

3D-моделирование объектов дополненной реальности с применением алгоритма поиска углов Shi-Tomasi

Демиденко О. М., Аксёнова Н. А., Кучеров А. И.

Гомельский государственный университет имени Франциска Скорины, Гомель, Беларусь

Аннотация. В данной статье описывается разработка программного модуля на Python для Blender 3D, исследуется алгоритм Shi-Tomasi определения углов на проектно-сметной документации застройщика. Извлеченные углы могут быть использованы для последующего построения трехмерных моделей, изменять план без внесения корректировки в проектно-сметную документацию или использоваться для извлечения точной информации.

Введение

Модуль поддержки моделирования «build-form» из разработанной библиотеки утилит Blue-Print SDK используется в Blender 3D для упрощения процесса построения 3D-моделей. В основе данного модуля лежит функция распознавания углов планов застройщика для последующего моделирования архитектурных проектов, зданий, домов, квартир и т. д. Данный процесс позволяет упростить работу создания трехмерной модели, позволяет вносить первоначальные корректировки изменения будущей трехмерной модели не прибегая к изменениям самой проектно-сметной документации. Ещё одной особенностью данной разработки является то, что готовые план-схемы являются метками дополненной реальности, на основе которых и реализуется мобильное приложение для просмотра полученной трехмерной модели.

Алгоритм Shi-Tomasi

Данный алгоритм используется для поиска и определения углов изображения, которое должно быть в оттенках серого. Углы изображения, в основном, идентифицируются как области с большим изменением интенсивности градиента во всех возможных измерениях и направлениях. При анализе цифрового изображения предлагается измерять изменение яркости пикселя (x,y) посредством смещения квадратного окна с центром в (x,y) на один пиксель в каждом из восьми направлений (вверх, вниз, вправо, влево, и в четырёх направлениях по диагонали). Размер окна чаще всего выбирается равным 3x3, 5x5 или 9x9 пикселей. Таким образом, для каждого из направлений (u,v) , где

$$(u,v) \in \{(1,0), (1,1), (0,1), (-1,0), (-1,-1), (0,-1), (1,-1)\}. \quad (1)$$

Для вычисления изменение яркости используется формула:

$$V_{u,v}(x,y) = \sum_{\forall a,b} (I(x+u+a, y+v+b) - I(x+a, y+b))^2, \quad (2)$$

где $I(x,y)$ – яркость пикселя (x,y) в изначальном изображении.

Для данного изображения I рассмотрим окно W (обычно его размер выбирается равным 5x5 пикселей, но может быть другим в зависимости от размера изображения) с центром в точке (x,y) , а также сдвиг этого окна на (u,v) (рисунок 1).

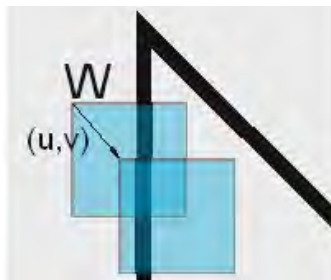


Рис. 1. Пример просматриваемого окна W и его сдвиг (u,v)

Тогда взвешенная сумма квадрата разностей между окном W и окном W, сдвинутым на (u,v) (т. е. изменение окрестности точки (x,y) при сдвиге на (u,v)) равна:

$$E(u,v) = \sum_{(u,v) \in W} w(x,y) (I(x+u, y+v) - I(x,y))^2 \approx \sum_{(u,v) \in W} w(x,y) (I_x(x,y)u - I_y(x,y)v)^2, \quad (3)$$

где W(x,y) – весовая функция (обычно используется функция Гаусса или бинарное окно).

Для обнаружения углов необходимо максимизировать функцию E(u,v). Для поиска производных применяем теорему Тейлора к приведенному выше уравнению и, используя некоторые математические шаги, получим:

$$E(u,v) \approx [u \quad v] M \begin{bmatrix} u \\ v \end{bmatrix}, \quad (4)$$

где M – автокорреляционная матрица:

$$M = \sum_{(u,v) \in W} w(u,v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}, \quad (5)$$

где I_x и I_y – производные в направлениях.

Угол характеризуется большими изменениями функции E(x,y) по множеству всех направлений (x,y), что эквивалентно большим по модулю собственным значениям матрицы M. Расположение собственных значений приведено на рисунке 2.

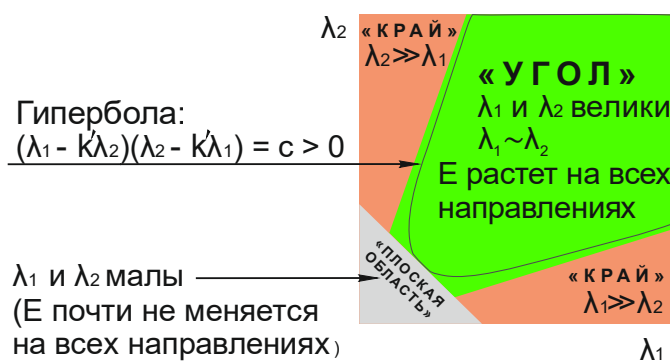


Рис. 2. Изменение значений функции E(x,y) при распознавании угла

Поскольку, считать собственные значения напрямую затратно с точки зрения ресурсов, предложена следующая мера отклика:

$$R = \min(\lambda_1, \lambda_2) \quad (6)$$

Таким образом, величины этих собственных значений определяют, является ли область угловой, краевой или плоской по следующим критериям:

1. Когда | R | мала, что происходит, когда λ_1 и λ_2 малы, область плоская.
2. Когда $R < 0$, при $\lambda_1 \gg \lambda_2$ или наоборот, то область является краевой.
3. Когда R велико, при λ_1 и λ_2 большие и $\lambda_1 \sim \lambda_2$, то область является углом.

