



За едно приложение за добавена реалност в уеб базирана среда и 2D баркод маркери

Теодор Л. Георгиев, Станислав К. Железов, Станислав Ст. Станев

About a Web-Based Environment Application for Augmented Reality with 2D Barcode Markers

Teodor L. Georgiev, Stanislav K. Zhelezov, Stanislav St. Stanev

Abstract: *The article presents the developed augmented reality software application requiring minimal resources for its development and implementation. The application is developed on a web-based platform and uses 2D markers. The AR.js open source JavaScript library is used as a development environment.*

Keywords: *Augmented Reality, web-based environment application, 2D barcode markers*

ВЪВЕДЕНИЕ

През последните години в ежедневието на хората бързо навлизат технологиите на виртуалната (virtual reality – VR) и добавената реалност (augmented reality – AR), които са разновидност на генералната концепция за „управляема реалност“.

Добавената реалност е информационна технология за директно или индиректно изобразяване с цел възприемане на околната среда в реално време, чиито елементи се допълват (добавят) с други, генерирани от електронни сензори на базата на графични, звукови, GPS и други входящи данни. Първоначално за тази област от компютърните науки се използваше терминът „Обогатена реалност (на английски “Augmented reality – AR”), занимаваща се с комбиниране на данни от реалния свят с компютърно генерирани данни [5, 3–6].

Приложното поле на тази технология е много широко – в обучението, транспорта, строителството, медицината, производството, търговията, услугите, сигурността и отбраната [1, 4–12].

По-голяма част от създадените приложения с добавена реалност използват маркери или са чувствителни към геолокацията на потребителя. Подобни решения са икономически целесъобразни, тъй като не е необходимо скъпо струващо оборудване. Достатъчен е компютър с уеб камера или смарт телефон [2, 1–3]. При използване на маркери се извършва разпознаване на изображението и се определя от кои негови части ще бъде генериран виртуален обект, сцена или видеопоследователност. Геолокационната добавена реалност се създава след обработка на информацията относно позицията на потребителя, отчетена чрез GPS или електронен компас, и информацията относно целевия обект.

1. Подходи за разработване на мобилни приложения

Уеб базираните решения с добавена реалност са нова тенденция. Пренасянето на добавената реалност в уеб пространството предоставя възможност за използване в реално време на онлайн достъпни бази данни, изследвайки знанията в глобален контекст. Този поход има следните предимства:

- уеб ориентираните среди са по-лесни за администриране и не са необходими инсталация и поддръжка от страна на клиента;
- те са платформено независими;
- осигуряват достъп до уеб базирани инструменти, API, набори с данни;

– предоставят възможност за работа в сътрудничество и социализация на разработчици, автори и обучаващи се.

– добра мобилност– често уеб приложенията са достъпни и от мобилни устройства.

Целта на настоящата работа е да се представи разработеното от авторите софтуерно приложение за добавена реалност в уеб базирана среда от маркерен тип и неспециализиран, масово достъпен хардуер – мобилни телефони и планшети – с приложение в електронната търговия.

С цел избор на ефективен подход и средства за разработване на приложението е направен анализ на трите известни подхода за разработката на мобилни приложения – разработката на т.нар. "native" приложения (чрез програмния език и съответните програмни библиотеки и развойна среда (API/SDK) за съответната платформа), разработката на уеб базирано приложение и разработката в хибридна среда чрез т.нар. универсални приложни среди (Xamarin, Apache Cordova, React Native и др.). Избран е подходът за създаване на мобилно приложение като уеб базирано поради следните предимства [3, 4–8]:

1. Разработката на едно уеб базирано приложение като цялостен процес (планиране, разработка, тестване, имплементиране, поддръжка) е по-ефективно от разработката му като native откъм ресурси (финансови средства и време).

2. Използване на утвърдени и разпространени технологии, като HTML5, CSS и Javascript (jQuery mobile, Angular, Ionic, Meteor, PhoneGap и др.), което прави разработката по-бърза, лесна, а оттам – и евтина. Отпада необходимостта от софтуерни разработчици, които да познават всеки един език за различните платформи – Java (Android), C++ (Blackberry, Tizen), Objective C (Iphone).

