

РСРХРНРØР¤РгРћРѓРѓРѓРќРЧРЄ
РУР¤РѓРћРУРӨРгРѓРћРЎРХР№Р№РќРЄ
РћР№РЧРЎРХРгРѓРЧРћРХРј

РАЗРАБОТКА ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ КОНТРОЛЯ
АВТОМАТИЗИРОВАННОЙ СБОРКИ
МОДУЛЕЙ CMS ДЕТЕКТОРА
БОЛЬШОГО АДРОННОГО
КОЛЛАЙДЕРА

by

Брикса Артём

РУРЧР»РњР«РёР„РёCS СГРёРçР«СЂРё СFP «РЧСFPРёCЂРхРњCS
СFCЂРхР»РхР„РЧ РѓР»РхСЗРЧРёРњРЧСFCЂРё

Р„Рё
PhРёРёCЂRњCÆCЂRхCЂ СГРёРўРЧР«ChРЧРЧРёРЧ РЧ
РёР«РёР»CÆCHCЂRхCГР„СКСЖ СЂRхCЖR„Р«РњР«РўРЧРё
РӨРёChРхРўCГРё СЂRхRњRхРёР«РёРёCЋR„РЧРёРёCЗРЧР€ РЧ
РЧР„ChР«СГРёРёCЗРЧР«Р„Р„СКСЖ СЂRхCЖR„Р«РњР«РўРЧРё

7 июня 2017 г.

Р҃РХРНРоРгРhРfРfРЭРЧР€ РҮРоРgРhРҮРӨРgРhРjРÜРХРnРnРkР€
РhРnРЧРÜРХРgРhРЧРjРХРj

PθP,,P,,P«CTБPθC3PψCS

PhРөРёСТhРnСÆСTРxСT СTРөРүРЧР«ChРЧРЧРЧРЧ РЧ
РэР«РёР»СÆСHСTРxСTР,,СКСЖ СTРxСЖР,,Р«РnР«РүРЧР€
РЭРөChPxРүСTРө СTРxРnРxРёР«РёРёСTР,,РЧРөРёС3РЧР€ РЧ
РЧР,,ChР«СTРёРёС3РЧР«Р,,Р,,СКСЖ СTРxСЖР,,Р«РnР«РүРЧР€

Брикса Артём

Трекер CMS детектора второй стадии модернизации состоит из двух типов модулей (PS и 2S), каждый состоящий из двух кремневых сенсоров, необходимой для их работы электроники и других элементов корпуса. В случае PS модуля, сенсоры должны быть собраны с относительной точностью около 40 мкм. Чтобы выполнить данные требования и сократить время на сборку, предлагается использовать систему автоматизированной сборки. Данная система основывается на высокоточной двигательной установке с интегрированной в ней камерой высокого разрешения, которая позволяет осуществлять распознавание образов, систему фиксации отрицательным давлением и специального программного обеспечения для контроля их работы.

Acknowledgements

Большое спасибо Джеймсу Кивени, Андреасу Муссгиллеру, Дорис Экштайн, Карстену Мул, Адаму Цуберу и всей CMS группе DESY за готовность помочь в любое время и ответить на все мои вопросы, в то время как у меня их было много.

Оглавление

Abstract	1
Acknowledgements	2
List of Figures	5
Abbreviations	7
1 Введение	1
1.1 Компактный мюонный соленоид	1
1.2 Вторая стадия модернизации CMS трекера	2
2 Сборка модулей	5
2.1 Ручная сборка с помощью механической фиксирующей платформы	5
2.2 Система автоматизированной сборки	6
2.3 Сборочная платформа	8
2.4 Быстро сохнущее вещество склеивания.	10
2.5 Процесс автоматизированной сборки	12
3 Control Software	15
3.1 General structure of the application	15
3.1.1 Model View Controller architectural pattern	15
3.1.2 OpenCV library	17
3.2 Pattern recognition	18
3.2.1 Pre-processing of raw image	19
3.2.2 Determination of positions and planar orientations of fiducial markers.	20
3.2.3 Location of other corners and extraction of final position and orientation.	22
3.3 Application functionality	22
4 Precision estimation tests and the first prototype	25
4.1 Pattern recognition precision tests	26
4.1.1 Pattern recognition on the painted corner of a glass dummy	26
4.1.2 Pattern recognition on the marker of the dummy sensor	28
4.2 Pickup precision tests	30
4.2.1 Pickup precision tests without assembly platform	30
4.2.2 Pickup precision tests with assembly platform	31
4.3 First assembled prototype	32

5 Conclusion	36
---------------------	-----------

A Appendix	37
-------------------	-----------

Список иллюстраций

1.1	Вид на слои CMS детектора в поперечном разрезе.	2
1.2	Пример разграничения высокого и низкого поперечного импульса. Частицы, которые пройдут в пределах двух соседних детектирующих полосок второго сенсора от соответствующий полоски первого сенсора, будут считаться как частицы с высоким поперечным импульсом.	3
1.3	PS модуль в разборном виде.	4
1.4	Требования точности к сборке PS и 2S модулей.	4
2.1	Прототип механической фиксирующей платформы.	5
2.2	Предлагаемая система автоматизированной сборки.	6
2.3	Устройства захвата системы автоматизированной сборки.	7
2.4	Эскиз сборочной платформы.	9
2.5	Алюминиевый прототип распорки для PS модуля.	10
2.6	Способ применение быстросохнущего вещества во время тестов. На практике достаточно капель намного меньших размеров для обеспечение необходимой прочности соединения.	11
2.7	Толщина слоя быстросохнущего клеящего вещества Loxeal 31-42 при высыхании под давлением $\approx 20 \text{ g/cm}^2$	12
3.1	A schematic view of the integration of the motion, vision and vacuum subsystems via the Qt-application.	15
3.2	Model, View and Controller (MVC) relative to user.	16
3.3	Model, View and Controller (MVC) relative to user.	17
3.4	Fiducial marker on one of the corner of a PS sensor.	19
3.5	Threshold application on fiducial marker of dummy PS module.	20
3.6	The technique of template matching is illustrated on the left image. The red arrows indicate the iterative calculation of a metric at each point of the master image. The result of a template matching routine on using test images is shown on the right image. The most probable location of the marker in the master image is outlined by the red rectangle.	21
3.7	A schematic illustrating the estimation of the sensors planar orientation is shown on the left image. The red arrows indicate the iterative rotational transformations applied to the master image. graph of the minimised metric value versus the angular transformation applied to the master image is shown on the right image. A clear minimum at ≈ 3.5 degrees is observed.	21
3.8	Screenshot of the Threshold tab of the PSAuto application.	23
3.9	Screenshot of the Assembly tab of the PSAuto application.	23

4.1	Glass sample with silver painted corner.	26
4.2	Distribution of the difference between detected X coordinate of the master image before and after moving the arm in each iteration. $\Delta X \approx 1 \text{ um}$. . .	27
4.3	Distribution of the difference between detected Y coordinate of the master image before and after moving the arm in each iteration. $\Delta Y \approx 2 \text{ um}$. . .	27
4.4	Distribution of the difference between detected angle orientation of the master image relatively to the acquired image before and after moving the arm in each iteration. $\Delta \theta \approx 0.15 \text{ degree}$	28
4.5	The view of the corner after applying the Threshold.	28
4.6	Sensor marker after applying Threshold.	29
4.7	Screenshot of application during precision estimation test with dummy silicon sensor and marker on it.	29
4.8	X and Y movement trends with respect to iteration number in the touch test.	31
4.9	Distribution of the difference between detected X coordinate of the master image before and after moving the arm in each iteration. $\Delta X \approx 4 \text{ um}$. . .	32
4.10	Schematic view of the first module prototype.	32
4.11	Photo of the first assembled prototype.	34
4.12	Microscope photo of the reference corner of the first prototype. Misalignment of the components is around 20 um.	35
A.1	Pattern recognition Flow Chart.	37

Abbreviations

AL-CF	Carbon-Fibre reinforced AL uminium
CF	Carbon- F ibre
CMS	Compact Muon Solenoid
HL-LHC	High Luminosity LHC
LHC	Large Hadron Collider
PS	Pixel Strip (module)
SSBA	Sensor-Spacer-Baseplate-Assembly
2S	2 Strip (module)

Глава 1

Введение

Большой адронный коллайдер (БАК) – крупнейший ускоритель частиц в мире. Во время его первого периода работы с 2010 по 2013 с его помощью были сделаны выдающиеся достижения. Возможно одно из наиболее известных – открытие теоретически предсказанного бозона Хиггса, за что Франсуа Энглер (François Englert) and Питер Хиггс (Peter Higgs) была присуждена Нобелевская премия по физике в 2013 году. Исследование явлений при столкновении частиц высоких энергий БАК происходит в четырёх позициях, расположенных вдоль основного кольца ускорителя и соответствующих четырём детекторам частиц – ATLAS, CMS, ALICE и LHCb [?].

1.1 Компактный мюонный соленоид

Компактный мюонных соленоид (Compact Muon Solenoid или CMS) – это цилиндрический детектор элементарных частиц, разработанный для измерения широкого спектра параметров различных элементарных частиц, производимых во время столкновений (коллизий) протонов и/или тяжёлых ионов БАК. Детектор имеет длину около 28 метров и диаметр около 15 метров. Он является самым тяжёлым детектором элементарных частиц в мире и весит приблизительно 14000 тонн. Своё имя CMS детектор получил благодаря его 3-м ключевым характеристикам: его относительно компактный размер, его исключительная способность в детектировании и измерении мюонов и третья, его основная особенность, – сверхпроводящий соленоид, способный создать магнитное поле до 3.8 Т.

CMS детектор состоит из нескольких слоёв, каждый из которых выполняет свою отдельную роль в детектировании и измерении пролетающих сквозь них элементарных частиц. Схематическое изображение поперечного сечения данных слоёв и их задачи в восстановлении пути элементарной частицы показаны на Рисунке 1.1.

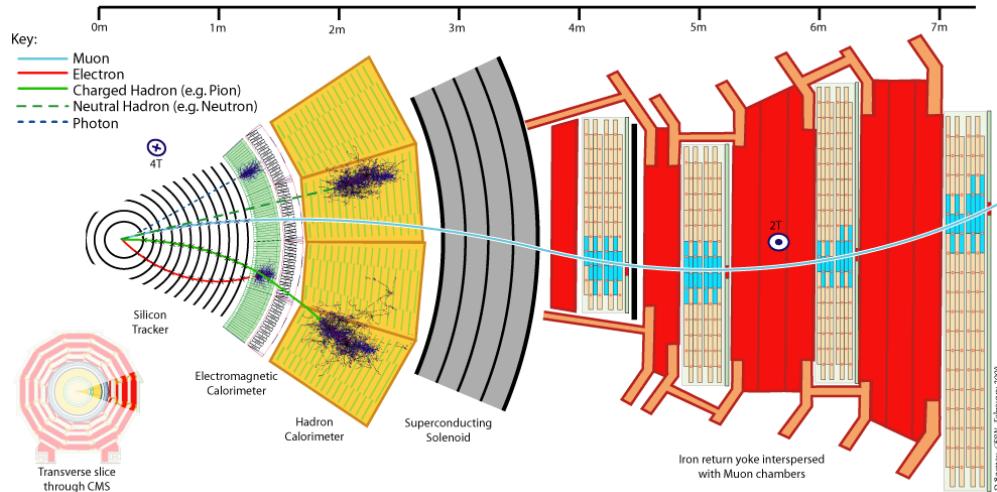


Рис. 1.1: Вид на слои CMS детектора в поперечном разрезе.

1.2 Вторая стадия модернизации CMS трекера

After Phase II Upgrade, the LHC will provide a much higher luminosity. This regime is known as the High Luminosity LHC (HL-LHC). A serious problem presented by these conditions is the enormous data readout rates that exceed far beyond the bandwidth foreseen for the readout electronics. However, the vast majority particles produced in the HL-LHC conditions are not of direct interest for new physics searches and are characterized by low transverse momentum. Thus rejecting tracker “hits” related to low transverse momentum particles can significantly reduce the amount of data to be readout. In order to provide momentum discrimination at the hardware level, a 2-layer module design was created. The central idea of the new modules is to provide fast discrimination between low and high transverse momentum particles by estimating the track curvature caused by the magnetic field within the volume of the module itself. For example, particle with high transverse momentum after hitting some pixel/strip at the first sensor layer would hit on of neighboring pixels/strips of the respective pixel/strip on the second layer. While a particle with low transverse momentum would have a more curved trajectory and hit pixel/strips at a displaced position from the first hit. By varying the distance between sensors and number of neighboring pixel/strips required to match hits in adjacent sensors, (2 neighboring strips in the Figure 1.2) it is possible to set the transverse momentum threshold for a hit [?].

