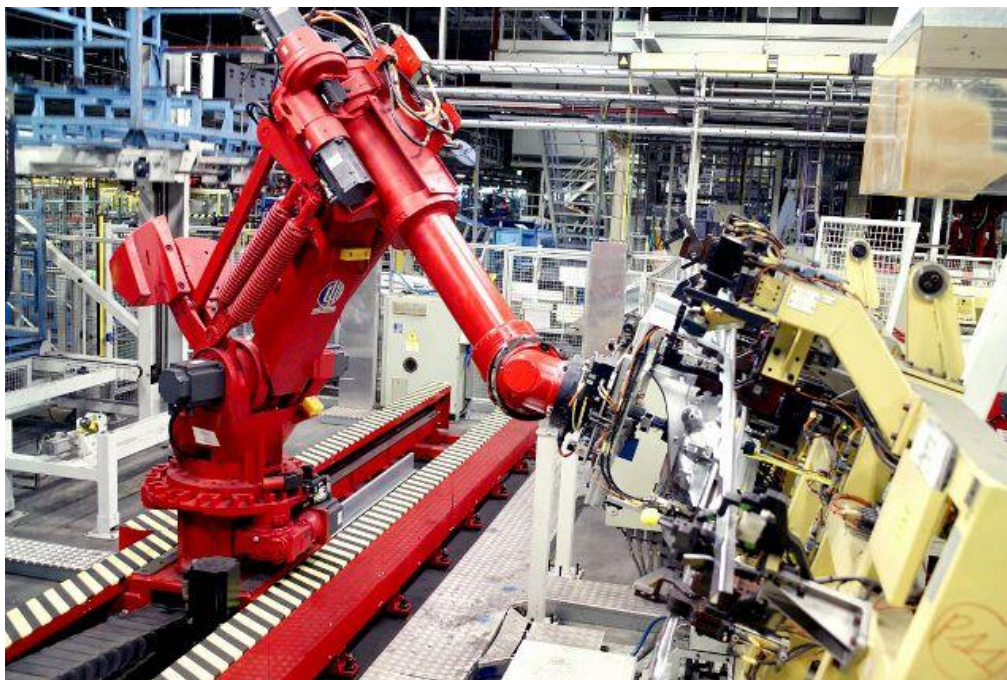


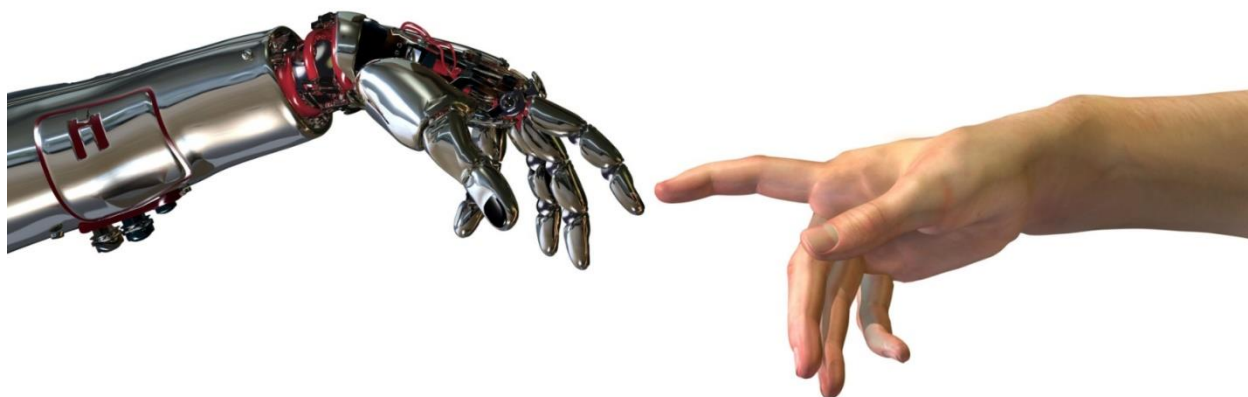
РОБОРУКА

**Удаленное управление манипулятором
при помощи 3D-сенсора**

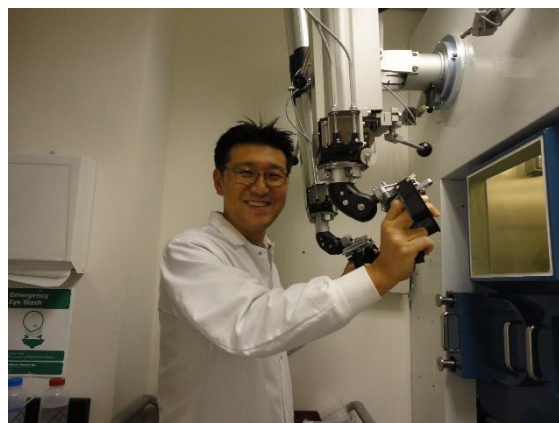
Манипуляторы уже давно используются в промышленности. В подавляющем большинстве случаев, эти манипуляторы предназначены для решения узкоспециализированных задач и совсем не похожи на руки человека. Это связано с тем, что на заводах эти роботы функционируют в “своей” среде, созданной специально для них.



Для роботов, которые функционируют в одной среде с человеком, существенным фактором становится эмуляция человеческих рук. Естественно, уже неоднократно разрабатывались модели манипуляторов в виде человеческой руки для человекоподобных роботов. Однако, повторение кинематики и механики конечностей, созданных живой природой, в механических манипуляторах представляется нетривиальной задачей.



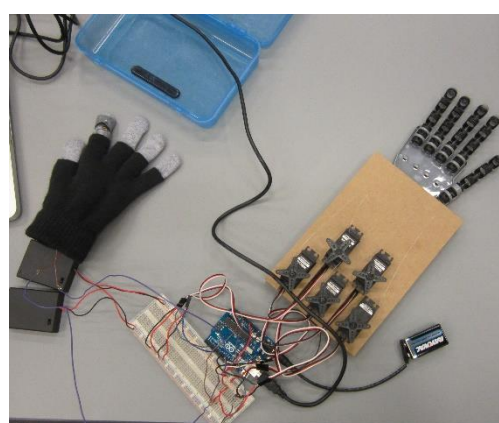
