

УДК 004.921

DOI: 10.15827/2311-6749.17.4.10

ВЫДЕЛЕНИЕ ОБЪЕКТОВ ПЕРЕДНЕГО ПЛАНА ИЗ ПОТОКОВЫХ ВИДЕОДАНЫХ В ТРЕНАЖЕРНО-ОБУЧАЮЩИХ СИСТЕМАХ

А.М. Гуацинтов, зав. отделом, algts@inbox.ru;

К.А. Мамросенко, к.т.н., руководитель Центра, kirillam@ya.ru

(Центр визуализации и спутниковых информационных технологий, Научно-исследовательский институт системных исследований РАН, Нахимовский просп., 36, г. Москва, 117218, Россия)

Одним из видов мультимедийной информации, используемой в тренажерно-обучающих системах, является графический виртуальный образ инструктора. Для его внедрения в единое синтезированное трехмерное окружение создан метод рир-проекции, базирующийся на методе 3D-кеинга. Основной идеей является выделение объекта с однородного фона. Данный процесс можно описать как процесс создания маски, содержащей информацию о прозрачности изображения, отделяющей объект от остального изображения.

Так как любая реализация рир-проекции является ресурсоемкой, при обработке изображений большого размера на центральном процессоре затруднительно достичь работы подсистемы визуализации в реальном масштабе времени. Это требование к подсистеме визуализации является одним из базовых. Учитывая данное обстоятельство, разработано решение для использования вычислительной мощности графического процессора для реализации метода рир-проекции. Также предложен алгоритм автоматической настройки параметров на основании текущего изображения.

Ключевые слова: рир-проекция, тренажерно-обучающие системы, рендеринг, визуализация, потоковые данные.

Увеличение числа технических систем и их сложности приводит к необходимости создания новых видов технических средств обучения. Одним из подобных средств обучения являются *тренажерно-обучающие системы* (ТОС). Такие системы незаменимы в тех отраслях, где ошибки при обучении на реальных объектах могут привести к негативным последствиям, а их устранение – к значительным финансовым затратам. Рассматриваемые системы применяются в военном деле, медицине, атомной энергетике, авиации и космосе. Таким образом, задача построения современных ТОС является актуальной.

Под ТОС оператора *сложной технической системы* (СТС) будем понимать техническое средство для подготовки операторов в едином информационном окружении, отвечающее требованиям методик подготовки и создающее условия для получения знаний, навыков и умений, реализующее модель таких систем и обеспечивающее контроль над действиями обучаемого, а также для исследований [1]. Кроме формирования индивидуальных профессиональных навыков и умений, ТОС могут применяться для отработки групповых операций. Одним из видов мультимедийной информации, используемой в ТОС, является графический виртуальный образ инструктора [2]. Для его внедрения в единое синтезированное трехмерное окружение создан метод рир-проекции, базирующийся на методе 3D-кеинга.

Основной идеей технологии кеинга является выделение объекта с однородного фона. Данный процесс можно описать как процесс создания маски, содержащей информацию о прозрачности изображения, отделяющей объект от остального изображения. Создание качественной маски прозрачности – трудоемкая задача. Одним из ее решений является создание маски, которая делает фон изображения полностью прозрачным, а объекты переднего плана полностью видимыми. Однако такой подход неприменим для сложных сцен: объекты переднего плана будут иметь резкие края и, соответственно, выглядеть нереалистично. Для отображения таких объектов, как волосы и дым, необходимо, чтобы маска прозрачности поддерживала не только дискретные значения прозрачности (0 – прозрачный объект, 1 – непрозрачный объект), но и значения в пределах от 0 и 1 для обеспечения частичной прозрачности объекта.

Существует несколько подходов, используемых для выделения объектов с фона. Один из подходов состоит в использовании нескольких изображений, где присутствует объект, который необходимо выделить. Данный подход требует строгих условий съемки и применим только для статических объектов. Второй подход основывается на том, что объекты фона и переднего плана заранее известны. Пользователь выделяет объекты на переднем плане и фоне, а оставшаяся часть изображения – границы объекта на переднем плане – считаются зоной перехода. Данный подход не может быть использован в масштабе реального времени, поэтому неприменим для использования в подсистеме визуализации ТОС. Наиболее распространенный подход – использование однородного фона, например синего или зеленого. Использование синего или зеленого цвета в качестве однородного фона обусловлено тем, что они не являются естественным цветом для человеческой кожи. Недостаток данного в необходимости точной калибровки света для получения однородного фона.