После проведения второй стадии модернизации БАК будет обладать гораздо большей светимостью. Данный режим его работы получил название High Luminosity LHC (HL-LHC) (с англ. – БАК на большой светимости). Серьёзной сложностью условий работы данного режима БАК является является намного больший поток производимых данных, который многократно превосходит пропускную способность

электронных устройств ввода-вывода. Однако подавляющее большинство элементарных частиц, образованных в результате работы HL-LHC малоинтересны в текущих исследованиях физики высоких энергий и характеризуются низким поперечным импульсом. Таким образом игнорируя попадания элементарных частиц с низким поперечным импульсом, можно значительно уменьшить объём считываемых данных. С целью обеспечения возможности распознавания момента импульса на уровне аппаратного обеспечения был создан проект двухслойных детектирующих модулей. Основная идея работы новых модулей состоит в разграничении проходящих частиц с низким и высоким поперечным импульсом основываясь на приближенном измерении кривизны траектории вызванной магнитным полем, в котором находится модуль. Например, частица с высоким поперечным импульсом после прохождения детектирующего пикселя/полоски первого сенсора пройдёт через соответствующий пиксель/полоску соседнего второго сенсора. В то время как частица с низким поперечным импульсом будет иметь большую кривизну траектории и пройдёт немного в стороне от соответствующего элемента первого сенсора на втором сенсоре. Варьируя расстояние между сенсорами и количество соседствующих детектирующих пикселей/полосок необходимых для сопоставления областей прохождения на смежных сенсорах модуля (2 соседних детектирующих полоски на Рисунке 1.2), возможно выставить пороговое значение поперечного импульса для проходящей частицы [?].

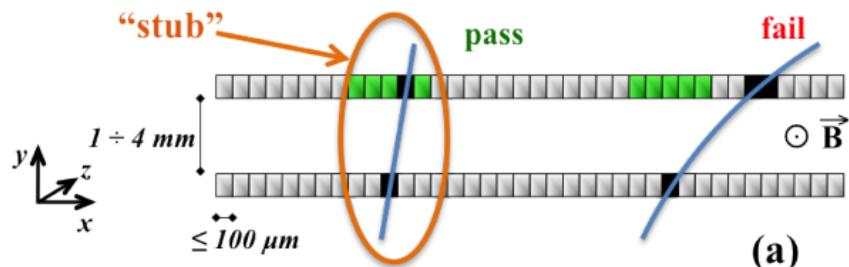


Рис. 1.2: Пример разграничения высокого и низкого поперечного импульса. Частицы, которые пройдут в пределах двух соседних детектирующих полосок второго сенсора от соответствующей полоски первого сенсора, будут считаться как частицы с высоким поперечным импульсом.

Двухслойный детектирующий модули

Трекер CMS детектора второй стадии модернизации будет использовать два типа детектирующих модулей – 2S модули и PS модули. Чтобы достигнуть эффективного распознавания частиц с низким поперечным импульсом по всему объёму трекера, в различных его частях сенсоры модулей будут располагаться на разном расстоянии друг от друга. Для 2S (PS) модулей расстояние будет варьироваться между 1.8 и 4 мм (1.6, 2.6 - 4 мм). PS модуль в разборном виде показан на Рисунке 1.3.

В PS модуле сенсоры приклейены к распоркам изготовленных из алюминия, усиленного углеродным волокном. Они обеспечивают необходимую теплопроводность,

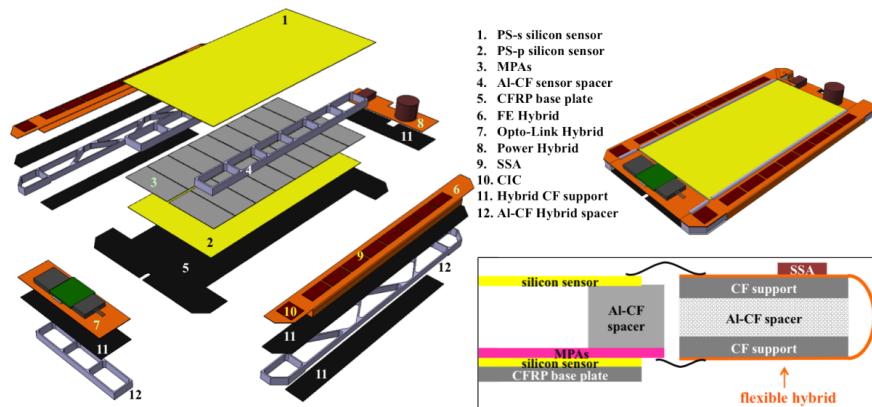


Рис. 1.3: PS модуль в разборном виде.

остро необходимую для охлаждения модуля. Два сенсора вместе с распорками в свою очередь приклеены к опорной пластине из углеродного волокна. Эта структура впоследствии будет называться Сенсор-Распорка-Сенсор-Опора (СРСО). Данная работа ориентирована исключительно на сборку СРСО. Требуемые допущения по точности сборки СРСО показаны на Рисунке 1.4. Для PS модуля сенсоры должны быть расположены с относительной точностью в 40 мкм измеряя с короткой стороны. Данная точность соответствует максимальной неточности в относительном вращении сенсоров на величину 0.8 мрад [?].

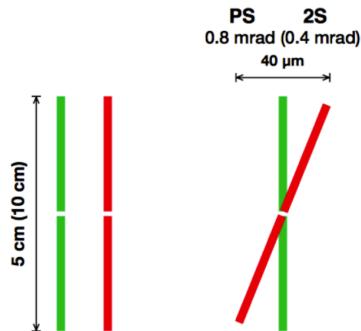


Рис. 1.4: Требования точности к сборке PS и 2S модулей.

Глава 2

Сборка модулей

Сборку высокоточных детектирующих модулей такого рода обычно проводят вручную используя изготовленные на заказ специальные механические фиксирующие платформы. В качестве альтернативы данному привычному методу предлагаются использовать систему автоматизированной сборки.

2.1 Ручная сборка с помощью механической фиксирующей платформы

Один из способов сборки модулей – ручная сборка с помощью изготовленной на заказ специальной механической фиксирующей платформы, прототип которой показан на Рисунке 2.1) [?].

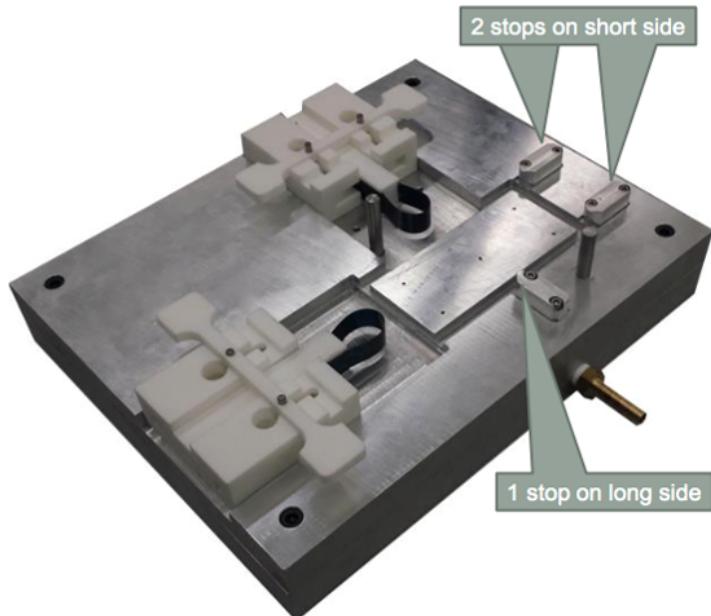


Рис. 2.1: Прототип механической фиксирующей платформы.

При таком способе сборке все составные части размещаются на платформе и собираются вручную. Для обеспечения необходимой точности сборки фиксирующая платформа имеет три опорных стопора: один на длинной стороне и два вдоль более короткой стороны. С противоположных сторонам собирайемый модуль аккуратно прижимается специальными пружинами. Вместе они обеспечивают необходимую точность позиционирования компонентов модуля перед и во время сборки/склеивания.

Однако, описанный выше способ сборки обладает рядом недостатков. Для сборки с его помощью необходимо много времени и данная механическая система плохо масштабируется на промышленные масштабы сборки. Также данный метод обладает относительно плохой повторяемостью собранных модулей, пусть и в пределах необходимой точности, и не предоставляет возможностей контроля процесса сборки. Более того, фиксирующая платформа выставляет высокие требования качества её изготовления (точность выточки деталей должна составлять несколько микрон), а также стопоры требуют регулярной калибровки из-за возможных сдвигов и/или стачиваний их поверхностей. Наконец, данная механическая система требует максимум участия человека в её работе. Несмотря на расчётную точность системы в теории, факт участия человека в сборке означает, что всегда будет существовать какой-то процент бракованных модулей, произведённых только по вине человека.

2.2 Система автоматизированной сборки

Предлагаемая система автоматизированной сборки состоит из трёх подсистем: двигательной подсистемы, подсистема наблюдения, подсистема отрицательного давления (Figure 2.2)[?].

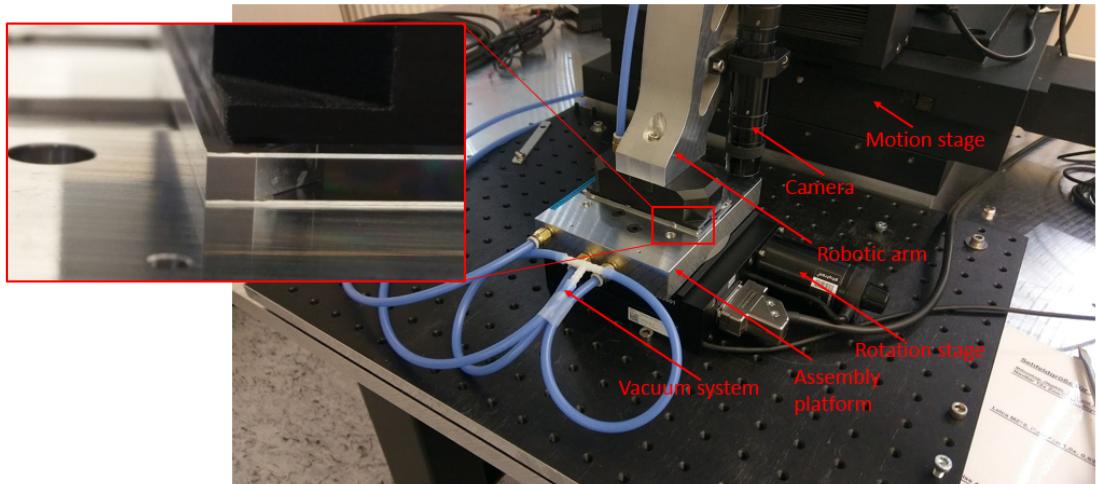


Рис. 2.2: Предлагаемая система автоматизированной сборки.

Двигательная подсистема обеспечивает точные передвижения необходимые для расположения компонентов, из которых состоит СРСП. Движение обеспечивается благодаря двум подвижным элементам двигательной подсистемы, обеспечивающим, первый, по трём координатам и, второй, вращение. Специально изготовленное приспособление из алюминия, далее именуемое "роборукой смонтированной на тот элемент двигательной подсистемы, который обеспечивает движение по трём координатам, таким образом позволяя перемещать компоненты модуля в прямоугольных координатах. Они размещаются на вращательной установке, осуществляющей вращение в плоскости xOy той же прямоугольной системы координат. Элементы двигательной подсистемы управляются специальным блоком управления. Вся аппаратная часть двигательной подсистемы произведена фирмой Lang, которая точность перемещений до 4 мкм и 2 мрад соответственно. *Подсистема отрицательного давления* позволяет закреплять компоненты собираемого модуля на роборуке и вращательной установке. Она состоит из одного насоса, подводящей отрицательное давление к четырём переключаемым клапанам, которые в свою очередь распределяют отрицательное давление по четырём независимым линиям. Переключение клапанов осуществляется путём приложения управляющих сигналов напряжением в 12 В (12 В – включенное состояние, 0 В – выключенное). Управляющие сигналы обеспечиваются специальной релейной картой. Одна линия отрицательного давления подведена к устройству захвата компонентов модуля, крепящегося на конце роборуки, другие – к сборочной платформе. Захватное устройство сделано из ESD пластика, в котором располагается внутренняя камера для распределения подключённого отрицательного давления по специальным направленным вниз присоскам, которые слегка выступают за нижнюю грань захватного устройства. Схематичное изображение устройства захвата отрицательным давлением показано на Рисунке 2.3.