Итак, для начала разберемся с тем, где и зачем могут применяться манипуляторы подобные человеческой руке. Возьмем, например, ядерные реакторы. Они вырабатывают энергию за счет выделения распадающимся урановым стержнем (состоящим из оксида урана) тепловой энергии, что приводит в движение турбину, и тем самым производят электричество. Но после того, как 5% урана подверглось распаду, весь стержень загрязняется другими элементами (такими, как плутоний, америций, церий) и становится непригодным для производства энергии. Однако существует метод очистки оксида урана от примесей, разработанный в США национальной лабораторией в Аргоне. К сожалению, по причине того, что уран радиоактивен, работать с ним голыми руками или даже в защитном костюме нельзя. Для этого существуют специальные камеры, в которых проводятся такие работы посредством манипуляторов, управляемых вручную рычагами.



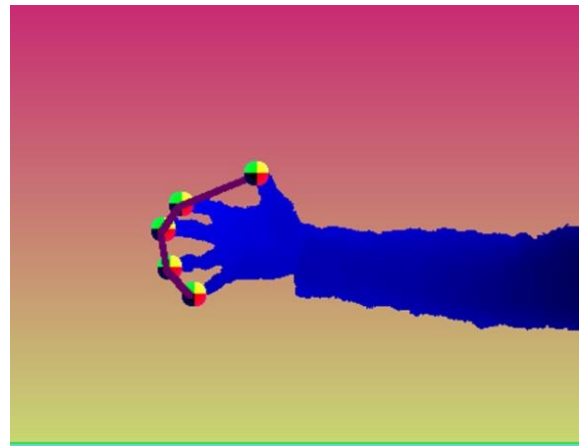
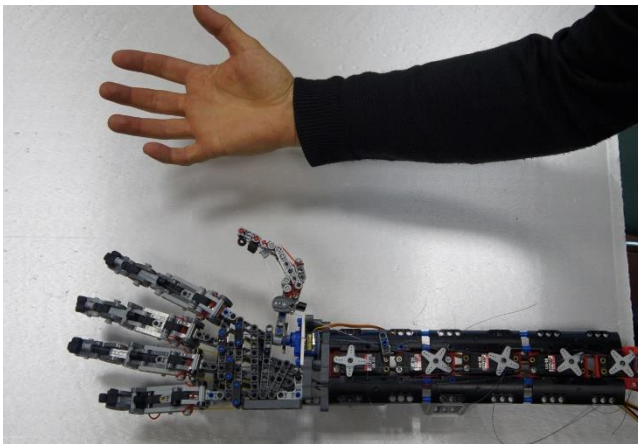
Проводить опасные для человека работы удобнее при помощи манипуляторов, повторяющих движение рук.

