

Колесников Андрей Александрович, Мыльников Владимир Владимирович

ВОПРОСЫ ПРОЕКТИРОВАНИЯ И СОЗДАНИЯ СИМУЛЯТОРА АВТОКРАНА КС-55727-1

Адрес статьи: www.gramota.net/materials/1/2009/11-1/7.html

Статья опубликована в авторской редакции и отражает точку зрения автора(ов) по рассматриваемому вопросу.

Источник

Альманах современной науки и образования

Тамбов: Грамота, 2009. № 11 (30): в 2-х ч. Ч. I. С. 47-52. ISSN 1993-5552.

Адрес журнала: www.gramota.net/editions/1.html

Содержание данного номера журнала: www.gramota.net/materials/1/2009/11-1/

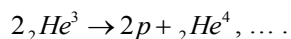
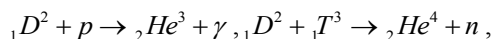
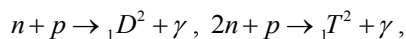
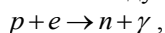
© Издательство "Грамота"

Информация о возможности публикации статей в журнале размещена на Интернет сайте издательства: www.gramota.net

Вопросы, связанные с публикациями научных материалов, редакция просит направлять на адрес: almanac@gramota.net

Кроме всего перечисленного выше необходимо понимать следующее: все ядра атомов, с их плотными упаковками полей протонов, избавленных от кулоновского расталкивания электронами из системы нейтрона, образуют своими полями, в занимаемом ими локальном объеме пространства (в системе поля ядра атома), структуру тождественную антиммиру. Именно в таком объеме и при взаимодействии с ним возникают наблюдаемые антиобразования или физические процессы свойственные антиммиру. Из объема атомного ядра, при его бомбардировке ускоренными частицами, вылетают частицы, не восстановившие структуру ячеек образований (или их систем) в момент деления - названные **мезонами**. Энергия гравитационного сжатия, затраченная на перестройку их структуры с добавочной кинетической энергией ускоренных частиц, у большинства из них имеет сферически симметричное распределение в пространстве, т.е. момент количества движения p_s или \vec{p}_s таких частиц равен нулю. Из изложенного выше, понятно, почему не было зарегистрировано солнечное нейтрино. Его и не должно быть. Таким образом, рассмотренная система, рожденная в гравитационном сжатии атомов водорода в центральном объеме протозвезды это не ипостась нуклона названная нейтроном, а сложная система из двух первичных образований. Поэтому её масса превышает массу протона на 1,29344 Мэв и эта система обладает дополнительным магнитным моментом, обеспеченным электроном, т. е. магнитным диполем.

Выше приведенную реакцию выгорания водорода в гелий (и не только в гелий) в протозвездах первого поколения следует рассматривать в другом варианте:



Выше изложенное в этой главе можно изложить в математической форме.

Эту возможность автор предоставляет каждому из Вас уважаемый читатель. Но это будет уже не “Структура пространства” а “Ядерная физика”.

Заключение

Все выше изложенное это еще не теория, а поэтапное исследование возможностей по описанию опыта наблюдений в физическом мире при взаимодействии всего лишь двух компонент материи. Автор считает, что может быть и не очень корректно, но решил поставленную перед собой задачу.

Список использованной литературы

1. **Кислицын А. П.** Введение в теорию инерциальных систем // Альманах современной науки и образования. Тамбов: Грамота, 2009. № 6: Математика, физика, технические науки и методика их преподавания. С. 78-98.
2. **Тулли Р. Б., Фишер Дж. Р., Йёвээр М., Эйнасто Я., Тиффт В. Г., Грегорн С. А.** Материалы Симпозиума Международного астрономического союза, г. Таллин. М.: Мир, 1981.
3. **Фейман Р., Лейтон Р., Сэндс М.** Феймоновские лекции по физике. М.: Мир, 1977. Т. 5. Гл. 1.

ВОПРОСЫ ПРОЕКТИРОВАНИЯ И СОЗДАНИЯ СИМУЛЯТОРА АВТОКРАНА КС-55727-1

*Колесников Андрей Александрович, Мыльников Владимир Владимирович
Адыгейский государственный университет*

Все мы привычно относимся к тренажерам и симуляторам различного типа самолетов, и др. летательных аппаратов. В настоящее время существует значительное количество тренажеров и симуляторов в авиации и космонавтике.