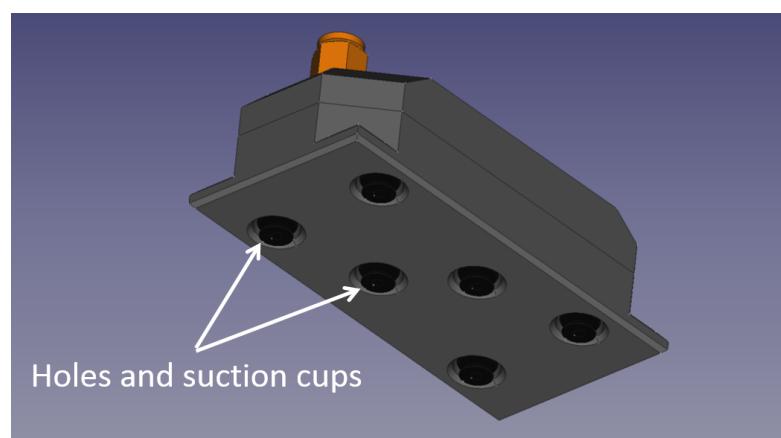


Рис. 2.3: Устройства захвата системы автоматизированной сборки.

Процедура захвата происходит путём касания присосками сенсора, включения подачи отрицательного давления устройству захвата и непосредственному перемещению роборуки вверх с захваченным сенсором. Процедура высвобождения сенсора

начинается с контакта нижней поверхности прикреплённого сенсора (или опорной пластины) поверхности на которую необходимо поместить прикреплённый компонент, выключения подачи отрицательного давления и отвода роборуки в сторону. С целью избежания нежелательных перемещений компонента модуля во время увода роборуки, на сборочной платформе компонент фиксируется специальными присосками снизу, принцип работы которых полностью схож с фиксацией на устройстве захвата.

Подсистема наблюдения делает снимки компонентов, проводит распознавание их расположения и ориентации в пространстве, что является особенно важным для осуществления точной сборки модуля. Она представлена камерой высокого разрешения фирмы IDS и крепится к роборуке. Она зафиксирована в положении, направленном вниз. Основной задачей камеры является делать снимки компонентов до и после их захвата и опускания вниз для последующей обработки этих снимков специализированным программным обеспечением, которое точно определяет положение компонента и его ориентацию.

2.3 Сборочная платформа

Одной из наиболее важных частей системы автоматизированной сборки является сборочная платформа. Её основной задачей является фиксация модулей во время сборки модуля.

Сборочная платформа должна выполнять следующие требования:

1. Фиксация всех необходимых компонентов с помощью отрицательного давления.
2. Обеспечивать возможность расположения двух распорок модуля строго параллельно и на определённом расстоянии.
3. Быть по-возможности лёгкой и иметь центр масс как можно ближе к оси вращения вращательной установки, на которой платформа будет установлена.

Чтобы выполнить упомянутые выше требования, следующий дизайн был предложен (Рисунок 2.4)

Сборочная платформа имеет два углубления, в каждом из которых находятся по три опорных стопора обеспечивающих точное позиционирование распорок модуля. В сравнении с механической фиксирующей платформой, стопоры данной сборочной платформы являются частью платформы, а не прикручиваются отдельно, таким образом они не нуждаются регулярной в калибровке. Однако, наличие данных углублений делает невозможным размещение других компонентов модуля (сенсоров опорной пластины) непосредственно поверх них, так как слишком большая часть их площади остаётся без поддержки под давлением устройства захвата сверху во время склеивания компонентов модуля. Данная проблема была решена путём размещением плоских компонентов модуля (сенсор, опорная пластина) перпендикулярно распоркам на платформе. В данном случае только небольшая часть

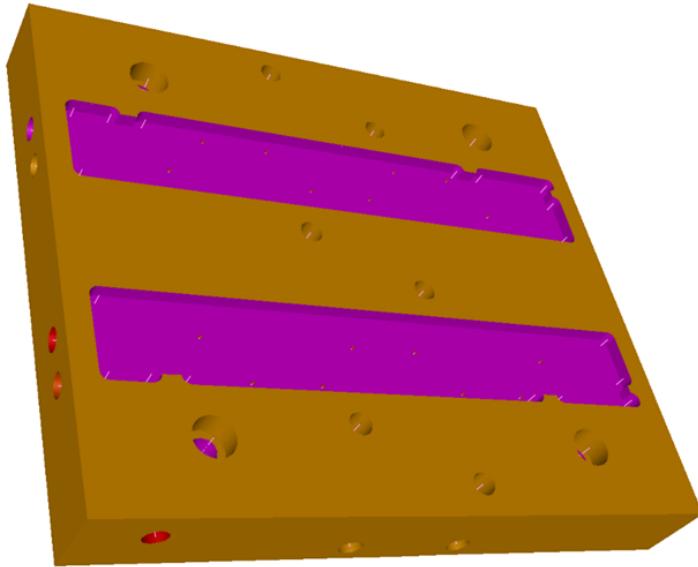


Рис. 2.4: Эскиз сборочной платформы.

данных компонентов не будет иметь поддержки снизу, что не является критичным. Перпендикулярное размещение некоторых компонентов собираемого модуля может быть с лёгкостью осуществлено вращательной платформой, на которой находится сборочная платформа. Центр масс спроектированной платформы находится очень близко к оси вращения – меньше 1 мм. В то время как её масса составляет около 1 кг, таким образом никаких нежелательных эффектов влияющих на точность работы вращательной платформы не ожидается.

Внутри сборочной платформы располагаются две независимые линии отрицательного давления: первая – для удержания распорок, вторая – для удержания сенсоров и опорной пластины. Линия распорок распределяет давление по ряду отверстий (0.7 мм в диаметре) на дне углублений для распорок. Размер и расположение данных отверстий определяется формой распорки (Рисунок 2.5). Эти отверстия не оборудованы присосками из-за своего очень малого размера. Присосок таких размеров попросту нету на рынке. Однако, соприкасающиеся поверхности (дно углублений платформы и соответствующая граня распорки) имеют достаточную плоскость, чтобы утечка "вакуума" была в допустимых пределах и распорки были зафиксированы достаточно плотно.

Вторая линия отрицательного давления распределяет его между присосками, удерживающими сенсоры или опорную пластину. В сборочной платформе используются присоски, абсолютно идентичные захватывающему устройству. Так же, как и в захватывающем устройстве, они слегка выступают за поверхность сборочной платформы. Однако, они не должны препятствовать процессу сборки, а именно этапу приклеивания сенсора к распоркам (этапы сборки будут подробно описаны позднее в данной Главе). Именно поэтому высота углублений меньше толщины распорок, что

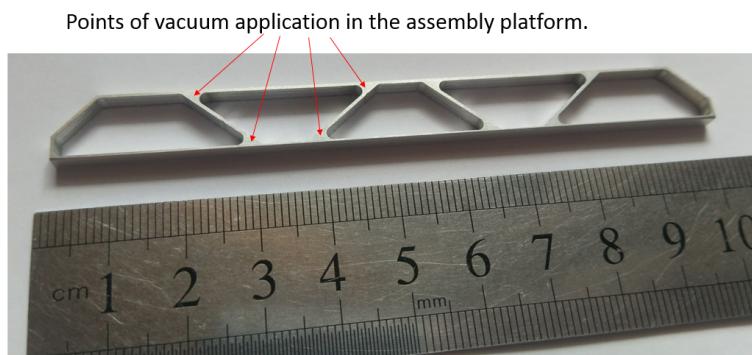


Рис. 2.5: Алюминиевый прототип распорки для PS модуля.

позволяет верней части распорок быть выше верхней части присоски. Другими словами, присоски не препятствуют этап сборки, в которых не участвуют, так как находятся ниже верхней части распорок.

2.4 Быстросохнущее вещество склеивания.

Другой важной частью системы автоматической сборки является склеивающее вещество, используемое во время сборки, говоря точнее – время его высыхания. Другими словами, мало смысла оставлять один модуль на длительное время зафиксированным в установке, просто ожидая пока высохнет склеивающее вещество. Например, основное клейкое вещество, используемое в сборке модуля, требует 24 часов для высыхания. Именно поэтому система автоматизированной сборки нуждается в технологии, которая позволит избежать длительного ожидания. Одно из предложенных решений – использование небольшого количества быстросохнущего клейкого вещества в дополнении к основному. Данное вещество должно удовлетворять следующим требованиям:

1. Обеспечивать необходимую прочность соединения спустя 15 минут применения.
2. Не вступать во взаимодействие с основным kleящим веществом.
3. Обеспечивать тонкий слой (до 30 мкм).

Более того, упомянутые выше требования должны быть выполнены с использованием как можно меньшего количества вещества. Для тестов использовались небольшие алюминиевые бруски формы прямоугольного параллелепипеда (в качестве приближения распорок и усиленного карбоновым волокном алюминия) и образцы стекла (в качестве приближения кремневых сенсоров). Достигение перечисленных требований в высокой степени зависит от двух факторов: первый – способ применения kleящего вещества, второй – непосредственный свойства вещества.

Возможны множество способов применения быстросохнущего kleящего вещества, среди них: несколько его капель внутри слоя основного kleящего вещества,

несколько его капель по краям соединения, заполнение скоса распорок быстросохнущим kleящим веществом, несколько его капель поверх края соединения и др. Для описываемой системы было принято проверить заполнение скоса стопоров быстросохнущим kleящим веществом и несколько его капель по краям соединения. Несмотря на то, что проклеивания по скосу может обеспечить необходимую прочность соединения после 15 минут, намного практичнее применять несколько капель вещества по краям соединения. Оно применялось способом, показанным на Рисунке 2.6.

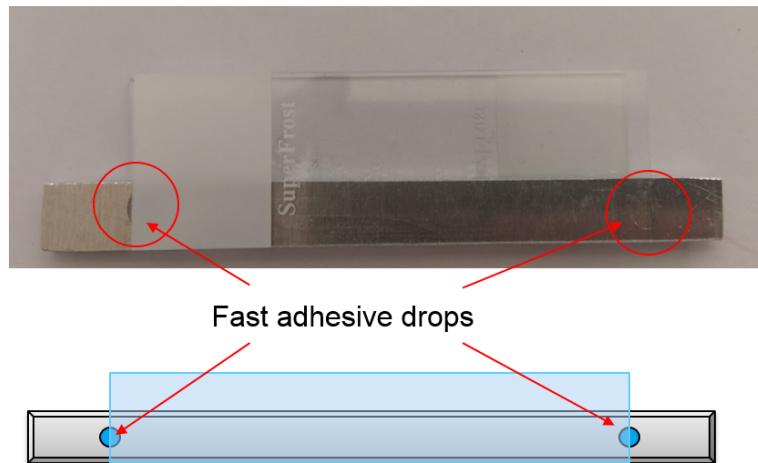


Рис. 2.6: Способ применения быстросохнущего вещества во время тестов. На практике достаточно капель намного меньших размеров для обеспечение необходимой прочности соединения.

Вторая часть вопроса – поиск подходящего быстросохнущего вещества на рынке. Основная сложность вопроса состоит в комбинации двух свойств: *быстрое высыхание и низкая вязкость*. Низкая вязкость есть неотъемлемой свойство kleящего вещества, если необходимо обеспечить тонкий слой вещества. Однако, быстрое высыхание означает, что вещество должно стать твёрдым быстро, чего в свою очередь намного проще достичь, если вещество изначально будет большей вязкости. Несмотря на данные сложности, были подобраны несколько кандидатов для тестирования:

1. *Polytec EP 660*. Согласно технической документации [?], время его высыхания составляет до 16 часов, что выходит далеко за рамки требований. Однако, производитель отметил, что необходимая прочность соединения может быть достигнута без полного высыхания. К сожалению, данное вещество, ожидаемо, не обеспечило необходимое соединение по прошествии 15 минут. Единственным его преимуществом является факт того, что основное kleящее вещество было изготовлено тем же производителем.
2. *Loxéal 31-42*. Согласно технической документации [?], время его полного высыхания составляет $\approx 20\text{--}30$ минут, в то время как время "схватывания" составляет около 3-8 минут. В результате после 15 минут данное вещество обеспечивает необходимую прочность соединения.

3. *Wekem WK5.* Согласно технической документации [?], время его высыхания составляет около пяти минут. Данный образец также обеспечивает необходимое качество соединения после 15 минут высыхания, но kleящее вещество от Loxeal имеет лучшее качество и удобнее в применении. Более того, Loxeal 31-42 обеспечивает более тонкий слой при высыхании под одинаковым давлением – < 20 – в то время как Wekem – $\approx 40 \text{ } \mu\text{m}$ (Рисунок 2.7).