Существуют и различные варианты управления такими механизмами.

Обычно оно осуществляется при помощи датчиков, подключенных непосредственно к руке оператора, или же специальных перчаток, надетых на руку оператора, который удалённо управляет манипулятором. Надо признать, что эти способы неудобны. Датчики - по той причине, что их приходится закреплять на руку; могут привести к раздражению; требуется курс обучения, чтобы правильно подключить их. Перчатки - потому что в них руки могут потеть. Для разных людей потребуются перчатки разных размеров, к тому же это не гигиенично.



В данном проекте применен бесконтактный способ считывания положения рук человека, что предотвращает все вышеуказанные проблемы. Данные, полученные с помощью 3D датчика, трансформируются в управляющий сигнал для механического манипулятора, построенного по подобию руки человека. Любой человек может стать оператором манипулятора без какого-либо обучения, достаточно поднести руку в определённое место на стенде. Система получает облако глубины и изображение, из них выявляет руку, из руки определяет положение каждого пальца руки. Полученные значения, переработанные в координаты относительно выбранного тела отсчета, передаёт на NXT по Bluetooth, где они преобразуются в сигналы на моторы, сгибающие и разгибающие пальцы. Таким образом, пальцы манипулятора повторяют движения пальцев руки оператора.



Целью проекта является создание руки манипулятора и управление им при помощи 3D сенсора.

Для решения задачи был создан аппаратно-программный комплекс, включающий модель манипулятора, датчик Leap motion и компьютер. Конструкция манипулятора собрана из конструктора Lego по подобию человеческой руки, с использованием контроллеров Lego Mindstorms NXT и Tetrrix, а также сервомоторов HiTec. Программа написана на языке C#. Для управления сервоприводами через контроллер NXT написана на языке RobotC.

```

260
261     int preMes = 0, pref5 = 0;
262     int wasd = 0;
263
264     void drwHand()
265     {
266         preMes = (int)mes2[0];
267         pref5 = f5;
268         int h1 = 0, h2 = 0;
269         int old = 0;
270         leap.OnFrame(controller);
271         while (true)
272         {
273             prevT = (int)leap.wrist[2];
274             if (count == 0)
275             {
276                 send_message_func();
277                 leap.OnFrame(controller);
278                 centerXpalm = leap.t[1];
279                 f2 = 190 - (int)(k * (Math.Sqrt((double)((leap.cordX3 - leap.cordX2) * (leap.cordX3 - leap.cordX2) + (leap.cordY3 - leap.c
280                 f1 = (int)(k * (Math.Sqrt((double)((leap.cordX3 - leap.cordX4) * (leap.cordX3 - leap.cordX4) + (leap.cordY3 - leap.cordY4)
281                 f4 = 190 - (int)(k * (Math.Sqrt((double)((leap.cordX7 - leap.cordX8) * (leap.cordX7 - leap.cordX8) + (leap.cordY7 - leap.c
282                 f3 = (int)(k * (Math.Sqrt((double)((leap.cordX11 - leap.cordX12) * (leap.cordX11 - leap.cordX12) + (leap.cordY11 - leap.c
283                 f5 = 220 - (int)(k * (Math.Sqrt((double)((leap.cordX17 - leap.cordX14) * (leap.cordX17 - leap.cordX14) + (leap.cordY17 - 1
284                 outputState = ((leap.t[2] - 50) / 2);
285                 rotation = (int)(leap.cordZ11 - leap.wrist[1]) * 4 + 140;
286                 if (f1 <= 0)
287                     mes[0] = 250;
288                 else if (f1 > 250)
289                     mes[0] = 0;
290                 else
291                     mes[0] = (byte)(250 - f1);
292                 f5 = (int)(leap.cordX17 - leap.cordX14) * (leap.cordY17 - 1

```



Проект реализован в формате демонстрационного стенда с манипулятором, имеющим три степени свободы. Также реализовано взаимодействие манипулятора с предметом (подобие ёмкости с радиоактивным веществом).

В ближайшее время планируется добавить четвёртую степень свободы и добавить второй манипулятор. В конечной версии проекта планируется создать камеру, в которой будет находиться два манипулятора, управляемые человеком с другой стороны при помощи нескольких 3D сенсоров.

Любое действие, произведённое оператором “Роборука” может запомнить и повторять его многократно.

Авторы:

Нечаев Даниил

Тюльпанов Александр

Погосов Левон

Руководитель:

Лосицкий И.А.

Санкт-Петербург, 2016