3. При обновяване на приложението (отстраняване на грешки, добавяне на функционалност) не се изисква никакво съдействие от страна на потребителя – да изтегля и обновява приложението на телефона си – обновява се единствено уеб приложението, което се намира на уеб сървър.

4. Не е необходима поддръжка на няколко версии различни приложения – по едно за всяка платформа мобилни устройства (Android, iOS, Tizen, Blackberry, Windows mobile).

5. Уеб приложенията не заемат дисково пространство върху потребителското устройство.

2. Развойна среда за разработване на софтуера за уеб базирано приложение и платформа

За целта на настоящата работа като развойна среда се използва JavaScript библиотеката с отворен код AR.JS [4, 1–8]. Тя изцяло е предназначена за разработката на уеб базирани приложения и е платформено независима – може да се използва от всеки уеб браузър, който поддържа HTML5, Javascript, WebGL и WebGL2. В AR.JS се използват следните библиотеки:

1. Three.js [8, 1–2] – това е мултиплатформена и браузър независима JavaScript библиотека с отворен код за създаване и визуализиране на 3D анимирана компютърна графика, която се базира върху технологията WebGL, като използва за целта наличните на устройството графични копроцесори (GPU). Тази библиотека предоставя възможности за анаглиф, cross-eyed and parallax barrier ефекти; сцени: добавяне и премахване на обекти по време на изпълнението на кода, мъгла; перспективни и ортографични камери, контролери: трекболове, FPS и др.; анимации, права и обратна кинематика, морфинг и кийфреймове (keyframes); осветяване, сенки; материали, текстури и други; пълен достъп до OpenGL Shading Language (GLSL); обекти мрежи, частици, спрайтове, линии, ленти; шейдъри; пълен набор от времеви и 3D математически функции; формати за данни: двоични, графика, JSON; кубове, сфери, конуси, 3D текст; експортиране и импортиране: инструменти за създаване на Three.js съвместими JSON файлове от програмни продукти; API/SDK документация, публичен форум и wiki страници; над 150 файла с примерен код, към които и шрифтове, модели, текстури, звуци и други ресурсни файлове;

2. ArToolkit [7, 1–2] – популярна мултиплатформена (Linux, Windows, Mac OS, iOS, Android, Windows Mobile) библиотека с отворен код за добавена реалност. Поддържа програмните езици C++, C#, Java и JavaScript.

3. A-frame – уеб базирана софтуерна платформа с отворен код за създаването на виртуална и добавена реалност, разработена върху библиотека three.js. Към момента a-frame е една от трите най-масово използвани софтуерни рамки/библиотеки за създаване на виртуална реалност.

4. Emscripten и asm.js [6, 5–6] – с тяхна помощ JavaScript кодът се компилира до C и се изпълнява от уеб браузъра като двоичен код. Благодарение на това приложенията, разработени под AR.JS, демонстрират производителност от 40-50-60 fps на телефони, модел 2013–2014 г.

С цел подобряване на производителността на разработваните под A-frame приложения е използвана системата за управление на ресурси Asset за работа с външни мултимедийни обекти в A-frame библиотеката – графични файлове (снимки/текстури), видео и аудио файлове, 3D обекти. Използването на системата за управление на ресурси минава през следните стъпки:

1. Дефинира се съответният ресурс в двойката тагове <a-assets>, където му се присвоява уникален идентификатор и се указва местоположението му, откъдето може да бъде зареден.

2. При позоваване на този ресурс в кода на приложението същият се идентифицира именно чрез присвоения му ресурсен идентификатор.

За платформа на проекта се използва собствен виртуален работен сървър с инсталирана 64-битова операционна система Linux, дистрибуция – Centos 6.9, находящ се в Германия, с IPv4 адрес: 93.104.213.240 и IPv6 адрес: a02:c200:0:10:3:2:2451:1. На сървъра е инсталиран уеб сървър – Apache, версия 2.2.15. Може да се използва както Windows, така и Linux или друга операционна система (FreeBSD, vxWorks и други), а за уеб сървър може да се използва практически всеки – lighttpd, IIS, thttpd, nginx, tomcat.