Симулятор высадки на луну обошелся НАСА в \$7500000.

По нашему мнению такие разработки должны существовать и для стройки, строительных машин и механизмов.

Теоретико-методологические основы статьи - основные понятия теоретических основ автоматизированного управления, проектирования программного обеспечения, проектирования АСОИУ, системного анализа, моделирования систем, создание систем обучения с использованием 3D графики.

Целью данной работы является создание 3D симулятора автокрана КС-55727-1, предназначенного для обучения крановщиков и контроля их знаний.

Практическая значимость. Быстрое и эффективное получение практических навыков по управлению автокраном КС-55727-1 без значительных денежных затрат. Снижение затрат на обучение на начальном этапе, за счет использования симулятора вместо работы обучаемых сразу на реальном автокране.

Для создания симулятора работающего в режиме реального времени необходимо решить задачи, обусловленные целью создания, в частности:

1. изучить методы и средства создания компьютерной 3D графики;
2. провести анализ основных подходов компьютерной 3D графики, используемых в играх;
3. создать реальную 3D модель автокрана КС-55727-1 в программе моделирования Autodesk 3D Max 2009 и модель окружения (База УПТК);

4. определить структуру программного обеспечения симулятора автокрана КС-55727-1;
5. разработать 3D движок, позволяющий отделить стандартные функции (система рендеринга, управление сценой, проверка столкновений, стандартное освещение и другие) от логики самого симулятора автокрана с использованием 3D графики;
6. с использованием движка создать программу-симулятор с использованием современных алгоритмов и подходов компьютерной графики (к ним относятся поддержка шейдеров и метод рендеринга, основанный на алгоритме отложенного затенения).

При визуализации технологий и производственных процессов возможности 3D графики на порядок превосходят обычные, двухмерные изображения. Черно-белые схемы и чертежи практически никогда не дадут такой наглядности, как реалистичная и многокрасочная 3D графика. Современные программы и системы 3D моделирования при необходимости могут создавать объекты и описания процессов со степенью реалистичности, близкой к фотографической. Правда последнее бывает нужно достаточно редко, так как при визуализации технологий и производственных процессов задача дизайнера сугубо практична: не удивить зрителя красотой картинки, а облегчить понимание.

Визуализация технологий и производственных процессов сегодня обычно использует 3D анимацию и 3D модели. Они необходимы, чтобы объяснить (пусть, чаще всего, упрощенно) смысл происходящих явлений. 3D визуализация стала наиболее эффективным и, в конечном итоге, экономичным способом наглядного иллюстрирования машин, механизмов и всего процесса в целом. Оно позволяет избежать чрезмерных финансовых затрат, сэкономить массу человеко-часов и других ресурсов при реализации технологического процесса на практике.

3D моделирование позволяет охватить вещь целиком, общим взглядом, и увидеть недостатки, которые не видны на двухмерных чертежах в силу их сложности. При этом внести изменения и дополнения в 3D модели проще. Для сравнения: если вносить изменения в конструкцию одной детали при помощи обычного рисования, это легко может привести к исправлению и перечерчиванию огромного числа схем, на которых изображена эта самая деталь.

Данная статья ставит своей целью создание симулятора автокрана. Для достижения этой цели выбираем свободное к распространению программное обеспечение Microsoft XNA GameStudio 3.0.

XNA GameStudio - это набор инструментов предназначенный для разработки игр и ориентированный на разработчиков-любителей создающих игры как под Windows так и под Xbox 360. XNA Framework - это набор .NET-библиотек классов основанных на .NET 2.0 Framework адаптированных для разработки игр.

XNA GameStudio очень удобна при прототипировании игр и других интерактивных приложений компьютерной 3D графики. Она позволяет в короткие сроки создать работающее приложение, которое будет отражать основные ключевые особенности приложения. Прототип не обязательно должен иметь продвинутую графику, в нем могут отсутствовать некоторые особенности финальной версии.

Для создания нашего прототипа симулятора нам понадобятся основные детали и агрегаты крана и окружения. Все эти 3D модели мы создадим в Autodesk 3D Max 2009 в формате .max. При необходимости или невозможности использования 3D Max можно использовать Blender или любую другую программу этой категории. Необходимо заметить, что программа должна поддерживать экспорт в формат FBX либо X, см Табл. 1. Это связано с тем, что программная среда понимает в стандартной комплектации только эти форматы, но может быть расширена за счет написания своих импортеров и процессоров.