Существует большое количество методов хромакеинга, применяемых для выделения объектов переднего плана с однородного фона, каждый со своими достоинствами и недостатками. Основными методами кеинга являются люма-кеинг (Luma keying, кеинг – по яркости), хромакеинг (Chroma keying – кеинг по цветовой составляющей), 3D-кеинг. Рассмотрим последний метод подробнее.

В 3D-кеинге маска прозрачности создается с использованием поверхностей и трехмерных объектов для отделения фоновых пикселей от остального изображения. Изображение представляется в виде точек определенного цвета (пикселей) в трехмерном пространстве *RGB* или *HLS*.

Обычно применяют два объекта: первый объект (*O1*), определяющий гарантированно фоновый цвет, перекрывается вторым объектом (*O2*), определяющим зону перехода. Одним из простейших объектов, определяющим границы цветов, является сфера. Сфера *S1* размещается в пространстве так, чтобы определять диапазон значений цвета, которые будут считаться полностью прозрачными. Координаты этой сферы могут быть заданы автоматически на основе анализа цветов изображения либо вручную указаны пользователем.

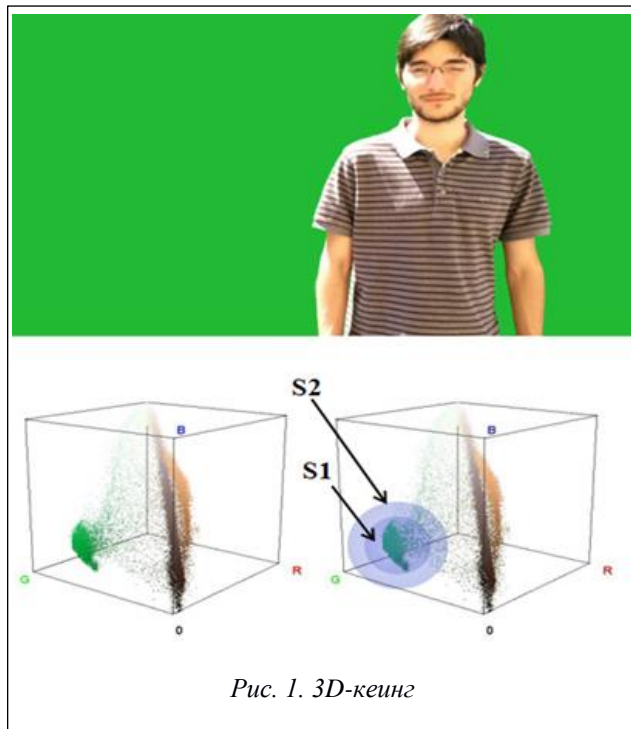


Рис. 1. 3D-кеинг

Для получения плавной зоны перехода создается вторая сфера большего радиуса *S2*, центр которой располагается в той же точке, что и центр первой сферы. Как видим на рисунке 1, пиксели, расположенные внутри сферы *S1*, являются полностью прозрачными, находящиеся вне сферы *S2* – полностью видимыми, а находящиеся между границами сфер *S1* и *S2* – полупрозрачными. Значения прозрачности рассчитываются на основе координат пикселя в трехмерном пространстве и центра сфер либо поверхностей *S1* и *S2*.

Практически любой тип трехмерного объекта может быть использован для отделения цвета фона от цвета объектов переднего плана, однако в большинстве случаев применяются выпуклые каркасы (convex hull). Соответственно, необходимо, чтобы пользователь выделил несколько пикселей фона с целью их перекрытия выпуклым каркасом.

Существуют и другие методы кеинга, которые требуют использования специального оборудования для захвата дополнительных данных, используемых для создания маски прозрачности. Один из методов использует инфракрасное из-

лучение для расчета дистанции каждого пикселя до камеры, поэтому он также называется z-кеинг. Используя информацию о глубине изображения, можно задать определенную границу и выделять только те объекты, которые соответствуют заданным параметрам глубины [3]. Другой метод подразумевает использование поляризованного света, который, отражаясь от кожи или поверхностей, становится неполяризованным, однако специальный материал фона отражает поляризованный свет. Разница в отраженном сигнале используется для создания маски прозрачности. Преимуществом данного подхода является независимость цвета объекта переднего плана от цвета фона, недостатком – значительные финансовые затраты на отражающий материал и оборудование для записи и обработки поляризованного сигнала. Еще один метод кеинга использует информацию о температуре объектов и создает маску прозрачности путем отделения холодных участков изображения от теплых [4]. Недостатком этого метода является то, что только живые или теплые объекты будут представлены как объекты переднего плана, в то время как некоторые детали, такие как волосы, не будут правильно вырезаны из фонового изображения [5].