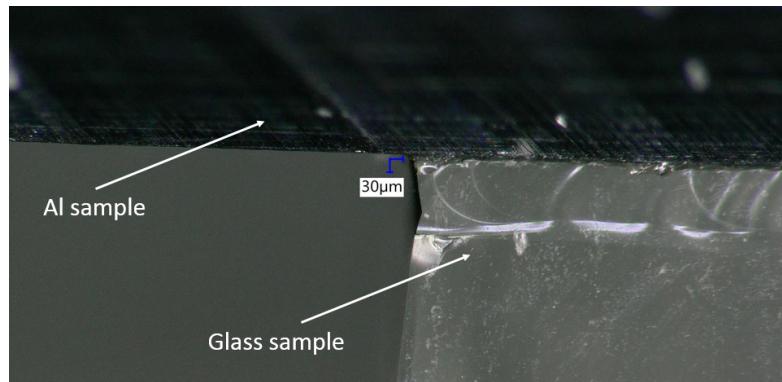


Рис. 2.7: Толщина слоя быстросохнущего kleящего вещества Loxeal 31-42 при высыхании под давлением $\approx 20 \text{ g/cm}^2$

Подводя итоги, Loxeal 31-42 показал лучшие результаты, удовлетворив все требования: он обеспечивает необходимую прочность соединения после 15 минут высыхания под небольшим давлением (around 20 g/cm^2), создавая при этом тонкий слой – < 20 , и демонстрирует отсутствие взаимодействия с основным kleящим веществом.

2.5 Процесс автоматизированной сборки

Процесс автоматизированной сборки СРСП для PS модуля состоит из следующих шагов:

1. *Подготовка верхнего сенсора.* Сперва необходимо поместить верхний сенсор на сборочную платформу и зафиксировать отрицательным давлением. После распознать его позицию и ориентацию в пространстве с помощью камеры и специального программного обеспечения и сохранить данную информацию. Затем захватить и поднять его с помощью устройства захвата и оставить прикреплённым. Начиная с самого первого шага подача отрицательного давления на устройство захвата не прекращается в течении всего процесса сборки.
2. *Подготовка распорок.* Как только верхний сенсор прикреплён к устройству захвата и поднят с платформы, сборочная платформа должна повернуться на 90 градусов таким образом, что распорки будут ложиться в специально отведённый для них углубления корректно ориентированными относительно верхнего сенсора. Далее их необходимо аккуратно прислонить к опорным стопорам и включить подачу отрицательного давления для их фиксации. После

система должна найти и определить положение опорного маркера на сборочной платформе, что позволит однозначно рассчитать положение распорок в пространстве. Отрегулировать их ориентацию с помощью вращательной установки таким образом, чтобы они были параллельны оси Ох двигательной установки, что означает быть сонаправленным верхнему сенсору.

3. *Склейивание верхнего сенсора с распорками.* Как только верхний сенсор и распорки готовы, их положения и ориентации определены, программное обеспечение может вычислить путь для устройства захвата с удерживаемым им верхним сенсором к распоркам. После необходимо поместить основное kleящее вещество на распорки и несколько капель быстросохнущего вещества на краях распорок и передвинуть устройство захвата в расчётную позицию для склеивания с распорками. Особое внимание стоит уделить движению устройства захвата по Z координате, так как это напрямую влияет на толщину слоя kleящего вещества.
4. *Подготовить нижеский сенсор.* По истечении 15 минут капли быстросохнущего вещества обеспечивают необходимое соединение для извлечение склеенных распорок и верхнего сенсора. Для этого необходимо сначала отключить подачу отрицательного давления, удерживающего распорки, и после просто поднять устройство захвата, которое уже будет удерживать не только верхний сенсор, но и приклеенные к нему распорки. После того, как платформа очищена её следует повернуть на 90 градусов так, что нижний сенсор может быть помещён на платформу и зафиксирован отрицательным давлением. Далее необходимо так же, как и для верхнего сенсора, определить его положение и ориентацию с помощью камеры и специального программного обеспечения и сохранить данную информацию. Отрегулировать ориентацию сенсора с помощью вращательной установки таким образом, чтобы он был параллелен оси Ох двигательной установки, что означает быть сонаправленным верхнему сенсору и распоркам.
5. *Склейивание нижеского сенсора с распорки + верхний сенсор.* Все необходимые для данного этапа компоненты зафиксированы и их расположение известно. Следовательно программное обеспечение способно рассчитать путь для устройства захвата и переместиться в позицию склеивания после нанесения главного и быстросохнущего kleящего вещества.
6. *Подготовка опорной пластины.* По прошествии 15 минут склеенная структура может быть поднята. Далее необходимо поместить опорную пластину на сборочную платформу согласно трём опорным штифтам и отверстий для них в самой пластине и зафиксировать её с помощью отрицательного давления. Данные штифты не представлены в текущей версии сборочной платформы.
7. *Склейивание сенсор-распорки-сенсор структуры с опорной пластиной.* Так как

программное обеспечение уже знает расположения опорного маркера на сборочной платформе, следовательно оно способно вычислить позицию склеивания для текущего этапа. Далее устройство захвата передвигается в эту позицию после нанесения основного и быстросохнущего клеящего вещества.

8. *Автоматическая сборка завершена.* По прошествии 15 минут быстросохнущее вещество обеспечивает необходимую прочность соединение и собранный СРСП может быть убран со сборочной платформы и оставлен в другом месте до полного высыхания склеивающих веществ на 24 часа. Однако опорные штифты опорной пластины очень плотно подходят самой пластине, следовательно ручное снятие СРСП может вероятно вызвать его разрушение. Поэтому лучше доверить и данный этап двигательной установке, которая может безопасно поднять собранный СРСП вертикально вверх, сняв его с опорных штифтов.

Глава 3

Control Software

In order to control the whole automated assembly system a PC-based Qt application was developed. All the necessary hardware is connected to the PC with USB interface.

3.1 General structure of the application

The control of the motion, vision and vacuum subsystems is integrated in a single software application henceforth referred to as PSAuto (from automated assembly of PS module). It is entirely written in C++ and utilises the Qt framework version 4.8.7. A schematic illustrating the integration of the subsystems with PSAuto is shown in Figure 3.1.

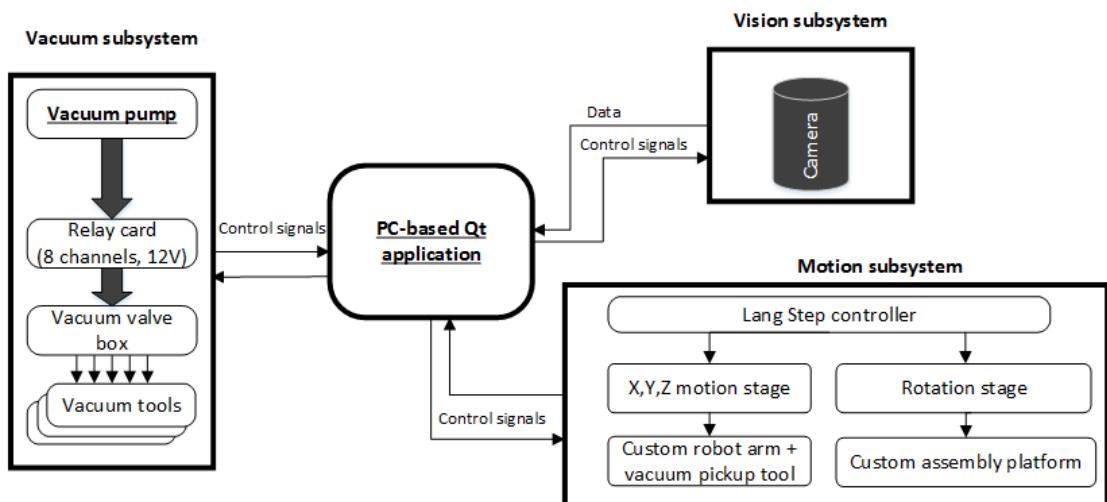


Рис. 3.1: A schematic view of the integration of the motion, vision and vacuum subsystems via the Qt-application.

3.1.1 Model View Controller architectural pattern

The architecture of the application is based on Model-View-Controller (MVC) architectural pattern. It considers there to be three main types of objects: *model* objects, *view* objects

and *controller* objects. When designing an application, a major step is choosing or creating custom classes for objects that fall into one of these three groups. Each of the three types of objects is separated from the others by abstract boundaries and communicates with objects of the other types across those boundaries. The pattern defines not only the roles objects play in the application, it also defines the way objects communicate with each other [?]. The key point of MVC is that View and Controller depend on Model, but Model does not depend on them. The interaction of these three types is schematically shown in the Figure 3.2.

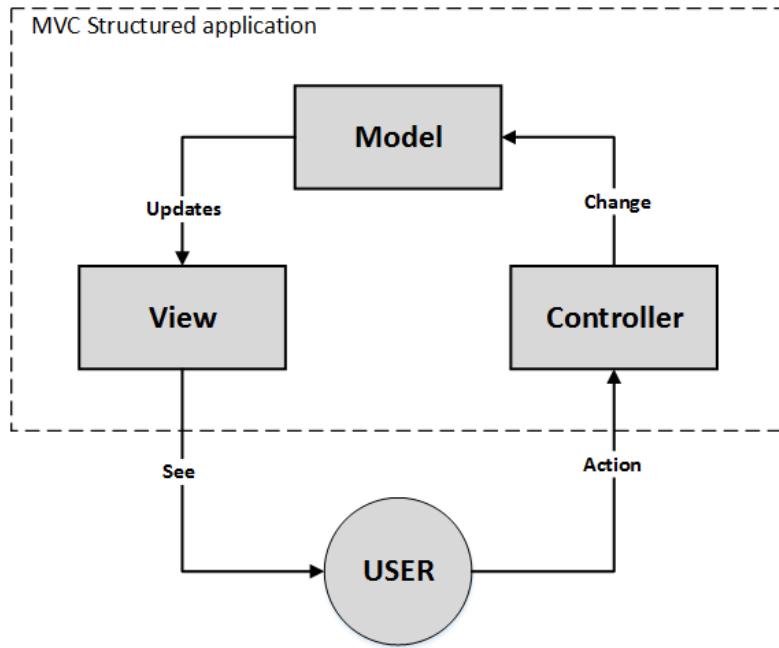


Рис. 3.2: Model, View and Controller (MVC) relative to user.

Shown in the Figure 3.2 interaction is a classical MVC architecture. There are various realization of this diagram in terms of interaction between three main object types of MVC. The main reasons of it are various application types and their realizations. For instance, in the PSAuto application this diagram will look like in the Figure 3.3.

Comparing to classical MVC structure from Figure 3.2, the Model do not directly inform/update the View. This information passes through Controller. A controller object acts as the intermediary between the Model and the View. Controllers are often in charge of making sure the views have access to the model objects they need to display and act as the conduit through which views learn about changes to the model. Controller objects can also perform set-up and coordinating tasks for an application and manage the life cycles of other objects [?].

Depending on the application logic and demands, Model, View and Controller follow special properties and rules.

Model is central component of the system. It directly manages the data, defines the logic and rules of the application which manipulates the date. Moreover, like in our case, it regulates the interaction between the application and hardware. For instance, there is

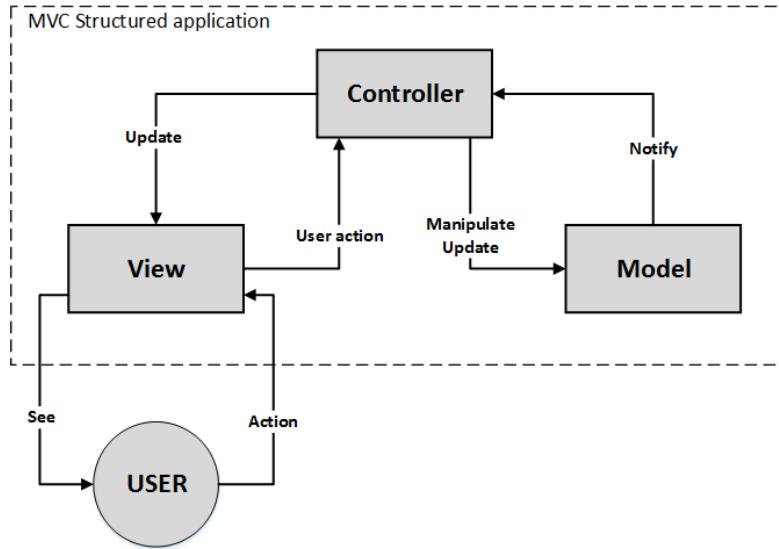


Рис. 3.3: Model, View and Controller (MVC) relative to user.

a Model class *ConradModel* in PSAuto application. It is responsible for communication with a relay card for turning on and off vacuum lines. This class completely meet MVC rules as a Model object because it does not depend on any other class, while others depend on it using its possibilities for their functions.

View unite all objects responsible for visual part of the application: windows, tabs, labels, forms, buttons, etc. They know how to display the data from the application's model and also give a user the possibility to edit the data. For instance, one task of *AssemblyModuleAssembler* class is to display the information about Vacuum lines of the system, as well as control elements for it. However, it only display the control elements, but not process the user's actions.

Controller is mainly responsible for accepting input and processing it, generating commands for the Model or the View. For instance, *ConradManager* provides all necessary functions to control the vacuum lines and to get the current status of it. It is worth noting that very often there is no direct connection between the Model and the View, like it is shown in the classical MVC architecture in the Figure 3.2. Instead of it, the Controller acts as the intermediary between the application's model objects and its view objects.