Използваната библиотека за създаване на добавена реалност е изградена изцяло върху JavaScript, което обуславя пренасянето на изпълнението (а оттам и тежестта – ресурси и др.) на програмния код изцяло в брауъра на клиентското устройство. Така всяко лице, което разполага с компютърна конфигурация, снабдена с мрежова връзка, с дисплей и камера с добра разделителна способност, може да зареди и използва приложението. За целта е нужно единствено на устройството да е инсталиран HTML5 – съвместим уеб брауър с включена и безпроблемна поддръжка на JavaScript.

По този начин се избягват и неудобствата от инсталирането на какъвто и да е допълнителен софтуер върху крайните устройства на потребителите, което, от друга страна, отново допринася за по-широкото и лесно разпространение на услугата/продукта. Не се създават и компромиси със сигурността при някои мобилни операционни системи, при които е забранено (от производителя, от мобилния оператор, от учреждението, което е предоставило тези устройства на потребителите) инсталирането на софтуер върху тях.

3. Етапи на разработката на приложението и на програмния код

Основното предизвикателство при софтуерната реализация на технологията AR е в подреждането/наместването на виртуалните обекти със заобикалящата среда (фактическата действителност) в едно цяло (един цял образ). Най-често срещаният начин за решаване на тази задача е чрез използването на т.нар. „визуални маркери“. Такива са класическите квадратни маркери, 2D барковете и мулти-маркерите, които биват разпознавани от компютърната система с помощта на компютърно зрение.

На базата на анализа на известните технологии за разработка на AR приложения от гледна точка икономия на време за разработката и наличие на универсална технология – HTML5 и JavaScript е избран подходът да се разработва софтуерът като уеб базирано приложение. Избрана е технологията с 2D маркери за разпознаване на обектите в сцените на добавената реалност заради по-малкото процесорно време за обработка на данните.

Исходният код на приложението е вграден в HTML5 файлове, които са поставени на web сървър, откъдето ще се отварят чрез посочването на хипервръзките към тях (URL) в брауър от мобилни устройства (или от всякакви други). При разработката и тестването на програмния код са използвани две устройства, подбрани измежду средния ценови клас, и не по-нови от 3 години, с цена на пазар за устройства „втора употреба“ в порядъка от 100–150 лв. – мобилен телефон Motorola Moto G3 и Таблет Asus Zenpad 10.

Разработката на приложението включва следните стъпки:

- описание на баркода/барковете, които приложението ще разпознава за дефинираната триизмерна добавена среда;
- прикачване на 3D обекти и съответните им настройки, събития и други към тези маркери;
- дефиниране на една камера за триизмерната сцена на виртуалната реалност.

Тези стъпки се описват вътре в двойката тагове на директивата <a-scene>.

Например, за да се дефинират две фигури (примитиви) с определени параметри, които да се появяват в триизмерната сцена на добавена реалност при улавянето на баркод маркери, се описват следните директиви вътре в сцената:

```
<a-marker type='barcode' value='2'>
<a-box position="0 0.5 0" "color=red"></a-box>
</a-marker>
```

```
<a-marker type='barcode' value='3'>
<a-cone radius-bottom="0.5" radius-top="0" height="1" color="blue"></a-cone>
</a-marker>
```

Програмният код на A-frame се вгражда директно в HTML кода на уеб страницата посредством описания от него markup език с тагове. При разработката на програмно приложение под A-frame най-напред следва да се укаже използването на a-frame библиотеката в HTML кода, за да стане възможно използването на програмния език на a-frame библиотеката. В библиотеката a-frame се зарежда и библиотеката ar.js, за да може да се използват и нейните функции. След това в тялото на HTML кода (ограден от таговете <body>) се създава и настройва сцената на добавената реалност. Сцената се създава чрез двойката тагове на a-frame библиотеката <a-scene>, като в отварящия/началния/създаващия таг се указват и настройките за добавена реалност на библиотеката ar.js:

```
<a-scene embedded arjs='настройки'>
</a-scene>
```

В настоящата работа се използва arToolkit, тъй като е вградена в AR.js библиотеката и се използва масово от разработчиците на приложения под AR.js като най-стабилна, с активна разработка и даваща добри резултати откъм производителност.