Метод рип-проекции, реализованный в подсистеме визуализации ТОС, базируется на методе 3D-кеинга [6]. Так как любая реализации кеинга ресурсоемка, при обработке изображений большого размера на центральном процессоре затруднительно достичь работы подсистемы визуализации в режиме реального времени (не менее 25 кадров в секунду), что является одним из базовых требований к подсистеме визуализации. Даже использование современных многоядерных процессоров для выполнения процесса кеинга не гарантирует существенного увеличения скорости обработки. На сегодняшний день производительные многоядерные процессоры состоят из 8–16 ядер и способны одновременно обрабатывать от одного до восьми (при использовании различных оптимизаций) пикселей изображения на каждом ядре. Современные видеокарты состоят из тысяч шейдерных процессоров, способных одновременно обрабатывать пиксели изображения. Учитывая данное обстоятельство, было принято решение ис-

пользовать вычислительные мощности графического процессора для реализации кеинга.

Кеинг реализован в виде шейдера посредством языка GLSL и выполняется при помощи видеокарты. Обработка производится для каждого пикселя изображения. Пиксели изображения представляются в цветовой модели RGB, в то время как компонент альфа используется для определения прозрачности пикселя.

Первым этапом работы является сравнение компонентов цвета пикселя друг с другом. Каждая компонента цвета представляется в диапазоне от 0 до 255. В зависимости от настроек параметров алгоритма происходит выбор базовой компоненты, доминирующей в фоне обрабатываемого изображения. Рассмотрим случай, когда базовой является компонента зеленого цвета (то есть фон, на котором снимается объект, зеленый). Осуществляем проверку на преобладание зеленой компоненты цвета пикселя над остальными, то есть $(G > R)$ и $(G > B)$. Однако такой подход подвержен ошибкам, например, если $G = x$, $R = x - 1$ и $B = x - 1$, то данный цвет является оттенком серого. По вышеуказанному критерию этот пиксель будет считаться не принадлежащим объектам переднего плана, что неверно. Одним из способов повышения точности выбора пикселей с преобладающей зеленой компонентой является введение параметра дистанции:

$$d = \sqrt{(G - R)^2 + (G - B)^2}, d > d_{\max}, \tag{1}$$

где d – дистанция, определяющая степень преобладания зеленой компоненты пикселя над остальными; G – зеленая компонента пикселя; R – красная компонента пикселя, B – синяя компонента пикселя; d_{\max} – дистанция, задаваемая пользователем, свыше которой пиксели признаются фоном.

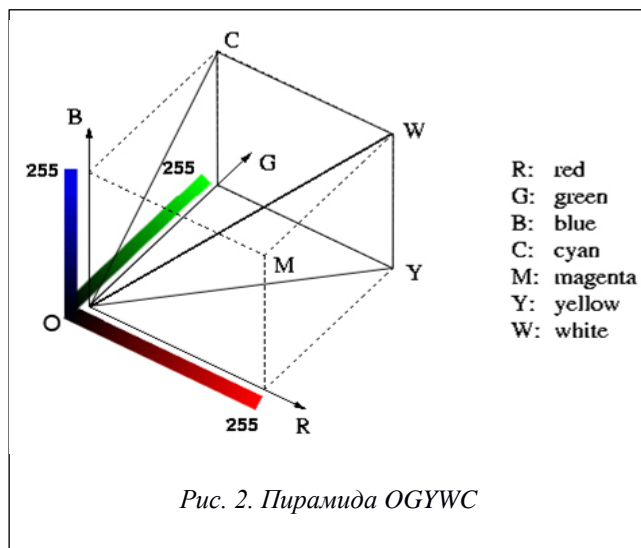


Рис. 2. Пирамида OGYWC

Если рассчитанное значение дистанции превышает задаваемое пользователем пороговое значение d_{\max} , пиксель считается фоном и, соответственно, должен быть прозрачным в итоговом изображении.

На рисунке 2 изображен RGB-куб, содержащий усеченную пирамиду OGYWC. Для каждой точки внутри пирамиды с координатами RGB выполняется условие $G \geq R$ и $G \geq B$.