3.1.2 OpenCV library

A lot of algorithms needed for the application were found in the open-source library — OpenCV (*Open Source Computer Vision*). It is a cross-platform library of programming functions mainly oriented on real-time computer vision. It was originally developed by Intel's research center in Nizhny Novgorod (Russia). Later it was supported by Willow Garage and is now maintained by Itseez. The library is free for use under the open-source BSD license [?].

OpenCV was designed for computational efficiency and, as was already mentioned, with a strong focus on real-time applications. It is written in optimized C++ and can

take advantage of multicore processors. There is also a possibility for further automatic optimization on Intel architecture with Intel's *Integrated Performance Primitives (IPP)* libraries, which consist of low-level optimized routines in many different algorithmic areas [?].

One of the main goals of OpenCV is to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly. In its libraries one can find over 500 functions that span many areas in vision. In the list below one can see main features of the OpenCV library [?]:

- Image data manipulation (allocation, release, copying, setting, conversion).
- Image and video I/O (file and camera based input, image/video file output).
- Matrix and vector manipulation and linear algebra routines (products, solvers, eigenvalues, SVD).
- Various dynamic data structures (lists, queues, sets, trees, graphs).
- Basic image processing (filtering, edge detection, corner detection, sampling and interpolation, color conversion, morphological operations, histograms, image pyramids).
- Structural analysis (connected components, contour processing, distance transform, various moments, template matching, Hough transform, polygonal approximation, line fitting, ellipse fitting, Delaunay triangulation).
- Camera calibration (finding and tracking calibration patterns, calibration, fundamental matrix estimation, homography estimation, stereo correspondence).
- Motion analysis (optical flow, motion segmentation, tracking).
- Object recognition (eigen-methods, HMM).
- Basic GUI (display image/video, keyboard and mouse handling, scroll-bars).
- Image labeling (line, conic, polygon, text drawing).

The application of the OpenCV libraries will be described more detailed in the next section.

3.2 Pattern recognition

Pattern recognition is a crucial part of the whole automated assembly system, in particular – vision subsystem. It provides the software with the information where all the components of the modules are situated in the space and theirs orientation. Within this information the software is able to calculate where to move each component of an assembling module [?].

An important task performed by PSAuto is the determination the *location* and *planar orientation*. By definition, planar orientation is the rotational orientation of the sensor in the horizontal (X-Y) plane of the sensors during the assembly process. This is generally achieved in two basic steps: the independent determination of *localised* positions and orientations of the four markers at the corners of the sensor and a global fit to these positions from which the final position and orientation of the entire sensor is extracted [?].

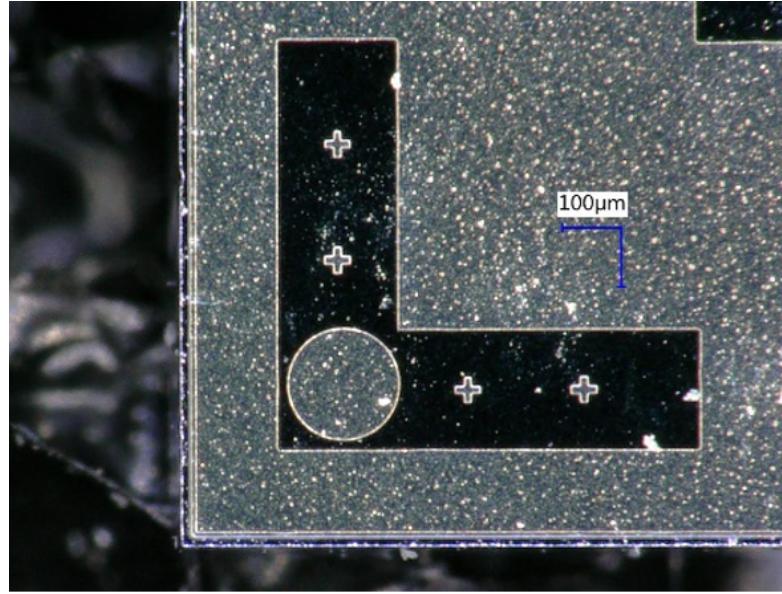


Рис. 3.4: Fiducial marker on one of the corner of a PS sensor.

This is done by processing the images of *fiducial markers* at the corners of the sensors (Figure 3.4) acquired by the vision system with a Pattern Recognition algorithm. The markers are precisely positioned with respect to the strips or pixels or the sensors. Hence precise alignment of the markers ensures precise alignment of the pixels or strips. The algorithm takes raw images as input and returns the position and orientation of fiducial markers located at the corners of the PS sensors. The pattern recognition algorithm utilises the *OpenCV* package. The steps comprising the pattern recognition are now outlined:

1. Pre-processing of raw image.
2. Determination of positions and planar orientations of fiducial markers.
3. Location of other corners and extraction of final position and orientation.

3.2.1 Pre-processing of raw image

The raw images from the camera are first converted from colour to black and white which is known as *grayscaleing* in OpenCV. The pixels comprising the grayscaled image contain intensity information only in the form of a single number ranging from 0 to 255 describing the darkness of a shade of gray as opposed to the intensity and colour information contained in the pixels of a colour image. As the corner markers are based on simple shapes and not colours, the colour information is not helpful and is thus disregarded. The grayscaled image is then converted to a *binary* image in which each pixel is either black or white in a process known as *thresholding*. Thresholding simply converts each pixel of the image into a white(black) pixel if the intensity of the pixel is above(below) a pre-defined threshold (.). Thresholding serves to reduce the differences between images of identical sensors due to random *noise* arising from dust and random differences in the surfaces of the patterned sensors. Examples of grayscaled

and thresholded images of the same corner marker of a dummy PS sensor is shown in Figure 3.5. The optimal threshold will depend on the ambient lighting conditions around the assembly area and the contrast of the fiducial markings on the sensors. Currently, typical values of threshold are around 90.

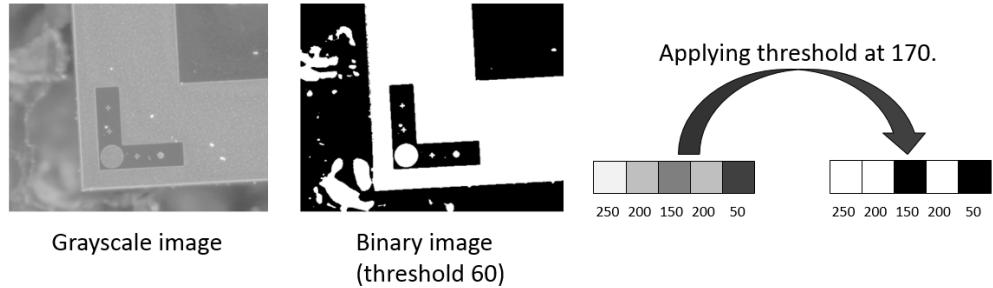


Рис. 3.5: Threshold application on fiducial marker of dummy PS module.

3.2.2 Determination of positions and planar orientations of fiducial markers.

The position and orientation of the fiducial marker within the thresholded image is determined using a standard image processing technique known as *template matching*. In template matching, parts of a master image that closely resemble a template image are located. In this case, the master image corresponds to the thresholded image of the fiducial marker and the template image corresponds to a thresholded image of a fiducial marker.

Template matching proceeds by iteratively superimposing the template image at each point of the master image and calculating a metric which describes the similarity of the template image and the portion of the master image with which it coincides. The OpenCV package provides multiple options for the metric. Similar results are observed for each possible metric with the chosen metric based on the normalised squared difference between the intensities of coincident pixels of the master image and superimposed template:

$$R(x, y) = \frac{\sum_{x',y'} (T(x', y') - I(x + x', y + y'))^2}{\sqrt{\sum_{x',y'} T(x', y')^2 \cdot \sum_{x',y'} I(x + x', y + y')^2}}$$

where I denotes *master image*, T – *template image* and R – *resultant metric*.

This metric is referred to as CV_TM_SQDIFF_NORMED in OpenCV. The point in the master image where the metric reaches a minimum represents the most probable location of the fiducial marker. In Figure 3.6 the determination of marker position with template matching and its result using images of the dummy sensor is illustrated. The most probable location of the marker as determined by the algorithm is indicated by the white rectangle. The observed location matches closely with expectation.

In order to deduce the orientation of the marker in the plane transverse to the optical axis of the camera, the matching procedure is repeated iteratively with different

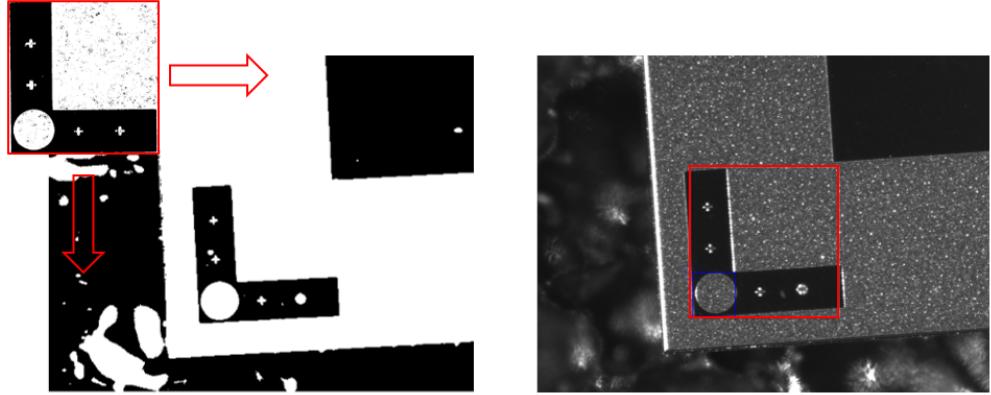


Рис. 3.6: The technique of template matching is illustrated on the left image. The red arrows indicate the iterative calculation of a metric at each point of the master image. The result of a template matching routine on using test images is shown on the right image. The most probable location of the marker in the master image is outlined by the red rectangle.

rotational transformations applied to the master image. For each iteration, the minimal value of the metric is recorded with the minimal metric value across all iterations denoted as α . The planar orientation of the sensor is estimated as $-\alpha$. In Figure 3.7 a schematic illustrating the determination of the orientation is shown. A graph of the resultant minimised metric values versus the size of the angular transformation applied to the master image is shown on the right image of the Figure 3.7. The graph corresponds to a test extraction performed with images of a dummy sensor where the sensor in the master image had a planar orientation of ≈ 3.5 degrees. A clear minimum is observed at ≈ 3.5 demonstrating the method's validity. More precise determination of the sensor orientation can be achieved when factors such as ambient light conditions, image focus and marker design are further optimised. The best accuracy reached during tests was ≈ 0.025 degrees.

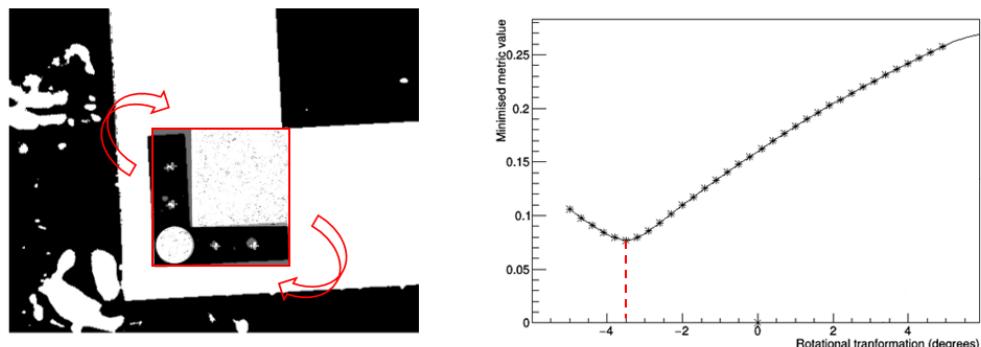


Рис. 3.7: A schematic illustrating the estimation of the sensors planar orientation is shown on the left image. The red arrows indicate the iterative rotational transformations applied to the master image. graph of the minimised metric value versus the angular transformation applied to the master image is shown on the right image. A clear minimum at ≈ 3.5 degrees is observed.

3.2.3 Location of other corners and extraction of final position and orientation.

The procedure described in step 2 is repeated at each sensor corner. The planar orientations determined at given corner are used to set the direction of movement needed for the motion stage to automatically travel to an adjacent corner. The final position and orientation of the sensor is determined by a χ^2 fit to the four (x,z) points. The orientation determined from the fit is cross-checked with the estimations of the orientation at the corners which accuracy could be not far from χ^2 deduced. If there is agreement between the fit and four corner orientations, the fit results are used.

Detailed Flow Chart of the pattern recognition is shown in the Appendix ??.

3.3 Application functionality

The main window of the application has several buttons and check boxes on top of the window and a number of tabs including: Finder, Threshold, Assembly, Autofocus, Motion Manager and others.