XNA Framework - набор .NET-библиотек классов, делающих процесс разработки игр под платформы Windows и Xbox 360 более продуктивным.

Выбранный нами набор инструментов программного обеспечения позволяет выполнить эту работу в краткие сроки. Программное обеспечение свободно к распространению, т.е. бесплатное. Создание 3D модели автокрана, написание необходимых модулей программ выполняла группа из четырех программистов.

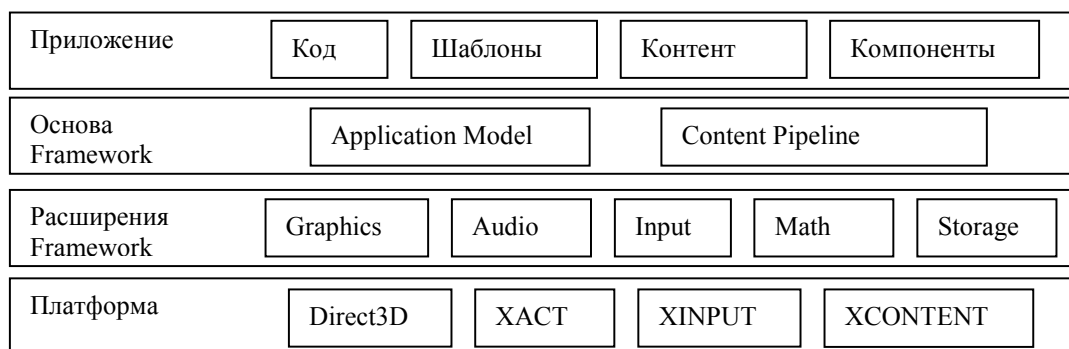


Рис. 1. Компоненты XNA Framework

Устройство Content Pipeline

3D программы на сегодняшний день включают в себя большое количество контента (текстуры, модели и т.д.). Добавление контента в игры не является простым занятием. Можно столкнуться с большим количеством проблем, например, при поиске экспортера или нужных инструментов для поддержки этого экспортера. Далее нужно найти инструменты для обработки контента и использовать их для создания данных для программы. Также нужно позаботиться о загрузке контента и о его правильном отображении в программе. На самом деле, у больших студий есть команды людей, которые работают только над внедрением контента в программы.

В Content Pipeline улучшен этот процесс и сделан отличающимся от стандартного подхода. Он прост и гибок в использовании. В Content Pipeline используются два основных понятия: импортер и процессор.

Необходимо заметить, что программа должна поддерживать экспорт в формат FBX либо X, см. Табл. 1. Это связано с тем, что программная среда понимает в стандартной комплектации только эти форматы, но может быть расширена за счет написания своих импортеров и процессоров.

Итак, рассмотрим, какие импортеры существуют для XNA Game Studio Express.

Табл. 1. Форматы, поддерживаемые в XNA

Форматы для 3d-файлов	Форматы для 2d-файлов	Форматы для материалов	Форматы для аудио-файлов
.FBX .X	.DDS .BMP .JPG .PNG .TGA	.FX	.XAP (XACT)

После того как импортер сделал свою работу, данные будут находиться в DOMе контента. Термин DOM используется для простого представления коллекции классов или схем (как в XML файле). Данные находящиеся в DOMе представляют собой строго типизированные данные. Процессор отвечает за получение данных из DOMа контента и создание объекта для вас в реальном времени. Этим объектом может быть все, начиная от простой модели, заканчивая сложным обрабатывающим устройством для программы. Конвейер контента включает в себя несколько процессоров, такие как Model (для простых объектов с текстурами), Texture2D (для текстур) и Effect (для работы с материалами объектов). Это означает, что, не нужно больше заботиться о вершинных буферах, но в тоже время можно более гибко работать с контентом, потому как мир может быть построен не только из обычных моделей.

