

УДК 004.946

МОДЕЛИРОВАНИЕ ЦИФРОВОГО РАССЫПАНИЯ ИЗОБРАЖЕНИЯ С ВИРТУАЛЬНЫХ ТЕЛЕКАМЕР В ТРЕНАЖЕРНЫХ КОМПЛЕКСАХ ПОДГОТОВКИ КОСМОНАВТОВ

А.В. Мальцев

Канд. физ.-мат. наук А.В. Мальцев (НИЦ «Курчатовский институт» – НИИСИ)

В статье рассматривается проблема имитации помех на синтезируемых изображениях трехмерной виртуальной среды, используемой в тренажерах для подготовки космонавтов. Предлагается оригинальный подход для реализации эффекта рассыпания визуализируемых изображений от виртуальных моделей цифровых телевизионных камер.

Ключевые слова: тренажеры, подготовка космонавтов, визуализация, виртуальная телекамера, рассыпания изображения

Modeling Breakup of a Digital Image from Virtual Television Cameras in Cosmonaut Training Simulators. A.V. Maltsev

The article addresses the problem of simulating interference on synthesized images within a three-dimensional virtual environment used in cosmonaut training simulators. An innovative approach is proposed to achieve the image breakup effect from virtual models of digital television cameras.

Keywords: simulators, cosmonaut training, visualization, virtual TV camera, image breakup

Имитация помех при визуализации виртуальной среды в тренажерах

В настоящее время трехмерное компьютерное моделирование активно используется при реализации как отечественных, так и зарубежных тренажеров подготовки космонавтов [1–4]. Окружающая обстановка, которую в космосе можно наблюдать посредством иллюминаторов и телекамер, в условиях тренировок заменяется на визуализируемую трехмерную виртуальную среду, содержащую цифровые визуальные модели (ЦВМ) космической станции, кораблей, Земли и т. д. Также существуют решения для полного погружения в моделируемое пространство с помощью гарнитур виртуальной реальности, когда необходимо отработать задачи перемещения и выполнения работ на поверхности космической станции.

Важным аспектом тренировок является обучение космонавтов управлению сложными процессами (например, стыковкой корабля к станции) по изображениям, получаемым с бортовых телекамер. В реальности эти изображения содержат помехи и искажения, которые могут возникать вследствие

целого ряда причин, а именно: конструктивных особенностей телекамеры, физических свойств ее компонентов, влияния внешних факторов, нестабильного сетевого соединения, неисправностей оборудования и каналов передачи информации. Искаженное изображение может существенно затруднить управление и привести неподготовленного человека в замешательство. Поэтому в ходе тренировок необходимо имитировать ситуации возникновения помех на изображениях с телекамер и прививать навыки работы в соответствующих условиях.

В настоящее время ведутся активные научные исследования в области создания виртуальных прототипов устройств видеонаблюдения и методов синтеза получаемых с их помощью видеосигналов [5–7]. Это касается и разработки новых подходов к моделированию помех. Изначально все бортовые телекамеры представляли собой аналоговые устройства. В работе [8] рассматриваются решения по реализации характерных для них помех, среди которых можно выделить «белый шум», срыв синхронизации по строкам и кадрам, засветку и пересветку изображения. Несмотря на то, что аналоговые приборы еще могут существовать в продолжающихся проектах, новые технические разработки по большей части ориентированы на цифровизацию. Однако переход к цифровым телекамерам не решает задачу устранения помех на изображении как таковых, а скорее переводит вопрос их существования в другую плоскость, в которой имеют место уже иные типы искажений. Наиболее распространенным из них является так называемое рассыпание картинки. Передача цифрового видео обычно осуществляется в формате множественных информационных пакетов, которые формируются при отправке данных и собираются воедино при их приеме. Во время пересылки пакеты могут теряться из-за проблем в линиях связи, перегрузок сетевого оборудования и т. д. Задержки приема принадлежащих одному кадру пакетов также могут различаться. Результатом описанных процессов является сборка кадров с недостающей информацией, приводящая к артефактам рассыпания. Другой причиной цифрового рассыпания изображения могут служить ошибки методов кодирования и декодирования видеосигнала, применяемых для экономии трафика передаваемых данных и повышения скорости их пересылки.