1. *Finder*. A simple tab the only purpose of each is to show the last acquired image of the camera. This tab also provide a possibility to save current image.
2. *Threshold*. The result of the Threshold operation highly depends on the ambient light thus each new test required threshold value calibration before starting it. This tab is specifically created to control the applying threshold value. It provides the possibility to configure this value and gives an immediate respond by applying Threshold operation on the last acquired image and showing the result under the grayscale original image. The screenshot of the Threshold tab is shown on the Figure 3.8.
3. *Assembly*. The primary tab of the application (Figure 3.9). It contains four image boxes on the left, control tools on the right and motion stage real time status information on the bottom. Left-top image box contains last acquired image. Left-bottom image box – Thresholded (binary) image of it. Right-bottom image box shows the template image. Finally, right-top image box represents the graph of the last pattern recognition metric distribution of the template matching metric along theta value.

On the right side of the tab there are a number of control tools (beginning from the top one):

- Two buttons to move the pickup tool to the absolute or relative position which is written in the forms beside buttons respectively.
- A set of radio-buttons choosing type of markers on samples and the pattern recognition mode.

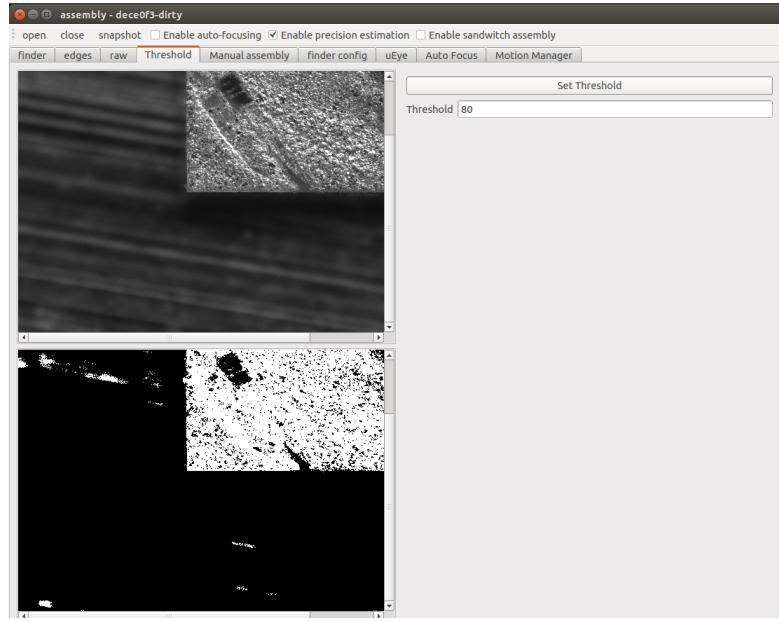


Рис. 3.8: Screenshot of the Threshold tab of the PSAuto application.

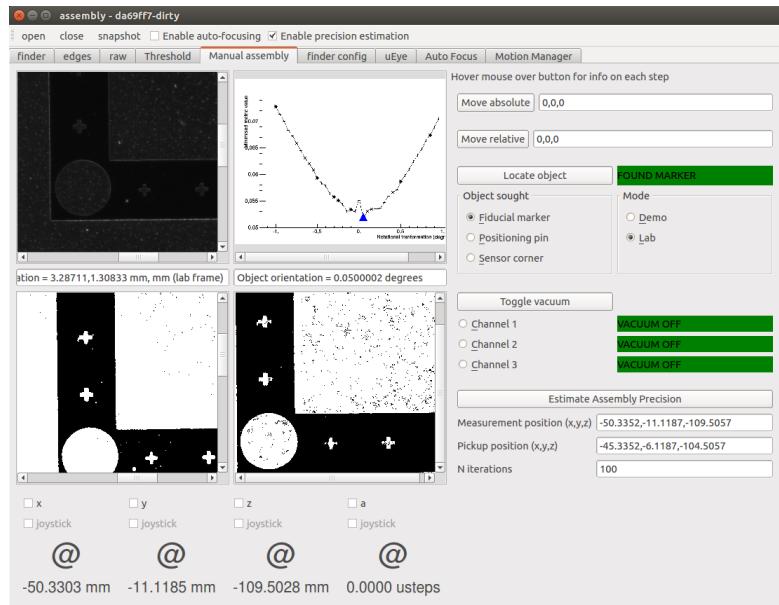


Рис. 3.9: Screenshot of the Assembly tab of the PSAuto application.

- A set of buttons to control the relay card of vacuum lines. Beside each button there is a real time indicator of the current vacuum line status.
- The last tool in this tab provides the possibility to conduct tests of the system. It consists of several forms for input test information and a button to start the test.

4. *Auto focus*. Another nice tool of the application is autofocus. It place an essential role in the pattern recognition as soon as its accuracy directly depends on the quality of acquired image. There are two features provided by this tab: find the focus position of the camera and move to this position. To find the focus position

of the camera it is required to set the step size along Z-axis and the number of steps. As smaller the step is, the more precise focus position will be identified. The idea of auto focus detection is similar to best theta detection in the pattern recognition. For each step position it saves the metric data representing the blur of the acquired image thus in the end the software can find the position with least blur on the image comparing saved metrics for each image.

This algorithm can also be used for another purpose. As soon as the camera is constantly fixed on the robotic arm, it is possible to make relative Z-axis distance measurement. For instance, that can be used for measuring the glue layer thickness of an assembled modules.

5. *Motion Manager.* This tab provides the control of the motion and rotation stages. Among its functionality are:

- Independent control of X,Y,Z axis and rotation stage.
- Self-calibration.
- Real time monitoring of the status of stepper motors of each axis and rotation stage.

Глава 4

Precision estimation tests and the first prototype

The automated assembly system has a number of properties in terms of precision:

1. Motion stage movement repeatability.
2. Image acquiring repeatability.
3. Precision of pattern recognition.
4. Possible movements of a sensor while picking them up and down with the vacuum pick up tool.

In order to investigate this properties a series of tests were done.

Real sensors will be very thin (around 200 um). This fact makes them very fragile. Even though dummy sensors, which will be used for further experiments, is a bit thicker (around 300 um), they are still too fragile for the first tests, because the bottom surface of the pick up tool and the plane underneath testing samples are not yet parallel enough. That is why for the first pick up and down tests we used glass samples. They have the same dimensions and represent close enough the properties of silicon sensors. Moreover, they are much cheaper, so that in case of test failure (sample break) it will not be such a big problem as if silicon sample crashes. Despite all mentioned above, none of glass samples where crashed.

Even though we did not do the pick up test with silicon samples, there is still an opportunity to get some information of the pick up and down precision of the silicon samples without direct tests with them. By making a full range of tests with glass samples, we will be able to say how pick up and down influences the precision. Based on this results we will be able to approximately predict the precision of pick up and down tests with silicon samples. Later, when parallelness of the bottom surface of the pick up tool and samples will be provided, we will be able to confirm the results of the prediction.

4.1 Pattern recognition precision tests

For investigation of the pattern recognition precision the following tests were done. During these tests samples were not moved, so that the additional errors by vacuum pick up and down can be excluded.

4.1.1 Pattern recognition on the painted corner of a glass dummy

In the very first test we investigated the pattern recognition on the corner of the sample. Thin pieces of glass with a silver painted corner (Figure 4.1) were used for the tests as an approximation of a silicon sensor. Silver painted corner was used as a marker for pattern recognition to be found in the acquired image.

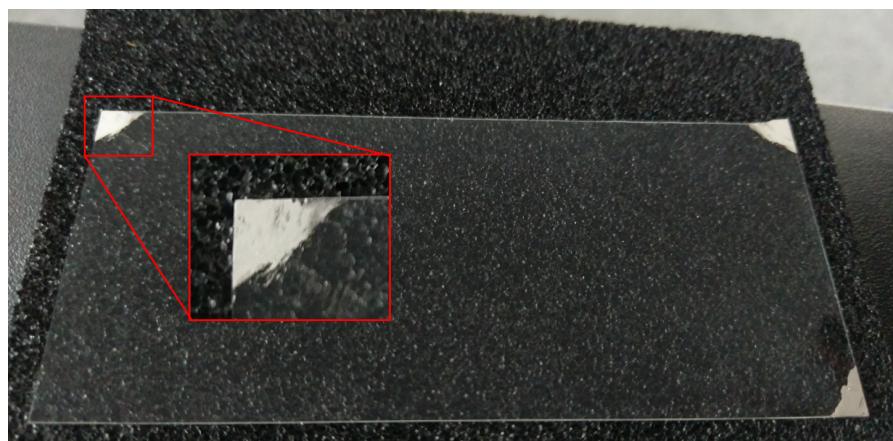


Рис. 4.1: Glass sample with silver painted corner.

The step-by-step outline of this test is listed below:

1. Move to the image acquiring position.
2. Acquire image and run pattern recognition.
3. Move aside for 5 mm in all three axes.
4. Move to the image acquiring position.
5. Acquire image and run pattern recognition.
6. Save data of the current iteration and go to the next one.

After each step software saves the difference between measured coordinates before and after moving the arm with the camera. The distributions of these values are showed in Figure 4.2, Figure 4.3 and Figure 4.4 for X-axis, Y-axis and theta, respectively. The test had 100 iterations done in a row.

Looking at the Figures 4.2 and 4.3 one can see that the X, Y detection of the pattern recognition has precision of around 1-2 um, while the theta detection results do not look so precise. There are several reasons of such behaviour. The main one them is shown on the Figure 4.5.

Silver painted surface is not flat in 10 um scale. Due to this roughness different amount of light reflects to the camera from different points along the painted surface.

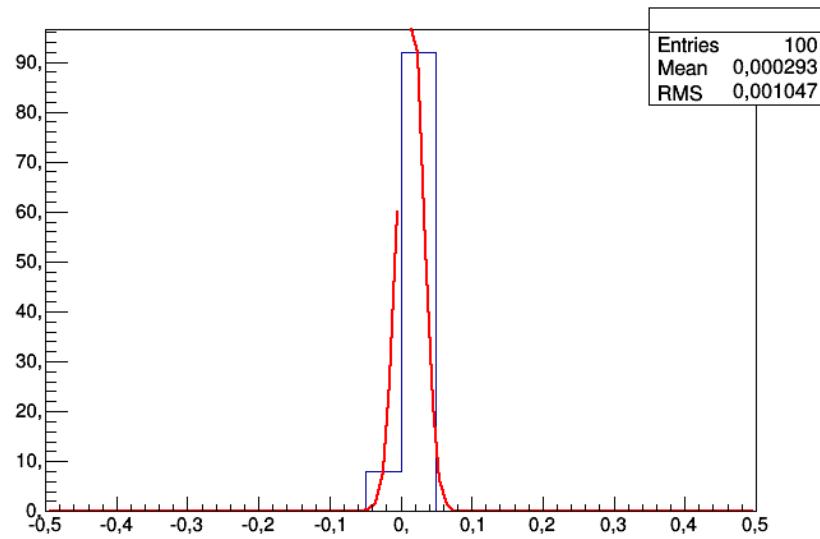


Рис. 4.2: Distribution of the difference between detected X coordinate of the master image before and after moving the arm in each iteration. $\Delta X \approx 1 \text{ } \mu\text{m}$.

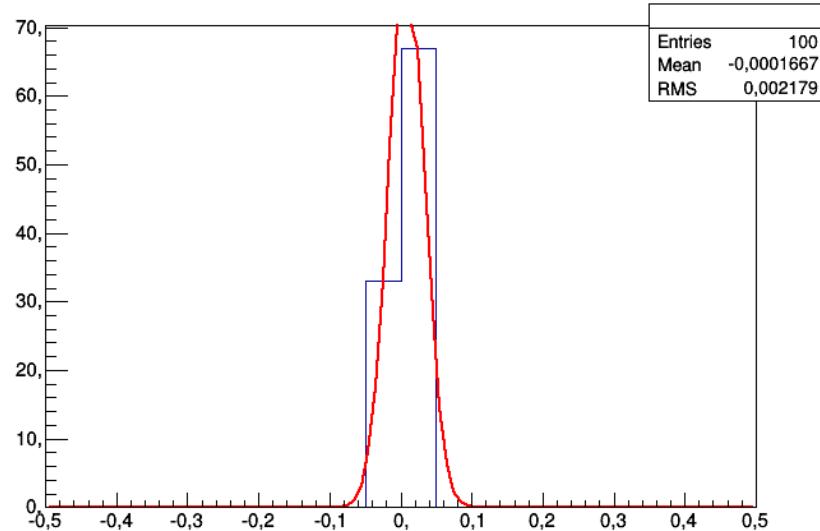


Рис. 4.3: Distribution of the difference between detected Y coordinate of the master image before and after moving the arm in each iteration. $\Delta Y \approx 2 \text{ } \mu\text{m}$.