Если это значение выше определённого порога, то точка считается углом и соответственно особой. Из рисунка 3 видно, что только когда λ_1 и λ_2 выше минимального значения λ_{\min} , это считается углом (зеленая область).

Работа модуля «build-form»

Первоначально пользователь указывает количество углов, которое необходимо найти. Далее указывается уровень качества, представленный значениями от 0 до 1, что означает минимальное качество угла, ниже которого все остальные углы отклоняются. Минимальное расстояние между углами обеспечивается евклидовым пространством. Указав данную информацию, функция находит углы на изображении. Все углы ниже уровня качества отклоняются, и функция сортирует оставшиеся углы по качеству в порядке убывания.

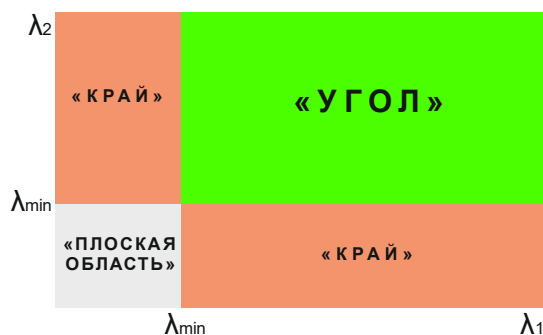


Рис. 3. Определение значений при распознавании угла

Затем функция берет первый самый сильный угол, отбрасывает все ближайшие углы в диапазоне минимального расстояния и возвращает N самых сильных углов. После этого для каждого обнаруженного угла программа помещает примитивную точку, вершину, в трехмерное пространство, которое лежит в границах ранее созданного плана здания, и визуально выравнивается с углами плана.

Он должен использоваться пользователем как подвал для создания на нем индивидуальной модели здания. После этого для каждого обнаруженного угла программа помещает примитивную точку, вершину, в трехмерное пространство, которое лежит в границах ранее созданного плана здания, и визуально выравнивается с углами плана.

После создания трехмерных угловых точек программа пытается выяснить, где находятся стены здания. Если пространство на линии между двумя точками имеет темный цвет, считается, что это пространство имеет стену. Программа создает края между точками там, где это необходимо (рисунок 4). Теперь пользователь может взаимодействовать с созданной моделью: устанавливать толщину всех или отдельных стен с помощью модификаторов, выдавливать здание по вертикали, чтобы придать ему заданную высоту, вставлять окна или дверные рамы с помощью логических операций или повторять процесс для каждого уровня здания. Таким образом, точная модель любого сложного здания может быть создана менее чем за день, если у пользователя есть доступ к план-схеме.

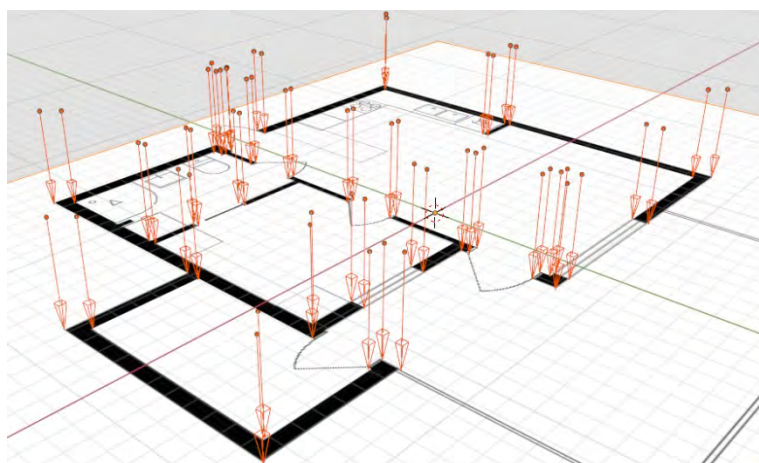


Рис. 4. Результат работы модуля «build-form» в Blender 3D

Заключение

В результате проведенных исследований разработан и создан программный модуль для реализации трехмерного моделирования в Blender 3D, использующий в своей основе алгоритм распознавания углов Shi-Tomasi. Модуль «build-form» входит в разработанную BluePrint SDK и является одним из основных модулей в создании мобильного приложения дополненной реальности.

Список литературы

1. Stefan Leutenegger, Margarita Chli and Roland Siegwart: BRISK: Binary Robust Invariant Scalable Keypoints. ICCV 2011: 2548–2555.
2. Harris, C. and Stephens, M. A combined corner and edge detector // In Fourth Alvey Vision Conference, Manchester, UK, 1988, pp. 147–151.
3. Аксёнова Н. А., Кучеров А. И. Разработка SDK для мобильного приложения с применением технологии дополненной реальности. Известия Гомельского государственного университета имени Ф. Скорины, № 3(126), Гомель, 2021, стр. 81–84.
4. Воруев А. В. Информационная ёмкость объектов в сопряжении с дополненной реальностью. Известия Гомельского государственного университета имени Ф. Скорины, № 3(126), Гомель, 2021, стр. 92–95.
5. Аксёнова Н. А., Воруев А. В., Кучеров А. И. Представление и реализация учебного материала с применением технологии дополненной реальности. Электронный сборник материалов Международной научной конференции «Тенденции и перспективы развития науки и образования в условиях глобализации», Украина, 2021. № 71, стр. 235–239.