Процессоры также сделаны так, что их можно с легкостью писать и использовать. Процессоры собирают данные из DOMа контента, и что это очень важно. Это потому что процессор не обращает внимания, откуда файл пришел. Все данные сохраняются в едином формате, и не нужно заботиться о переделке процессора под .x или .tga файлы. Также доступны функции манипулирования и генерирования новых данных из содержимого DOMа. Построение контента является важной частью системы, оно обрабатывается координатором компоновки. Так как контент будет находиться в Visual C# Express, когда мы нажмем Build весь контент будет скомпонован и сохранен на диск, готовый к использованию в реальном времени. Также есть система, которая отслеживает изменения контента, т.е. например, если вы поменяли текстуру, то перекомпируются только те объекты, которые используют эту текстуру.

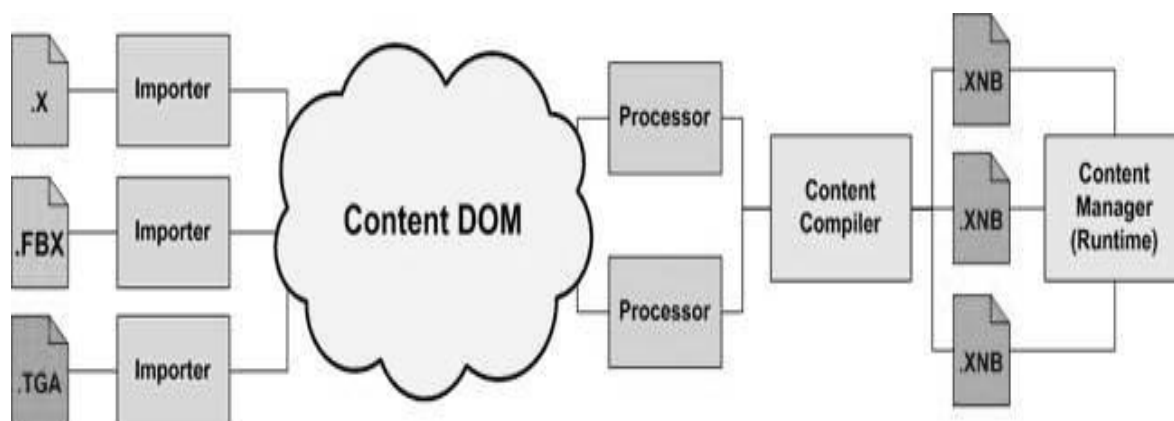


Рис. 2. Создание моделей в 3D max

Для создания моделей используются стандартные приемы, которых множество в различных книгах или ресурсах интернет. Детально их описывать нет необходимости. Модели можно создавать в любой программе моделирования. Однако, нужно заметить, что программа моделирования должна иметь возможность экспорта моделей в форматы X или FBX. Можно использовать и другие форматы, но тогда придется писать свой импортер для XNA GameStudio. Программами, которые имеют все необходимые функции, являются Blender, Autodesk Maya, Autodesk 3D Max.

Следует также отметить некоторые рекомендации, к которым следует прислушаться при создании моделей для симулятора автокрана:

- Модель крана состоит из отдельных сеток.
- Для каждой сетки существует материал, который может содержать стандартные данные 3D Max, а также карты (bump, diffuse, specular).
- Группы сеток назначены «костям», которые составляют иерархию. Кости нужны будут для управления моделью крана в программе.

Процесс создания модели автокрана довольно сложный и трудоемкий.

Сначала мы создаем все объекты крана по отдельности, затем создаем для каждой части материал.