That is why the painted corner contains various shades of grey, which in some points are darker, than the table underneath the sample (background). All these result into the picture one can see in the Figure 4.5. This kind of pictures has random distribution of dark areas on it. That is why the pattern recognition algorithm has such error while comparing two pictures (master template and acquired image) with random distribution of black areas. Moreover, this kind of tests lasts around one hour, which is long enough for the sun to change the ambient light in the laboratory. Even though all reasonably possible measures were done to prevent such effect, the acquired image is very sensitive

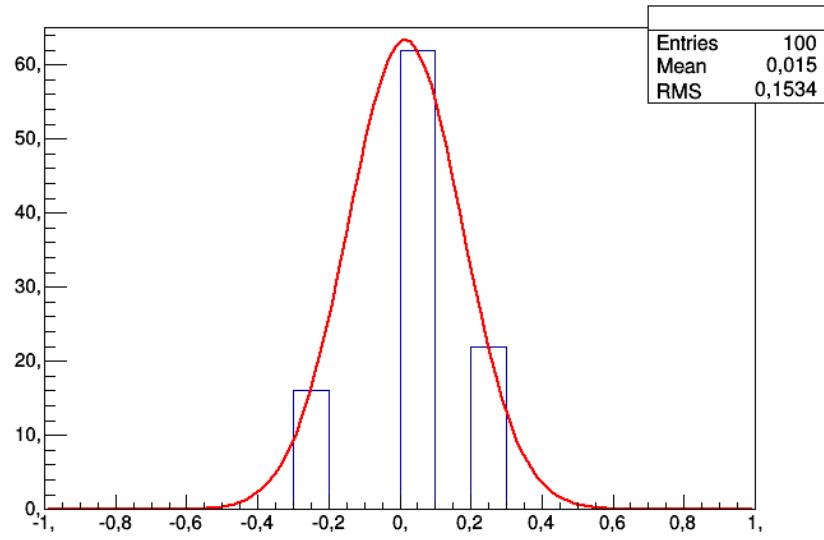


Рис. 4.4: Distribution of the difference between detected angle orientation of the master image relatively to the acquired image before and after moving the arm in each iteration. $\Delta\theta \approx 0.15 \text{ degree}$.

for light. For example, the effect of the sun light can results in the threshold variation for 20 units (the color depth is 256) even with covered window in the laboratory.



Рис. 4.5: The view of the corner after applying the Threshold.

4.1.2 Pattern recognition on the marker of the dummy sensor

The same test, but with dummy silicon sensor and real marker on it, was done. Before the test marker was aligned as much close to zero degrees as possible. At the Figure 4.6 one can see that the edge of the marker after applying Threshold is almost

perfect (+/- one pixel). This fact itself is already a proof that Threshold step of pattern recognition is feasible.

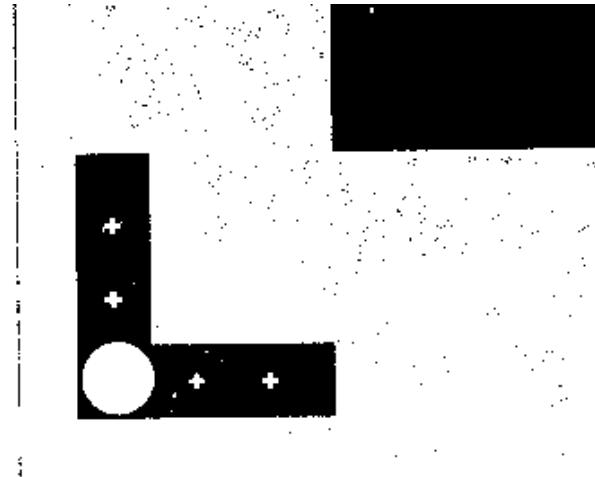


Рис. 4.6: Sensor marker after applying Threshold.

The Distribution of X, Y coordinates and theta shows better results than with painted corner, which was expected. For X and Y it is less than a micron, which is already at the limit of camera resolution. $\Delta\theta$ is one order of magnitude better than with painted corner — $\approx 0.02\text{degree}$.

A screenshot of the application during the test is shown on the Figure 4.7. On the pattern recognition curve one can see that at 0 degree there is an unexpected short upward shot. This peak is not a fluctuation and it is not consist of only one point in the plot. As the scale increases, more points form this peak appear. It was not observed in the previous test just because the theta step was one order of magnitude bigger.

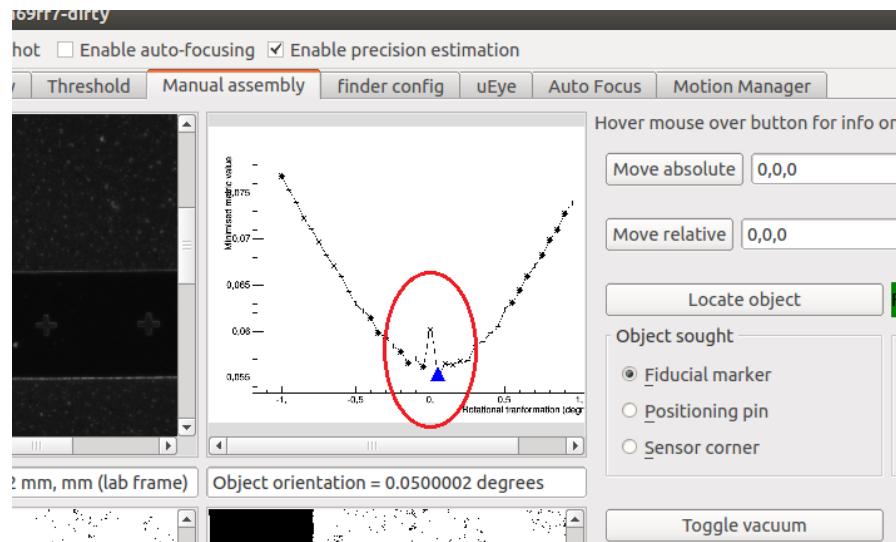


Рис. 4.7: Screenshot of application during precision estimation test with dummy silicon sensor and marker on it.

4.2 Pickup precision tests

Next set of tests were oriented to explore the effect of pickup and setdown on the precision of the system. In other words, will this process move the sample or not in XY plane. Due to the risk of breaking fragile silicon samples, we used glass ones.

4.2.1 Pickup precision tests without assembly platform

The first pick-up/down test was done without assembly platform. The step-by-step outline of this test is listed below:

1. Move to the image acquiring position.
2. Acquire image and run pattern recognition.
3. Move to pre-pickup position.
4. Move to pickup position.
5. Turn negative vacuum on.
6. Move up.
7. Move down.
8. Release vacuum.
9. Move down.
10. Move to pre-pickup position.
11. Move to the image acquiring position.
12. Acquire image and run pattern recognition.
13. Save data of the current iteration and go to the next one.

One can notice the step of moving first to pre-pickup position before going to the pickup position and visa versa. This step is essential. The motion stage provides equal speed in all three axes, so when it receives a command to move to some position it starts to move with equal speed in each of three directions towards the destination simultaneously. As soon as destination in one axis is reached, it obviously stops moving in this axis while moving in other axes is going on. Therefore there is an unlike situation when the robotic arm reaches the sample in Z-axis while X and Y axes would still keep moving, which may cause damage or even break a sample. To prevent such situation we added the step of moving to pre-pickup position.

The results of the test showed movement of the sample, which could be noticed with a naked eye. In order to minimize this movement we decided to make a touch test – the same as pickup and setdown, but without vacuum. Such kind of test can show the contribution of the touch to the sample movement in the pick-up and -down test. The results if this test was quite similar to the previous test, which means that the touch itself contributes the most to the sample movements in the pick-up and -down test. The interesting fact of this movement is its trend. On Figure 4.8 one can see X and Y movement trends with respect to iteration number in the touch test. This plots show that the movement is not random and it is more or less constant both in value and direction.

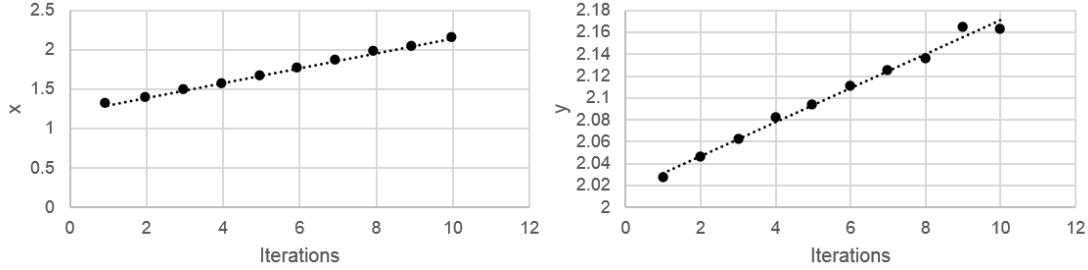


Рис. 4.8: X and Y movement trends with respect to iteration number in the touch test.

The most probable reason for such effect is that pickup tool surface is not parallel to the table, where samples lay waiting for being picked-up. This fact perfectly corresponds to the constant movement of a sample to the one direction. Unfortunately, it is very complicated to align this surface parallel enough to make sample movement negligible. Another way to avoid this movement is to use an assembly platform, which can hold samples with vacuum underneath so they would be fixed.

4.2.2 Pickup precision tests with assembly platform

The step-by-step outline of the pickup and setdown test with the assembly platform is very similar to the one without it. The only difference is that samples are always fixed by the underneath vacuum and are only released, when the vacuum from pick-up tool is provided, so it can lift the sample upwards.

1. Move to the image acquiring position.
2. Acquire image and run pattern recognition.
3. Move to pre-pickup position.
4. Move to pickup position.
5. Turn pick-up tool negative vacuum on.
6. Turn assembly platform negative vacuum off.
7. Move up.
8. Move down.
9. Turn assembly platform negative vacuum on.
10. Turn pickup tool negative vacuum off.
11. Move down.
12. Move to pre-pickup position.
13. Move to the image acquiring position.
14. Acquire image and run pattern recognition.
15. Save data of the current iteration and go to the next one.

Distribution of the difference between detected X coordinate of the master image before and after moving the arm in each iteration is shown on the Figure 4.9.

As one can see from the Figure 4.9, $\Delta X \approx 4 \text{ um}$, which is very close to the limit of 1 um of the pattern recognition itself without touching the sample. Moreover, the X and

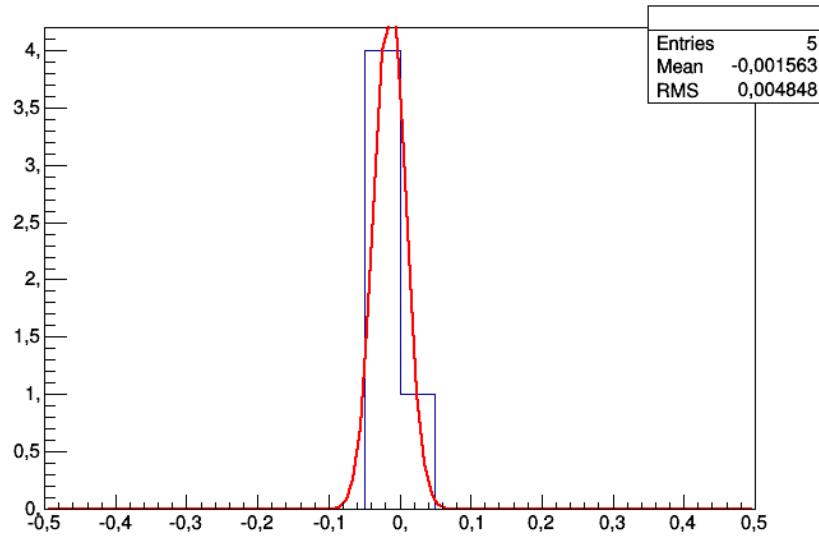


Рис. 4.9: Distribution of the difference between detected X coordinate of the master image before and after moving the arm in each iteration. $\Delta X \approx 4 \text{ } \mu\text{m}$.

Y coordinates show no trend with respect to the iteration number. Taking into account these results, it is possible to say that using an assembly platform can fix samples enough tight, so it is possible to neglect their movements.

4.3 First assembled prototype

After all tests mentioned above it was possible to assemble the very first prototype of the module. To test the precision of the assembling module for the first time it was decided to use simplified assembly algorithm, which would provide maximum precision only on one corner. Schematic view of the first module prototype is shown on the Figure 4.11.

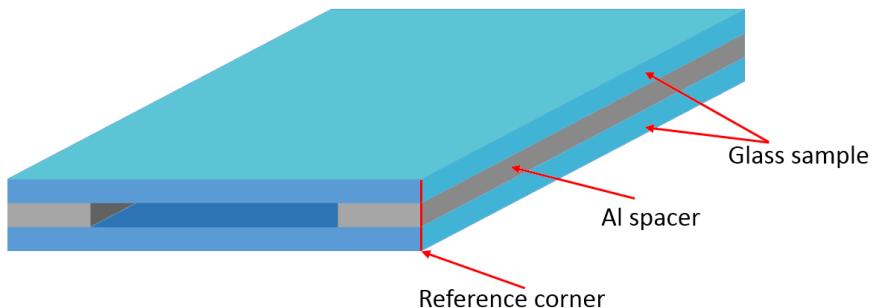


Рис. 4.10: Schematic view of the first module prototype.