В данной работе предлагаются оригинальные подходы для моделирования в масштабе реального времени цифрового рассыпания синтезируемого изображения от виртуальных моделей телекамер, размещенных на космических кораблях. В основе рассматриваемых решений лежит применение широких возможностей для параллельных вычислений на современных многоядерных графических процессорах (Graphics Processing Unit, GPU).

Цифровое рассыпание изображения

Имитацию рассыпания синтезируемых видеопотоков с виртуальных моделей телекамер в данной работе предлагается реализовывать на базе технологии распределенной постобработки изображения, выполняемой на многоядерном

GPU после окончания этапа основного рендеринга кадра. Пусть визуализируемый кадр имеет размер $W \times H$ пикселей. После того как неискаженное изображение сформировано в буфере кадра, его необходимо скопировать в специальную текстуру T , разрешение которой эквивалентно разрешению этого буфера. Процесс постобработки производится с использованием собственного фрагментного шейдера. Данный шейдер выделяет и требуемым образом закрашивает n псевдослучайных прямоугольных «битых» областей в границах кадра, где $n \in N$. Максимальные размеры таких областей ограничиваются с помощью регулируемого вещественного коэффициента $k > 0$ и составляют $\lfloor kW \rfloor$ пикселей по длине и $\lfloor kH \rfloor$ пикселей по высоте. Здесь скобки обозначают округление числа до ближайшего целого снизу. В данной работе эмпирическим путем установлено, что для выбора значений k целесообразно использовать отрезок $[0,01; 0,7]$. Пара параметров n и k позволяют пользователю выполнять гибкую настройку интенсивности и плотности эффекта рассыпания изображения.

Выделение и обработка упомянутых областей выполняется в шейдере с помощью цикла из n итераций, каждая из которых отвечает за собственную область. В начале цикла формируются две пары псевдослучайных вещественных чисел (x_i, y_i) и (w_i, h_i) , удовлетворяющих требованиям $x_i, y_i \in [0,0; 1,0]$ и $w_i, h_i \in [0,0; k]$. Они задают относительные координаты левого нижнего угла i -й области эффекта и ее размеры соответственно. Генерация чисел осуществляется на основе метода хэш-функции, где в качестве аргументов выступают текущее время t_{cur} в мс, номер итерации и значения y_{i-1}, x_{i-1} . Значение t_{cur} определяется с помощью вызова функции получения системного времени. Необходимо заметить, что имя и параметры такой функции могут отличаться для различных операционных систем. Так, при реализации в системе Windows можно использовать функцию `timeGetTime`.

На следующем шаге цикла нужно проверить, попадает ли текущий обрабатываемый пиксел в рассматриваемую область. Для этого вычисляются его относительные координаты $(x_{\text{cur}}, y_{\text{cur}})$ путем деления абсолютных координат $X \in [0, W-1]$ и $Y \in [0, H-1]$ на ширину W и высоту H кадра соответственно:

$$x_{\text{cur}} = X / W, \quad y_{\text{cur}} = Y / H.$$

Далее применяется пороговая функция:

$$S(p, a) = \begin{cases} 0, & a < p \\ 1, & a \geq p \end{cases}.$$

Чтобы пиксел принадлежал области эффекта, требуется одновременное выполнение двух условий. Во-первых, он должен располагаться в горизонтальной полосе, ограниченной прямыми, которые определяют нижнюю и верхнюю границы этой области. Во-вторых, пиксел также должен