Програмният продукт за създаване на добавена реалност, който е разработен, използва маркери вместо NFT (Natural Feature Tracking). Маркерите представляват предефиниран и лесно разпознаваем обект (тип „матрица“), намиращ се в рамка, който се поставя в пространството на триизмерната сцена на виртуална реалност и при разпознаването му в пространството софтуерът реагира и върху него (в зависимост от настройките) наслагва слоя добавена реалност.

Двуизмерните (2D) баркод маркери представляват квадратни маркери с рамка и условно или дефинирано от потребителя изображения/шаблони във вътрешността. В едно приложение, а по-точно – в една сцена на добавена реалност, могат да се използват и няколко маркера едновременно, които да представят различни обекти или координатни системи в това приложение/сцена.

Двуизмерните баркод маркери се разпознават от софтуера за константен времеинтервал, т.е. времето за разпознаване на всеки двуизмерен баркод маркер е едно и също. По тази причина използването на голям брой двуизмерни баркод маркери в една сцена не изисква допълнителна изчислителна мощ. В допълнение на това, когато се използват двуизмерни баркод маркери, вероятността един маркер да бъде объркан с друг е значително по-ниска.

Недостатък на двуизмерните баркод маркери е това, че шаблонът във вътрешността на маркера вече не е графичен/изобразителен като при тези, дефинирани от потребителя, което от маркетинг гледна точка често може да е сериозен аргумент за избор срещу 2D баркод маркерите.

В настройките на сцената е дефинирано, че ще се работи с двуизмерен баркод маркер във формат HAMMING 3 x 3 (матрица от три реда в три колони) 6.3, то ето как трябва да изглеждат маркерите за числата 2 и 3 (Фиг. 1.):





Фиг. 1. Маркери на числата 2 и 3

Важен аспект при конфигурирането на камерата в сцената на добавената реалност е дали камерата ще е статична, или динамична. При динамичната камера, когато един маркер бъде фиксиран/уловен от софтуера (чрез камерата), върху него се изобразява съответния обект, като наблюдението (tracking) на маркера продължава през цялото време и с местенето на маркера се премества моментално и обектът, който е привързан към него. Изчезне ли от обсега на камерата маркерът – премахва се от нея и свързания, с него обект, т.е. обектът/събитието е скачено за маркера.

Динамичната камера се дефинира самостоятелно вътре сцената като отделна единица, но извън всички дефиниции на обекти, ресурси и маркери:

```
<a-entity camera></a-entity>
```

При статичната камера, след като веднъж софтуерът е уловил маркера, обектът/събитието се изобразява в сцената на добавената реалност и не изчезва/спира, дори и маркерът да изчезне от обхвата на камерата. По този начин не е нужно постоянно да държим маркера в обхвата на камерата, а просто да го поставим един път на желаното място пред нея и след това да го отстраним. Както ще се види по-долу, всеки от тези два способа има своето реално приложение в практиката.

За да се дефинира статичната камера, освен горепосочената директива следва да се укаже и още една – отделна за съответния маркер и обект. Това става с двойката тагове <a-marker-camera>, като в отварящия таг се указва маркера, прикрепен към тази камера, а вътре в двойката тагове – обектите/събитията, които ще се активират, щом веднъж маркерът бъде разпознат:

```
<a-marker-camera preset="hiro">
  <a-cone radius-bottom="0.5" radius-top="0" height="1" color="tomato">
  </a-cone>
</a-marker-camera>
```

