3D-модель, чтобы “подставить парус свету”.

18. Вычислить шаг поворота r_s : угол, на который должна повернуться модель в текущем кадре, учитывая скорость обновления сцены и желаемую скорость вращения (которая задается при описании алгоритма в редакторе диаграмм потока данных системы SciVi).

19. Если $a < r_s$, повернуть модель на угол a вокруг оси A . Это действие приведет к мгновенному выставлению требуемой ориентации модели.

20. Иначе повернуть модель на угол r_s вокруг оси A . Это действие приведет к отображению очередного шага анимации поворота.

Рассмотренные алгоритмы обеспечивают калибровку устройства и постоянную реакцию 3D-модели на изменение условий освещения. Диаграмма потока данных, описывающая эти алгоритмы, приведена на рис. 3.

На основе указанной диаграммы система SciVi автоматически генерирует прошивку для микроконтроллера ESP8266. Код для устройства-хозяина и код сервера визуализации генерируются на языке C++, код клиента визуализации — на языке JavaScript. Интерфейс клиента основан на HTML5, рендеринг производится с помощью движка Three.js и использует API WebGL для обеспечения высокой производительности.

Фотографии устройства, лежащего в основе интерактивного экспоната *Dimetrodon grandis*, на которое была установлена сгенерированная прошивка, приведены на рис. 4.

4. ЗАКЛЮЧЕНИЕ

Интернет вещей — активно развивающаяся технология, способная привнести в музеи новые средства навигации для посетителей, новые способы мониторинга их активности и, главное, очень широкие возможности по организации интерактивных и запоминающихся экспонатов. Однако нехватка ИТ-специалистов в штате музейных сотрудников во многом тормозит внедрение данной технологии. Предлагается решить эту проблему путем создания высокоуровневых программных инструментов и рекомендаций относительно используемой аппаратуры, которые были бы просты в освоении и понятны музейным сотрудникам. Наличие таких инструментов и рекомендаций позволит реализовать концепцию “умного музея” без привлечения сторонних программистов и инженеров.

В составе разработанной ранее адаптивной системы научной визуализации SciVi был реализован модуль автоматической генерации прошивок для IoT-устройств. Встроенный в систему SciVi визуальный редактор диаграмм потока данных позволяет в интуитивно-понятном виде декларировать логику визуализации, интерактивности и калибровки для конкретного IoT-устройства без единой строчки низкоуровневого программного кода.

С точки зрения аппаратуры предлагается использовать популярные микроконтроллеры ESP8266 со встроенным WiFi-модулем. Периферию (датчики и исполнительные устройства) предлагается подключать в виде шилдов, что позволяет собирать IoT-устройства как конструктор. Это позволяет музейным сотрудникам быстро создавать необходимое оборудование под конкретные задачи, а также переиспользовать модули от более невостребованных экспонатов при создании новых

(например, после обновления экспозиции, в мобильных выставках и т.п.).

В настоящее время мы протестировали предложенную концепцию на практике при разработке интерактивного палеонтологического экспоната звероящера *Dimetrodon grandis*, созданного в музее пермских древностей. Данный экспонат состоит из научно-точной виртуальной 3D-реконструкции диметродона и IoT-устройства, определяющего направление света. Трехмерная модель, снабженная соответствующей анимацией, автоматически разворачивается боком к падающему свету, что демонстрирует поведенческие особенности древнего животного.

В дальнейшем планируется разработать мастер-класс для сотрудников музея по использованию предложенных инструментов, обучающий основным принципам создания интерактивных экспонатов на базе технологий IoT. Кроме того, планируется создавать другие интерактивные палеонтологические экспонаты, представляющие животных Пермского периода.

СПИСОК ЛИТЕРАТУРЫ

1. Rose K., Eldridge S., Chapin L. The Internet of Things: an Overview The Internet Society (ISOC), 2015. <https://www.internetsociety.org/resources/doc/2015/iot-overview>.
2. Kale N., Itkarkar R. Smart Museum Based on IoT // International Journal of Innovations & Advancement in Computer Science. 2017. V. 6. № 8. P. 241–247.
3. Chianese A., Piccialli F. Designing a Smart Museum: when Cultural Heritage Joins IoT // Third International Conference on Technologies and Applications for Smart Cities (I-TASC'14). 2014. 7 p. <https://doi.org/10.1109/NGMAST.2014.21>.
4. Alletto S., Cucchiara R., Del Fiore G., Mainetti L., Mighali V., Patrono L., Serra G. An Indoor Location-aware System for an IoT-based Smart Museum // IEEE Internet of Things Journal. IEEE. 2016. V. 3. № 2. P. 244–253. <https://doi.org/10.1109/JIOT.2015.2506258>
5. Ryabinin K., Chuprina S. Development of Ontology-Based Multiplatform Adaptive Scientific Visualization System // Journal of Computational Science. Elsevier, 2015. V. 10. P. 370–381. <https://doi.org/10.1016/j.jocs.2015.03.003>
6. Bock C., Fokoue A., Haase P., Hoekstra R., Horrocks I., Ruttenberg A., Sattler U., Smith M. OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition). World Wide Web Consortium (W3C), 2012. <https://www.w3.org/TR/owl2-syntax/>.
7. Chuprina S., Nasraoui O. Using Ontology-Based Adaptable Scientific Visualization and Cognitive Graphics Tools to Transform Traditional Information Systems into Intelligent Systems // Scientific Visualization. National Research Nuclear University “MEPhI”, 2016. Q. 1. V. 8. № 1. P. 23–44.
8. Lee B., Hurson A.R. Issues in Dataflow Computing // Advances in Computers. Elsevier, 1993. V. 37. P. 285–333. [https://doi.org/10.1016/S0065-2458\(08\)60407-6](https://doi.org/10.1016/S0065-2458(08)60407-6)
9. Ryabinin K., Chuprina S., Kolesnik M. Calibration and Monitoring of IoT Devices by Means of Embedded Scientific Visualization Tools // Lecture Notes in Computer Science. Springer, 2018. V. 10861. P. 655–668. https://doi.org/10.1007/978-3-319-93701-4_52
10. Blom J. Arduino Shields. Sparkfun. <https://learn.sparkfun.com/tutorials/arduino-shields>.
11. Bakker R.T. Dinosaur Renaissance // Scientific American, Inc. 1975. P. 58–78.
12. Witton M.P., Naish D., Conway J. State of the Palaeoart // Palaeontologia Electronica. 2014. V. 17. № 3. 10 p.
13. Hartman S. Taking a 21st Century Look at Dimetrodon Scott Hartman's Skeletal Drawing.com, 2016. <http://www.skeletaldrawing.com/home/21stcenturydimetrodon>.
14. Conway J., Kosemen C.M., Naish D. All Yesterdays: Unique and Speculative Views of Dinosaurs and Other Prehistoric Animals. Irregular Books, 2012. 100 p.
15. Ryabinin K.V., Chuprina S.I. Using Scientific Visualization Systems to Automate Monitoring of Data Generated by Lightweight Programmable Electronic Devices // Programming and Computer Software. Pleiades Publishing, 2018. V. 44. № 4. P. 278–285. <https://doi.org/10.1134/S0361768818040102>