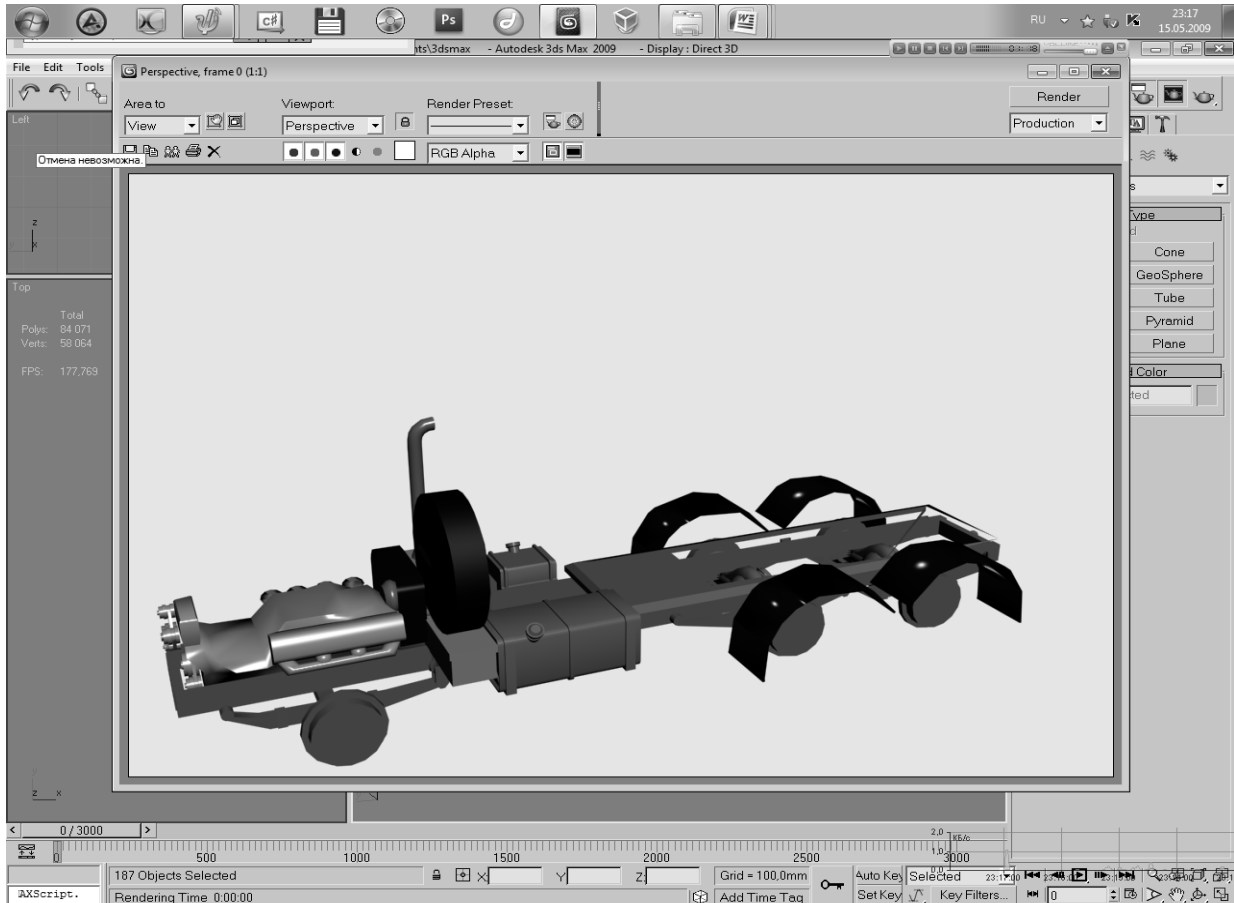


Рис. 3. Шасси автокрана

Кроме стандартных параметров материал может содержать карты. Для реализации приемов изложенных выше необходимы будут следующие карты: DiffuseMap, SpecularMap, BumpMap. Позже, возможно, понадобятся и другие. Их можно и не назначать, тогда 3D Max сгенерирует их сам. Используя инструмент Render To Texture, создаем необходимые карты для всех частей модели.

Далее все части модели крана необходимо переместить в места, где они должны находиться. После этого необходимо создать кости, создать их иерархию и назначить костям сетки.

В результате получаем модель крана полностью готовую для использования в симуляторе. Результат этого этапа виден на Рисунке 4.

На конечном этапе необходимо экспортировать модель в формат FBX и скопировать готовые карты.

Программирование приложения - программы-симулятора автокрана КС-55727-1

При программировании мы должны в Visual C# 2008 Express Edition создать класс системы визуализации (она будет отвечать за визуализацию с использованием отложенного затенения), класс сцены (сцена будет передаваться в визуализатор для отрисовки), класс базового объекта (будет содержаться в сцене, должен как минимум содержать матрицы трансформаций), класс крана, наследуемый от базового объекта (кроме базовых свойств содержит кости, трансформации костей и их ограничения). Для системы визуализации необходимо написать несколько шейдеров на языке HLSL с использованием шейдерной модели 3.0. Они будут отвечать за обработку специального графического буфера. Для аппаратной части будет необходима видеокарта с поддержкой MRT (множественные цели визуализации).