На рисунке 3 показана усеченная пирамида OGYWC, пересеченная плоскостью S, параллельной диагонали OW. Необходимо отметить, что уравнение (1) применимо только для точек, расположенных внутри усеченной пирамиды (пиксели, в цвете которых преобладает зеленая компонента).

Положение плоскости S определяется значением параметра d_{\max} . Для каждого пикселя изображения производится расчет значения d ,

и маска прозрачности строится на основе сравнения значений d и d_{\max} .

Генерируемая маска прозрачности позволяет задавать частичную прозрачность пикселя изображения. При объединении обработанного кеингом изображения с изображением, генерируемым подсистемой визуализации, итоговый пиксель будет являться линейной интерполяцией цвета пикселей двух изображений, а рассчитанное значение альфа будет выступать в качестве параметра.

Использование частичной прозрачности позволяет добиться плавного перехода между объектами переднего плана и фоновым изображением, генерируемым подсистемой визуализации. Дистанция d , рассчитанная по формуле (1), используется в качестве входящего параметра для функции расчета значения альфа. На основе параметров d , d_{\max} и параметра мягкости (используемого в качестве порога, после которого пиксель считается лежащим в зоне перехода) рассчитывается значение альфа для конкретного пикселя. Формула расчета значения альфа основана на линейной интерполяции значений. Функция расчета значения альфа возвращает значение в отрезке $[0:1]$ (0 – цвет полностью прозрачный, 1 – полностью непрозрачный). В OpenGL цвет представляется в виде значений от 0 до 1, а не от 0 до 255. Формула (1)

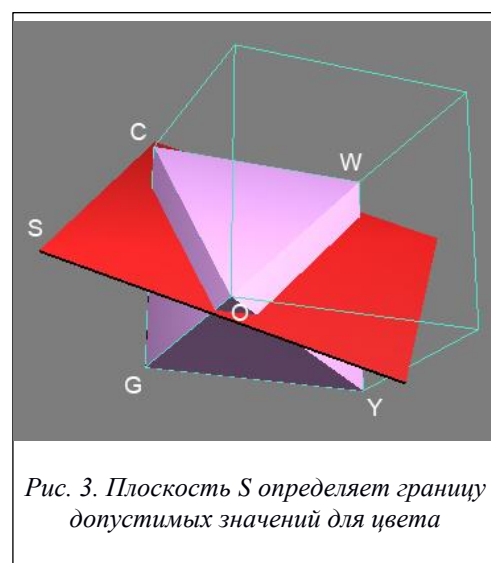


Рис. 3. Плоскость S определяет границу допустимых значений для цвета

возвращает значение в отрезке от 0 (полностью серый) до 360 (полностью зеленый).

Зачастую на объектах переднего плана после применения кеинга можно заметить цветové артефакты. Например, в случае применения зеленого однородного фона в области попадания отраженного света значение зеленой компоненты может оказываться меньше значения красной компоненты, соответственно, такой пиксель считается принадлежащим к объектам переднего плана. Такие цвета близки к светло-желтому, поэтому хорошо заметны на коже актера. Для преодоления данной проблемы была введена цветокоррекция. Пользователь выставляет пороговые значения, на основе которых алгоритм проверяет необходимость выполнения цветокоррекции для каждого обрабатываемого пикселя. Необходимость применения цветокоррекции основывается на выполнении следующих условий:

$$\begin{cases} G \geq R - k_r, \\ G \geq B + k_b, \end{cases} \quad (2)$$

где B , G , R – цветовые компоненты пикселя, k_r – задаваемый пользователем коэффициент для красной компоненты пикселя; k_b – задаваемый пользователем коэффициент для синей компоненты пикселя.

Если определено, что цветокоррекция необходима для данного пикселя, значение зеленой компоненты данного пикселя будет изменено на значение, лежащее в середине отрезка между значениями красной и синей компонент пикселя.

Разработанная реализация метода рип-проекции была внедрена в подсистему визуализации и протестирована на нескольких видеоматериалах, воспроизводимых в реальном времени. Была измерена производительность подсистемы визуализации на видеоматериалах с разрешениями 1280x720 (HD), 1920x1080 (FullHD), 3840x2160 (4K). Результаты отражены на графике (рис. 4).