находиться в вертикальной полосе, которая задается прямыми, содержащими левую и правую границы области. Эти условия можно сформулировать в виде следующих равенств:

$$S(y_i, y_{\text{cur}}) - S(y_i + h_i, y_{\text{cur}}) = 1,$$

$$S(x_i, x_{\text{cur}}) - S(x_i + w_i, x_{\text{cur}}) = 1.$$

Если хотя бы одно из них не соблюдается, то шейдер выполняет закраску рассматриваемого пиксела исходным цветом, выборка которого производится из текстуры T по текстурным координатам x_{cur} и y_{cur} . Иначе, то есть при соблюдении обоих условий, необходимо закрасить пиксел в соответствии с требуемым визуальным видом артефакта для имитируемого рассыпания изображения.

В данной работе были реализованы несколько типов таких артефактов. Их упрощенные схемы представлены на рис. 1. Первый тип представляет собой

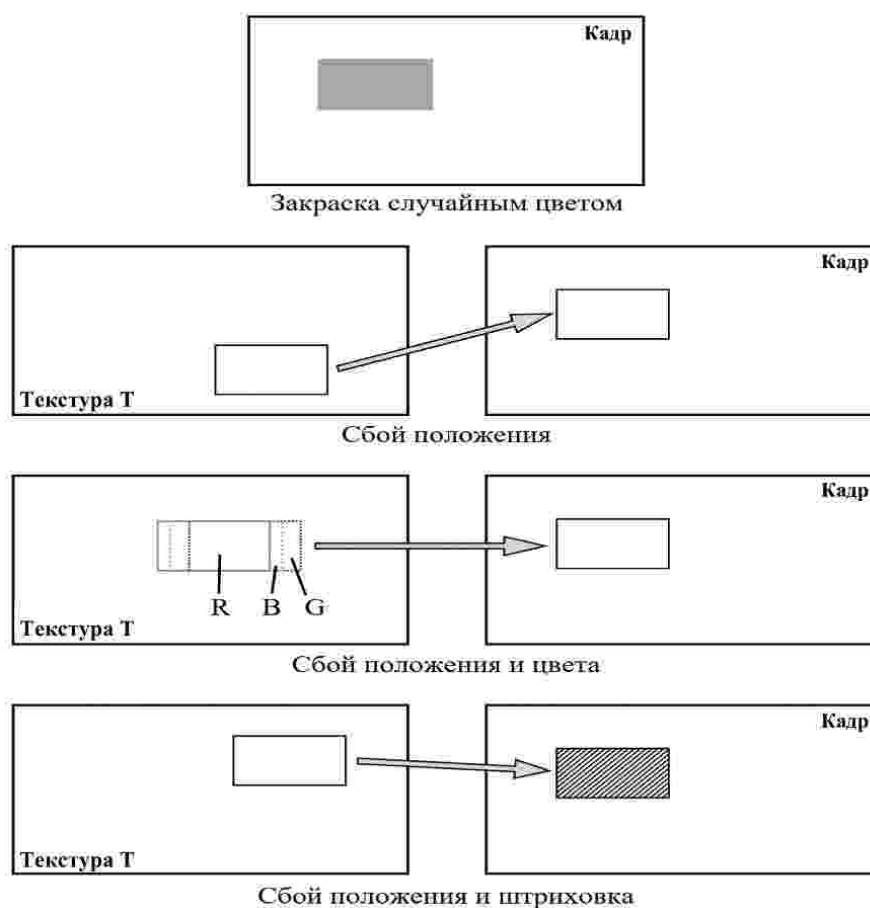


Рис. 1. Схемы заполнения области эффекта на кадре

однородную закраску области эффекта некоторым цветом $C = (C_R, C_G, C_B)$, где $C_R, C_G, C_B \in [0, 0; 1, 0]$. В нашем случае этот цвет рассчитывается на основе полученных ранее псевдослучайных чисел y_i, w_i, h_i и настроечного параметра k по формулам:

$$C_R = y_i; \quad C_G = w_i / k; \quad C_B = h_i / k.$$

Другой тип артефакта заключается в сбое расположений ряда блоков изображения, то есть они отображаются не на своих местах в кадре. Имитация такого явления происходит как бы в обратном порядке. Сначала определяются те части кадра, которые будут иметь некорректное содержимое. Им соответствуют описанные выше области эффекта. Затем выбираются эквивалентные по размерам участки кадра для отображения в данных областях. При этом для любого пиксела с относительными координатами $(x_{\text{cur}}, y_{\text{cur}})$, принадлежащего i -й области эффекта, цвет будет получен фрагментным шейдером путем выборки из текстуры T с помощью текстурных координат $(x_{\text{cur}} + \Delta x_i, y_{\text{cur}} + \Delta y_i)$, где $\Delta x_i, \Delta y_i$ – вещественные смещения, которые сохраняют свои значения для всей рассматриваемой области эффекта и на протяжении всего времени существования этой области (до пересчета ее границ). Модификациями описанного артефакта являются сбой положения с одновременным сбоем цвета, когда перемещаемая область дополнительно имеет некоторые смещения между цветовыми каналами R, G и B, а также сбой положения со штриховкой – перемещаемая область штрихуется параллельными линиями со случайным углом наклона.

Апробация предложенных решений

Предложенные в данной статье методы и подходы для моделирования цифрового рассыпания изображения от виртуальных телекамер были реализованы в программном комплексе VirSim [1], созданном в НИЦ «Курчатовский институт» – НИИСИ. Тестирование проводилось с использованием сцены с ЦВМ Международной космической станции, включающей порядка 1 млн треугольных полигонов. В ходе апробации применялись различные значения параметров интенсивности и плотности эффекта рассыпания изображения. В качестве аппаратных ресурсов для проведения рендеринга выступал персональный компьютер с процессором Intel Core i9 10940X и видеокартой NVIDIA GeForce 5070 Ti. Разрешение синтезируемого изображения составляло 3840×2160 пикселей (Ultra HD 4K), что в настоящее время является одним из лучших стандартов на рынке электроники. Проведенные измерения времени генерации кадра показали, что включение реализованного эффекта в общую визуализацию сцены вызывает прирост этого времени не более чем на 1 мс, что обеспечивает сохранение рендеринга в рамках масштаба реального времени (частота смены кадров не менее 25 раз в сек), необходимого для реализации тренажерных комплексов. На рис. 2 продемонстрированы

примеры изображений для различных типов артефактов рассыпания, полученных в ходе проведения экспериментов.