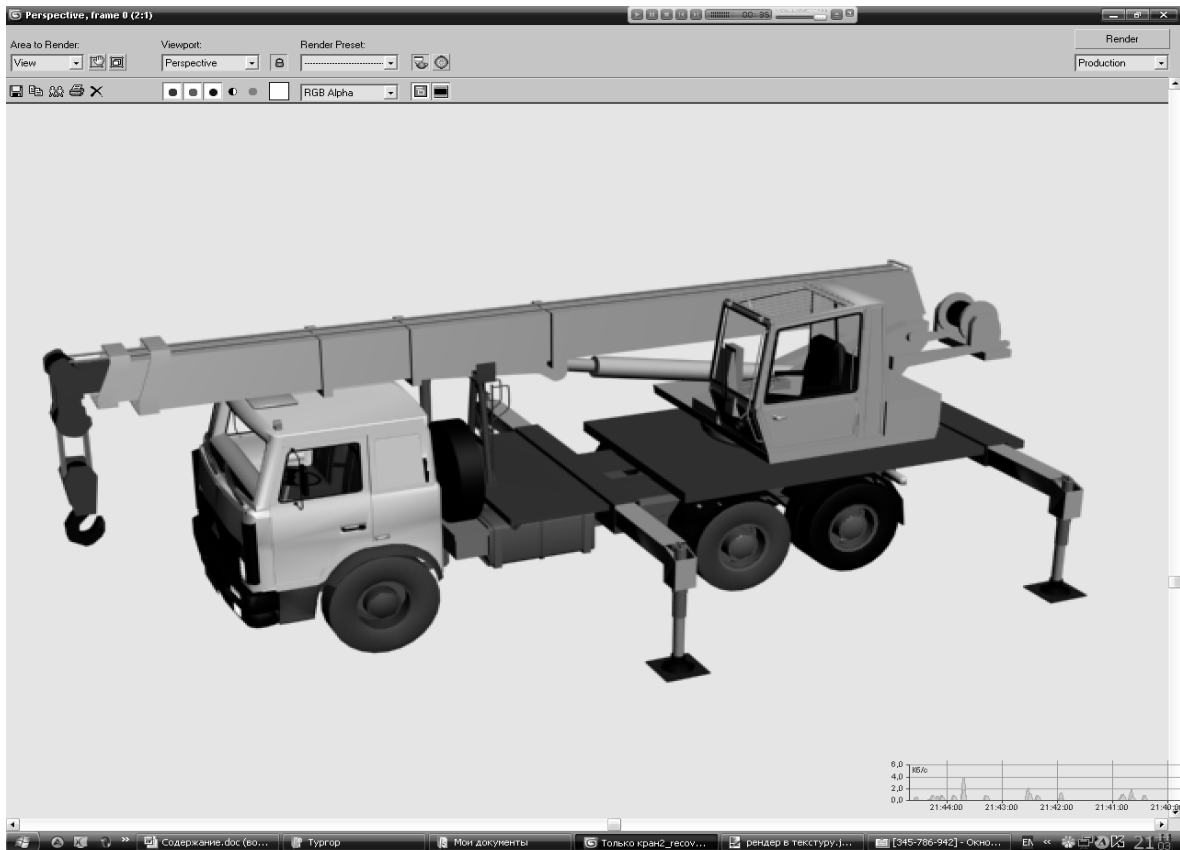


Рис. 4. Модель 3D автокрана

Исходный код приложения приводить нет смысла, так как он слишком большой даже не смотря на то, что XNA позволяет значительно упростить программирование. Приближенные результаты можно увидеть на Рис. 4, 5.



