

3D-Scanner with Kinect/LeapMotion

Роман Васильев

Санкт-Петербургский
Национальный Исследовательский
Академический Университет
Российской Академии Наук

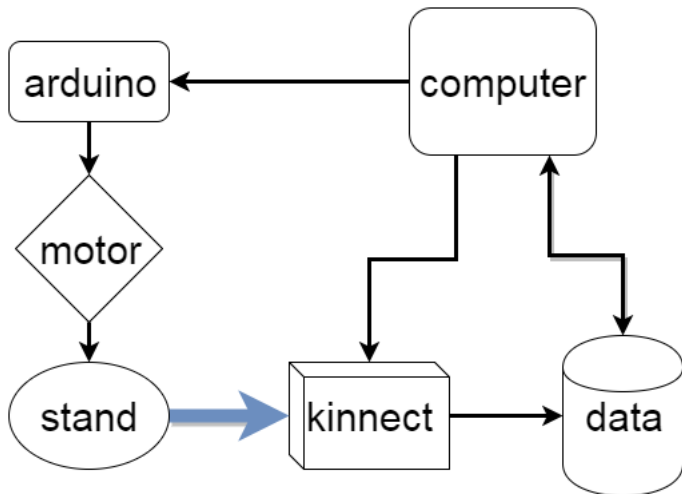
Научный руководитель
Кирилл Кринкин

21 декабря 2015 г.

Содержание

- 1 Введение
- 2 Схема
- 3 Программные компоненты
- 4 Оборудование
- 5 Пример
- 6 Итог

- Цель
 - Разработать 3D-сканер
- Задачи
 - Изучить API RGB-D сенсоров
 - Разработка аппаратной части сканера
 - Разработка алгоритма построения 3D-модели
- Результат
 - Разработать 3D-сканер



Програмные компоненты

- Для получение данных с Kinect использовалась библиотека `libfreenect`, написанная на C. Работа с кинектом происходила при помощи интерфейсов, написанных на C++.
- Для обработки облаков точек использовалась библиотека `Point Cloud Library`. Из-за того, что шаблонные классы каждый раз долго пересобирались, была собрана динамическая библиотека с нужными интерфейсами.
- Все интерфейсы были обернуты в классы Python при помощи SWIG для удобства работы.
- Чуть-чуть кода для Arduino на C.

Весь код находится в репозитории <https://github.com/nizshee/scannect>



Kinect — бесконтактный сенсорный игровой контроллер, первоначально представленный для консоли Xbox 360, и значительно позднее для персональных компьютеров.

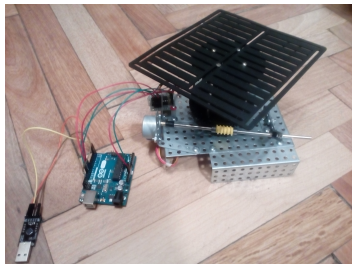
Предполагается, что компьютер работает под управлением Windows.

Состав:

- Color Sensor - просто камера
- IR Emitter - инфракрасный излучатель
- IR Depth Sensor - инфракрасный приемник.

Максимальное разрешение карты глубины 640x480 (30 fps).

Крутящаяся подставка



Состоит из: шагового двигателя (поворот подставки), Arduino (прием команд и управление шаговым двигателем) и безымянного конструктора.

С помощью библиотеки PySerial через цифровые порты ввода/вывода (RX/TX) подаются команды на Arduino, на какой угол нужно развернуть подставку.

Поворот с точностью ~ 1 градуса.



Kinect поднят для того, чтобы был виден верх образца.

Принцип работы:

- Образец сканируется при помощи Kinect, поворачивается на небольшой угол, снова сканируется и т.д.
- После этого полученные облака точек накладываются друг на друга (при помощи pndt) и сливаются в одно большое скопление точек.
- Оно сглаживается и фильтруется.

Пример

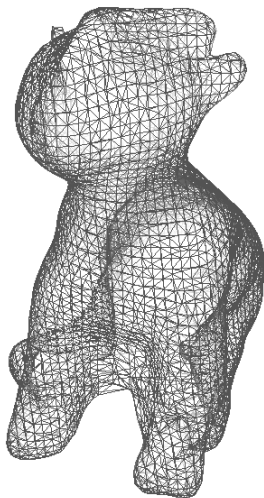


Попытка оцифровать козлика.

Стоит учесть, что козлик был достаточно большим и имел сложную форму.

Поэтому некоторые участки содержат больше точек, некоторые меньше.

Пример



- Получилось сделать прототип 3D-сканера.
- Можно делать 3D-модели объектов, которые при поворотах изменяются плавно.
- Хуже дела обстоят с объектами, типа параллелепипедов.
- Можно было бы добавить окраску точек в соответствующие цвета.

Спасибо!