The step-by-step outline of the assembly process is listed below:

1. *Place and align top sample parallel to X-axis of motion stage and lift it up with pick-up tool.* Place top sample on the assembly platform and fix it with vacuum. Find with the pattern recognition its orientation in space. Rotate the platform to align top sample parallel to the edge of the pick up tool. As soon as theta recognition do not provide enough accuracy, the following method was used. Move camera for 5 cm along the sample edge and measure the shift of the sample edge. Knowing two cathetus of it is possible to find the angle of the right triangle, which equals to the angle between sample edge and x-axis of motion stage. Rotate the assembly platform for this angle. To sum-up, first – rough top sample orientation estimation with direct pattern recognition, second – precise calculated sample orientation with Pythagorean theorem. Now the top sample is aligned enough parallel to the X-axis of the motion stage. Next, save position of the reference corner with a pattern recognition. Finally, pick-up top sample with pick-up tool and leave it there.
2. *Place Al spacers.* Put Al spacers to the inserts, fix them with bottom vacuum and align them parallel to the X-axis of the motion stage. Absolutely the same way as with top sample. Save the coordination of the reference corner with pattern recognition. Comparing coordinates of the reference corners of Al spacer and top sample, move the robotic arm in XY plane of the motion stage so they would match.
3. *Glue Al spacers to the top sample.* Place the glue on Al spacers and move robotic arm with attached top sample down. It is important to move robotic arm down for exactly correct distance. This topic was discussed in details in the Module Assembly Chapter. Wait glue to be cured and lift glued structure (Al spacers and top sample) from assembly platform.
4. *Place and align bottom sample parallel to X-axis of motion stage.* Place bottom sample on the assembly platform and fix it with bottom vacuum. Make the parallel alignment to the X-axis with the same method as with top sample. Find XY coordinates of the reference corner with the pattern recognition. Compare them to XY coordinates of spacers and top sample. Move the robotic arm with attached structure glued before to match reference corners of all parts of the module.
5. *Final gluing.* Put the glue on the bottom sample and move robotic arm down for correct distance to glue entire prototype. After glue is cured the assembly process is finished.

The glued prototype is shown on Figure 4.11.

We used simplified algorithm for the first prototype assembly aligning only one corner. The quality of assembled prototype is very promising. It meet all necessary requirements. On the Figure 4.12 one can see the photo of the reference corner from the microscope. Pictures form other directions look very similar having approximately the same accuracy.

This assembled prototype proved the feasibility of the automated assembly and showed lots of issues to be solved in future. Even though it has only silver pained corner,

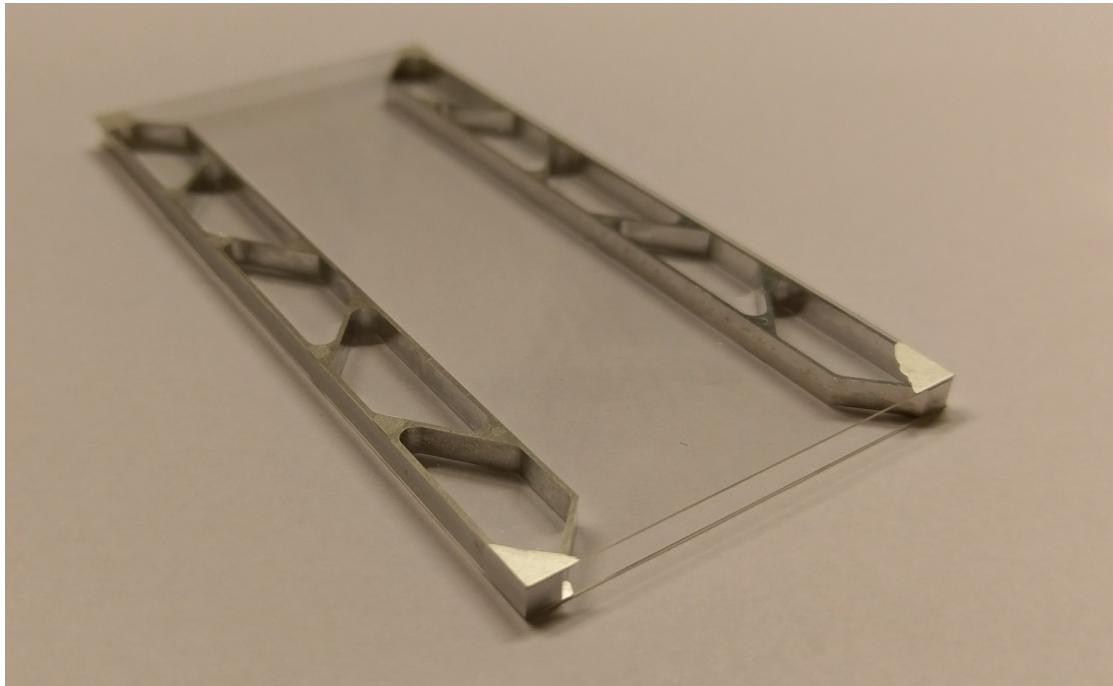


Рис. 4.11: Photo of the first assembled prototype.

not precise lithography, this prototype showed very good results in terms of assembling accuracy.

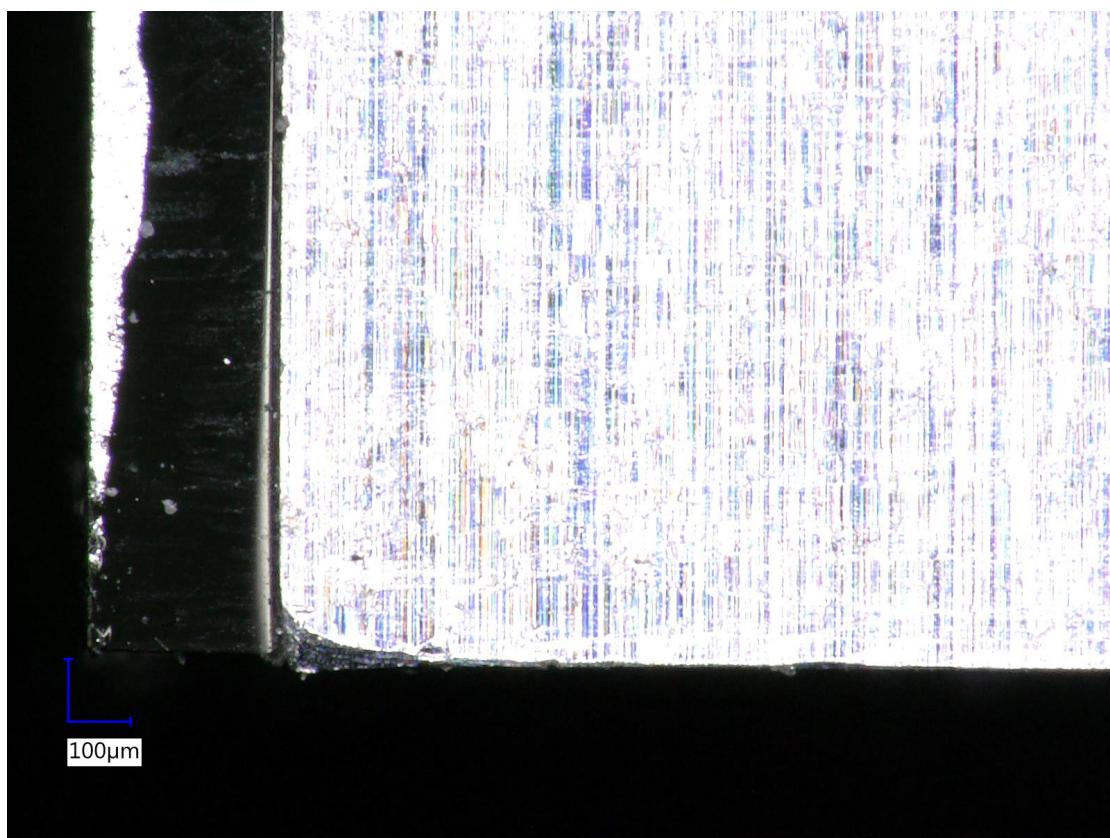


Рис. 4.12: Microscope photo of the reference corner of the first prototype.
Misalignment of the components is around 20 μm .

Глава 5

Conclusion

During my work on the project of Automation of CMS Phase II Tracker module assembly the following results were reached:

- Proved the possibility of an automated module assembly with the required accuracy.
- The first prototype is assembled demonstrating the potential of the system.
- The PSAuto software is modified.
- The assembly platform was designed, manufactured, implemented to the system and successfully tested.
- The fast adhesive technique is investigated and successfully implemented.
- Lots of tests were done showing abilities of the system and its accuracy.

Приложение А

Appendix

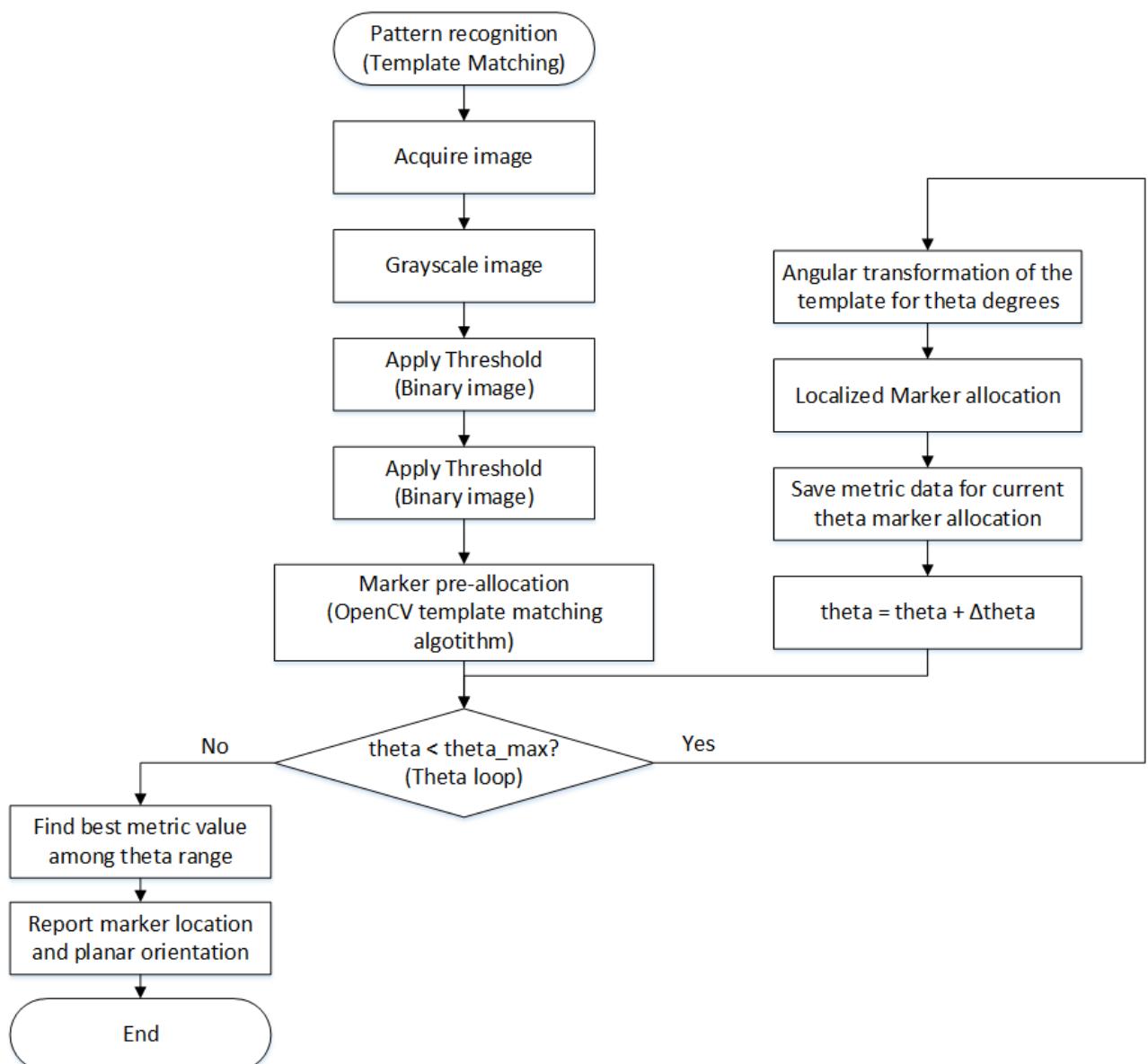


Рис. A.1: Pattern recognition Flow Chart.