Рис. 5. Демонстрация функций симулятора 3D автокрана

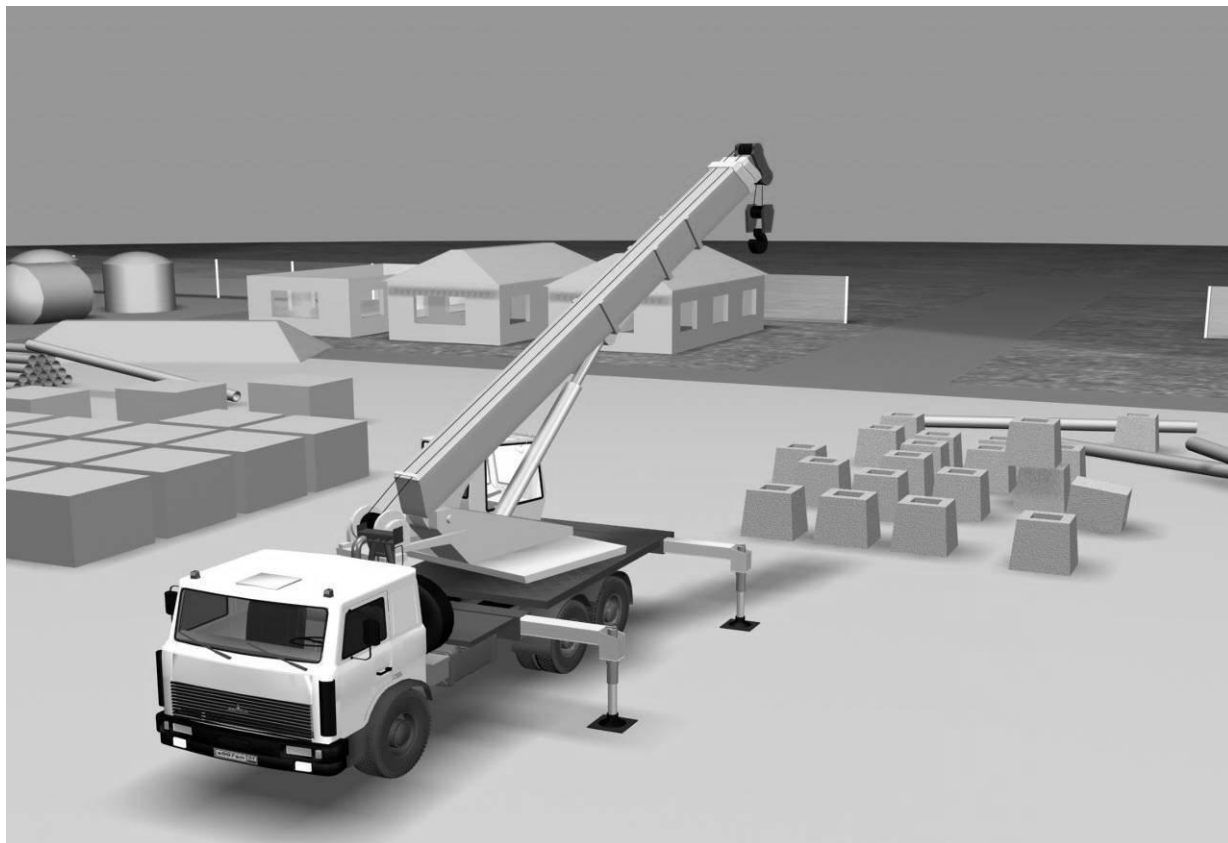


Рис. 6. Модель 3D автокрана с окружающей средой, УИТК

Основные действия, которые пользователь может сегодня выполнять с краном:

- Поднятие и опускание стрелы.
- Поворот платформы. Функциональные клавиши Y и H.
- Выдвижение и задвижение стрелы. Функциональные клавиши U и J.
- Поднятие и опускание крюка. Функциональные клавиши L и O.

Также существует возможность перемещаться по площадке с помощью клавиш W,A,S,D и мыши.

Можно отметить, что в программной среде XNA GameStudio 3.0 симулятор реальной модели автокрана КС-55727-1 работает в интерактивном режиме без задержек, выполняя все команды оператора пока с клавиатуры.

Список использованной литературы

1. Гайдуков Сергей. Усложненные технологии визуализации в XNA.
2. David H. Eberly. Game physics. Morgan Kauffman Publishers.
3. Fletcher Dunn, Ian Parberry. 3D math primer for graphics and game development. Wordware Publishing, Inc.
4. Morgan Kauffman. 3D game engine architecture. Morgan Kauffman Publishers, 2005.
5. Stan Trujillo. Direct 3D programming. The Coriolis Group, 1996
6. Stefan Zerbst, Oliver Duvel. 3D game engine programming.
7. Wolfgang F. Engel. ShaderX.
8. Wolfgang F. Engel. ShaderX2.
9. Wolfgang F. Engel. ShaderX3.
10. <http://ru.wikipedia.org/>
11. <http://ru.ziggyware.com/>
12. <http://ziggyware.com/>
13. <http://ru.ziggyware.com/>
14. <http://xnadev.com/>
15. <http://ru.ziggyware.com/>