Целият код на динамична камера и маркери:

```
<!DOCTYPE html>
<html>
<head>
  <title>University of Shumen - augmented reality example #1</title>
  <meta charset="UTF-8">
  <script src="aframe-master.min.js"></script>
  <script src="aframe-ar.min.js"></script>
  <script src="ar.min.js"></script>
</head>
<body style='margin : 0px; overflow: hidden;'>
  <a-scene embedded arjs='detectionMode: mono_and_matrix; matrixCodeType: 3x3_HAMMING63; debug: false'>
    <a-marker type='barcode' value='2'>
      <a-box position="0 0.5 0" "color=red"></a-box>
    </a-marker>
```

```
<a-marker type='barcode' value='3'>
  <a-cone radius-bottom="0.5" radius-top="0" height="1" color="blue">
</a-cone>
</a-marker>
<a-entity camera></a-entity>
</a-scene>
</body>
</html>
```

4. Постигнати резултати

Разработени са две приложения, изключително семпли – за изпробване на 3 модела ръчни часовници и за анимация на планетата Земя. На Фиг. 2. е демонстрирано използването на приложението за избор на часовници.

Обновяването на програмния код (добавянето на нова версия на библиотеките, нова функционалност в кода, отстраняване на грешки) става неусетно за потребителя, който получава и работи с новата версия автоматично – със зареждането на страницата се зарежда и обновеният програмен код.



Фиг. 2. Демонстрация на приложението

ЗАКЛЮЧЕНИЕ И БЪДЕЩА РАБОТА

Чрез JavaScript библиотеките A-frame и AR.js могат успешно да се разработват и внедряват уеб базирани програмни продукти за добавена реалност, като библиотеките предлагат добра развойна среда, бързо и лесно създаване на приложенията, могат да се използват не само за създаване на примерни, малки приложения, а и на сериозни за реално използване.

Технологията „Добавена реалност“ има огромни перспективи за подобряване качеството на живота на хората.

В бъдеще предстои изследване ефективността на практическото използване на представените в настоящата статия софтуерни приложения и разработване на проекти за дигитализация чрез добавена реалност на учебни материали и помагала за началното и професионалното образование, както и за образованието на деца със специални потребности.

БЛАГОДАРНОСТИ

Настоящата статия е подпомогната от научноизследователски проект „Надеждност и защита на информация в социалните мрежи, графичните и 3D обекти в добавена и виртуална реалност“, № РД-08-159/09.02.2018, финансиран ФНИ на Шуменски университет „Епископ Константин Преславски“.

ЛИТЕРАТУРА

- [1] **Зеньков. 2016.** Зеньков, А. Самый большой список вариантов использования дополненной реальности. *Rusbase*. Retrieved from <https://rb.ru/story/ar-cases/>. // Zen'kov, A. Samyy bol'shoy spisok variantov ispol'zovaniya dopolnennoy real'nosti. *Rusbase*, <https://rb.ru/story/ar-cases/>.
- [2] **Иванова. 2012.** Иванова, М. Добавена реалност и технологични решения. *CIO*, брой 9. // Ivanova, M. Dobavena realnost i tehnologichni reshenia. *CIO*, 9.
- [3] **Dua. 2018.** Dua, K. A Guide to Mobile App Development: Web vs. Native vs. Hybrid. Retrieved from <https://clearbridgemoible.com/mobile-app-development-native-vs-web-vs-hybrid/>.
- [4] **Etienne, Veres, Brewе, et al. 2018.** Etienne, J., Veres, Z. Brewе, T. et al. AR.js – Augmented Reality for the Web. Retrieved from <https://github.com/jeromeetienne/AR.js/blob/master/README.md>.
- [5] **Ganatra. 2015.** Ganatra, K. Augmented Reality. Retrieved from www.slideshare.net/Khyati14Ganatra/augmented-reality-ppt-47315337.
- [6] **Herman, Wagner, Zakai. 2014.** Herman, D., Wagner, L., Zakai, A. asm.js. Retrieved from <http://asmjs.org/spec/latest/>.
- [7] **Senthamil, Lamb, Bux, et al. 2018.** Senthamil, N., Lamb, P., Bux, T. et al. ARToolKit. Retrieved from <https://github.com/artoolkit>.
- [8] **Three.js. 2018.** Retrieved from <https://threejs.org/>.