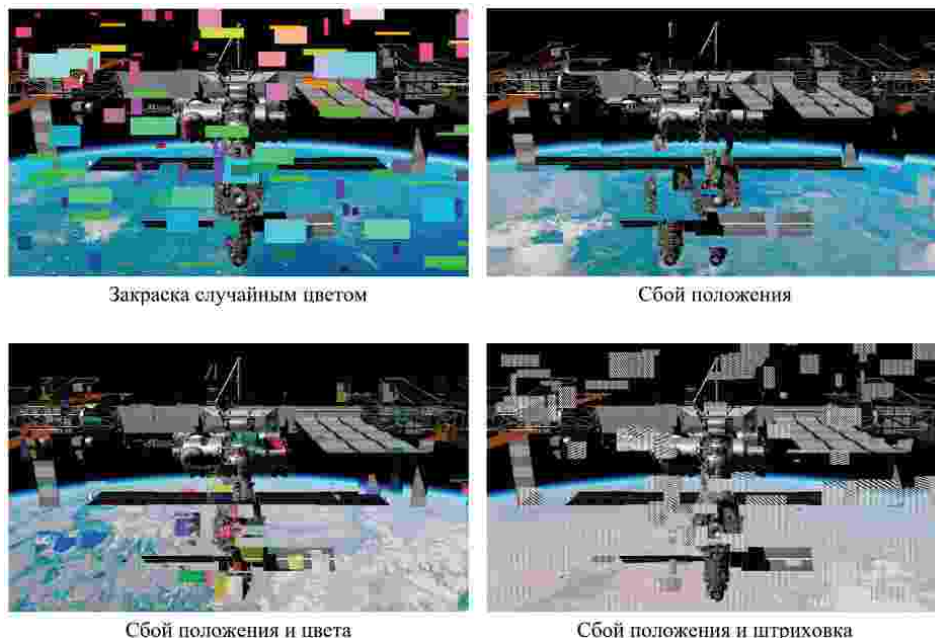


Рис. 2. Визуализация сцены с имитацией рассыпания изображения

Выводы

В работе рассмотрена актуальность имитации помех на изображениях от виртуальных моделей телекамер при реализации тренажерных систем для подготовки космонавтов. Представлены оригинальные распределенные методы и алгоритмы моделирования рассыпания изображения в случае цифровых видеопотоков, основанные на применении возможностей параллельных вычислений на современных многоядерных графических процессорах. Полученные в статье результаты могут быть использованы при разработке космических тренажеров, а также других тренажерных комплексов и систем виртуального окружения.

Публикация выполнена в рамках государственного задания НИЦ «Курчатовский институт» – НИИСИ по теме № FNEF-2024-0002 «Математическое моделирование многомасштабных динамических процессов и системы виртуального окружения».

ЛИТЕРАТУРА / REFERENCES

1. Михайлюк, М.В. Система виртуального окружения Virsim для имитационно-тренажерных комплексов подготовки космонавтов / А.В. Мальцев, П.Ю. Тимохин,

- Е.В. Страшнов, Б.И. Крючков [и др.] // Пилотируемые полеты в космос. – 2020. – № 4(37). – С. 72–95.
- Mikhaylyuk, M.V. The VirSim Virtual Environment System for Cosmonaut Training Simulators Facilities / A.V. Maltsev, P.Ju. Timokhin, E.V. Strashnov, B.I. Krjuchkov [et al.] // Manned Spaceflight. – 2020. – No 4(37). – P. 72–95.
2. Maltsev, A.V. Methods and Technologies of Cosmonaut Rescue Simulation in Virtual Environment Systems / E.V. Strashnov, M.V. Mikhaylyuk // Scientific Visualization. – 2021. – No 4(13). – P. 52–65.
 3. Garcia, A.D. Training Astronauts Using Hardware-in-the-Loop Simulations and Virtual Reality / J. Schlueter, E. Paddock // AIAA SciTech Forum. – 2020. – P. 1–13.
 4. Use of Virtual Reality for Astronaut Training in Future Space Missions – Spacecraft Piloting for the Lunar Orbital Platform – Gateway (LOP-G) / M.B. Bruguera, V. Ilk, S. Ruber, R. Ewald // 70th International Astronautics Congress. – 2019. – P. 1–10.
 5. Bisagno, N. Virtual Camera Modeling For Multi-View Simulation Of Surveillance Scenes / N. Bisagno, N. Conci // Proceedings of the 26th European Signal Processing Conference (EUSIPCO). – 2018. – P. 2170–2174.
 6. Maltsev, A.V. Simulation and Visualization of Optical Distortions for Virtual Surveillance Devices // Scientific Visualization. – 2025. – No 2(17). – P. 137–146.
 7. Simulation of Plenoptic Cameras / T. Michels, A. Petersen, L. Palmieri, R. Koch // 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D-Video (3DTV-CON). – 2018. – P. 1–4.
 8. Мальцев, А.В. Методы моделирования световых эффектов и искажений видеосигнала в виртуальных средствах наблюдения // Информационные технологии и вычислительные системы. – 2019. – № 2. – С. 54–61.
- Maltsev, A.V. Simulation Techniques for Lighting Effects and Video Signal Distortions in Virtual Surveillance Devices // Information Technologies and Computing Systems Journal. – 2019. – No 2. – P. 54–61.