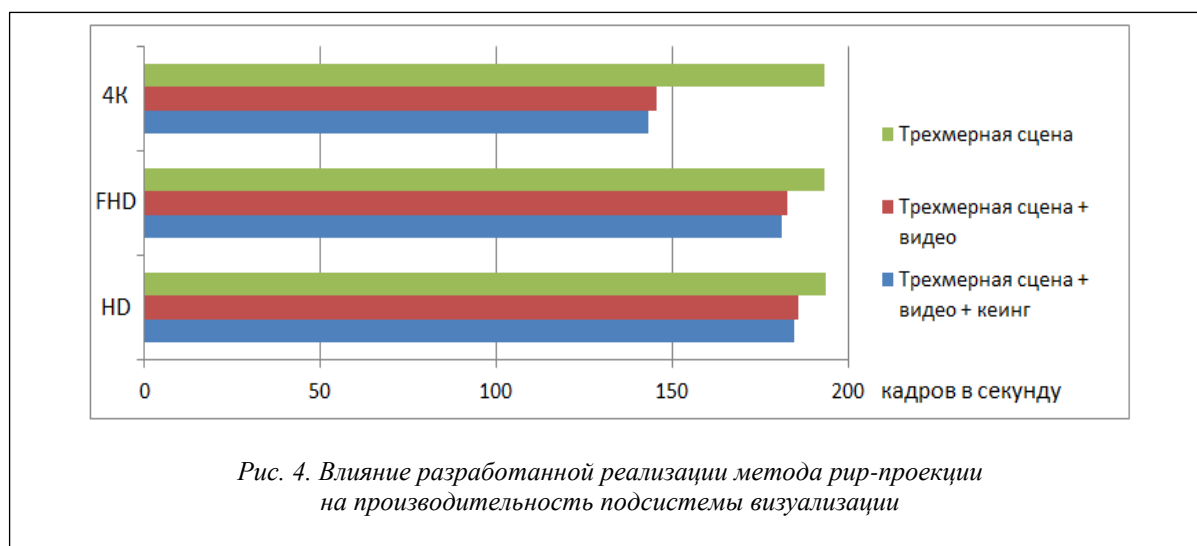


Рис. 4. Влияние разработанной реализации метода рип-проекции на производительность подсистемы визуализации

Полученные результаты показывают, что влияние работы шейдера кеинга на частоту кадров подсистемы визуализации незначительно и составляет доли процента. Применение разработанного шейдера в большинстве случаев позволяет отказаться от применения дорогостоящих аппаратных реализаций кеинга.

Реализованный метод позволяет выделять объекты переднего плана из однородного зеленого или синего фона. Для определения цвета фона и выбора методики обработки изображения производится однократный анализ каждого пикселя изображения. Анализ может осуществляться как на центральном процессоре, так и на графическом. Для каждого пикселя изображения определяется, превосходит ли зеленая компонента остальные компоненты пикселя. Аналогичные действия выполняются для анализа превосходства синей компоненты над остальными. После анализа всех пикселей изображения производится сравнение числа найденных пикселей, в которых преобладает зеленая компонента, и количества пикселей, в которых преобладает синяя компонента. В случае, когда у наибольшего числа пикселей преобладает зеленая компонента, для последующей обработки изображения в реальном времени применяется функция кеинга для зеленого однородного фона, в противном случае применяется функция кеинга для синего однородного фона. Описанный подход позволяет с достаточной точностью определить, какую функцию кеинга следует применить, когда количество пикселей с однородным фоном превышает количество пикселей объекта переднего плана (типовая ситуация).

Одним из недостатков описанного выше метода является необходимость его настройки под конкретные параметры съемки – от качества освещения и фона будет зависеть качество обработки изображения и выделения объекта переднего плана. Ручная настройка кеинга может занимать значительное время. Для уменьшения времени, затрачиваемого на настройку кеинга, был разработан алгоритм автоматиче-

ской настройки параметров обработки изображений.

Алгоритм основан на идее ранжирования пикселей на основании рассчитанного параметра дистанции. Диапазон возможных значений дистанции (от 0 до 360) делится на интервалы, в которые на основании расчета дистанции попадают пиксели изображения. Для повышения точности было принято решение разделить диапазон возможных значений дистанции на 36 интервалов.

Цвет каждого пикселя обрабатываемого изображения представляется в пространстве *RGB* и проходит проверку на преобладание одной цветовой компоненты над другими. Далее для каждого пикселя рассчитывается дистанция по формуле (1). На основании полученного значения определяется, в какой из интервалов попадает обрабатываемый пиксель.

После определения дистанции для всех пикселей изображения и распределения пикселей по интервалам начинается определение границ зоны перехода между объектом переднего плана и однородным фоном. Первой определяется ближняя граница зоны перехода. Для этого осуществляется поиск интервала, в котором количество пикселей значительно превосходит предыдущие. Поиск выполняется на основании задаваемого коэффициента, значения которого варьируются от 1,5 до 2.

Рассмотрим пример распределения пикселей в интервалах дистанций.

Интервал 1 – 792 пикселя, интервал 2 – 1 512 пикселей, 3 – 2 647, 4 – 4 468, 5 – 49 772, 6 – 168 134, 7 – 189 087, 8 – 177 682 пикселя.

Как видим, количество пикселей в интервале 5 значительно превышает количество пикселей в интервале 4. Значительное увеличение количества пикселей в данном интервале в большинстве случаев означает, что происходит переход от объекта переднего плана к однородному фону. Таким образом, ближняя граница должна перекрывать все пиксели, расположенные в пятом интервале, и располагаться между четвертым и пятым интервалами.

Дальняя граница зоны перехода определяет пороговое значение дистанции, после которого все пиксели считаются фоном и становятся полностью прозрачными. Для определения дальней границы интервалы дистанций начинают обрабатываться от конца к началу. Необходимо найти интервал, в котором присутствует максимальное количество пикселей. Как видно на примере, это интервал 7. Однако при таком подходе большое количество пикселей, расположенных в интервале 6, также попадут в зону перехода, что может привести к некорректному выделению объекта переднего плана. Для этого после определения интервала с максимальным количеством пикселей начинается поиск интервала, в котором количеством пикселей меньше максимального, на основании задаваемого коэффициента. Как и в случае коэффициента, применяемого при поиске ближней границы, значения данного коэффициента лежат в отрезке от 1,1 до 2,0. Также необходимо учитывать, что значение дальней границы зоны перехода не может быть меньше значения ближней границы зоны перехода.

После определения интервалов, в которых должны находиться значения ближней и дальней границ, происходит определение их точного значения. На данный момент значения границ определяются как среднее между значениями дистанций текущего и следующего интервалов. Соответственно, если для ближней границы был определен интервал 4, то значение ближней границы дистанции будет равно 45. Если для дальней границы был определен интервал 5, то значение дальней границы дистанции будет равно 55.

Данный алгоритм предназначен для работы по запросу и не рассчитан на использование в режиме реального времени. Была создана реализация данного алгоритма на центральном процессоре. В среднем при использовании достаточно современного процессора на анализ одного изображения в разрешении 1280*720 затрачивается от 0,5 до 1,5 секунды. Качество автоматической настройки кеинга можно оценивать только субъективно, однако в большинстве случаев после автонастройки ручная коррекция не требуется.

Задача выделения объектов переднего плана из однородного фона является комплексной и требует решения ряда сопутствующих задач, в частности, обработки данных в реальном масштабе времени. Были разработаны метод рир-проекции и его реализация, использующая мощности современных видеокарт и работающая в режиме реального времени. Разработанная технология в большинстве случаев позволяет отказаться от применения специализированных аппаратных средств и может быть использована на операционных системах Windows, Linux, macOS. Разработан алгоритм автоматической настройки параметров кеинга под конкретные параметры съемки, позволяющий в большинстве случаев корректно настроить параметры выделения объектов из фона. Планируется дальнейшее развитие разработанной системы, в частности, упрощение настройки параметров, относящихся к выделению объектов переднего плана из однородного фона.

Литература

1. Мамросенко К.А., Решетников В.Н. Моделирование подстилающей поверхности в имитационных системах // Программные продукты и системы. 2015. № 4. С. 70–74.

2. Гиацинтов А.М., Мамросенко К.А. Воспроизведение потоковых видеоматериалов в подсистеме визуализации тренажерно-обучающей системы // Программная инженерия. 2014. № 7. С. 33–39.
3. Robotics Institute: Z-Keying . URL: http://www.ri.cmu.edu/research_project_detail.html?project_id=79&menu_id=261 (дата обращения: 30.09.2017).
4. Thermo-Key. URL: <http://nae-lab.org/project/thermo-key/> (дата обращения: 30.09.2017).
5. Schultz C. Digital keying methods. Germany: Univ. of Bremen, 2006.
6. van den Bergh F., Lalioti V. Software chroma keying in an immersive virtual environment. South Afr. Comput. J., 1999, no. 24, pp. 155–162.