

**Система технического зрения для определения положения БЛА относительно  
взлётно-посадочной установки**

## **1. Введение**

Финальным этапом полёта практически каждого летательного аппарата (ЛА) является посадка. На этот этап приходится наибольшее число аварий, поскольку происходит непосредственный контакт ЛА и посадочной поверхности. Проблема особенно усугубляется в случае посадки на ограниченную по площади и перемещающуюся платформу, а также во время неблагоприятных метеорологических условий (тумана, ветра и осадков).

Посадочная платформа может располагаться, например, на водном или грунтовом транспортном средстве во время движения. В этом случае посадка ЛА дополнительно усложняется качкой или воздействием неровности дорожного профиля соответственно.

Перечисленные негативные факторы: случайный характер движения ЛА-платформа и плохая видимость не позволяют пилоту ЛА или оператору беспилотного летательного аппарата (БЛА) в достаточной мере оперативно производить управление для достижения точной и мягкой посадки. Поэтому необходимо автоматизировать процесс посадки для повышения вероятности её успеха, путём создания системы автоматического управления (САУ), представляющей собой программно-аппаратный комплекс (ПАК).

В данной работе рассмотрен принцип построения САУ основанной на перспективном направлении в робототехнике – техническом (компьютерном) зрении, а также создана программа, реализующая описанные в работе алгоритмы ориентации и позиционирования ЛА.

## **2. Структура САУ посадки ЛА**

Предполагается создать САУ с замкнутым контуром управления позволяющую выполнить точную и мягкую посадку ЛА на посадочную поверхность взлётно-посадочной установки (ВПУ).

Ориентировочная схема цепи подобной САУ представлена на рисунке 1а ниже:



Рисунок 1 - Общий вид САУ посадки ЛА на ВПУ

Следящее устройство (блок «Датчики» на схеме САУ), в данном случае, представляет собой совокупность множества датчиков, различных по физическому принципу действия, что обеспечивает высокую надёжность и точность измерений.

Работа САУ посадкой происходит в окрестности ВПУ, начиная с момента её визуального обнаружения. Подлёт в эту окрестность обеспечивается пилотом ЛА или оператором БЛА и системами глобальной навигации, например, GPS/ГЛОНАСС.

В рамках данной работы будет рассмотрена часть следящего устройства – систему технического зрения (СТЗ), установленную на ЛА. СТЗ сама по себе является отдельной САУ и её схема представлена на рисунке 1б. СТЗ включает в себя следующие компоненты:

1. Цифровую видеокамеру (ЦВ) – служит для получения цифровых изображений;
2. Расчетный модуль (РМ) – необходим для обработки полученных изображений;
3. Следящий привод (СП) – наводит оптическую ось камеры на объект слежения.

В качестве объекта слежения выступает графическая плоская чёрно-белая метка.

Целью настоящей работы является разработка прототипа СТЗ, а точнее программы распознавания метки относительного позиционирования и ориентации. Метку предполагается разместить в плоскости посадочной поверхности платформы, а видеокамера со следящим приводом установлены на ЛА. Посредством распознавания характерных угловых точек на метке производится расчёт относительных координат.

### **3. Краткий обзор существующих матричных маркеров**

На сегодняшний момент времени существуют различные плоские графические метки, распознаваемые при помощи методов технического зрения [1-2].

По назначению эти метки можно разделить на два класса:

1. Матричные штрихкоды, служащие для кодирования достаточно большого количества информации. Определение ориентации и положения метки, в ходе её распознавания, – задача второстепенная и подчинена основной задаче – получение закодированной на неё информации (о товаре или грузе, веб-адрес интернет ресурса, справочные сведения). При этом, метка распознается с небольших расстояний и при малом наклоне, а также содержит достаточно мелкие элементы. К подобному рода штрихкодам можно отнести: QR-код, Aztec code, Data Matrix и др.
2. Маркеры дополненной реальности. Дополненной реальностью называют физическую реальность со спроецированными на неё виртуальными объектами. Применение дополненной реальности уже достаточно широко и увеличивается с каждым годом: визуализации информации, индикация на шлеме пилота, сфера компьютерных развлечений и многое другое. Метки в первую очередь необходимы для достаточно точного определения положения камеры в широком диапазоне ракурсов и отличаются более простой структурой, чем матричные штрихкоды – имеют, как правило, просто квадратную форму и характерный простой рисунок для идентификации на окружающем фоне и последующего расчёта положения. Количество кодируемой информации минимально, в количестве необходимом для ориентации и позиционирования, в случае применения совокупности меток, также для присвоения номера данной метки. К подобному рода маркерам можно отнести: Aruco, ArToolKit, Siemens Corporate Research marker, Hoffman marker и другие.

Технология маркеров дополненной реальности наиболее близка для решения задачи, поставленной в данной работе, и являлась прототипом для разработки собственного решения.

#### 4. Структура метки

Разработанная метка представляет собой плоский чёрно-белый рисунок, нанесённый на поверхность посадочной плоскости. Рисунок состоит из совокупности точно расположенных друг относительно друга квадратов. Форма квадрата хорошо выделяется на изображении – в общем случае представляет собой параллелограмм (проекция квадрата). Изображение метки с указанием назначения фигур при различных режимах работы представлено на рисунке 2 ниже:



Рисунок 2 - Структура разработанной метки

Метка имеет следующие особенности, выделяющие её на окружающем фоне:

1. Форма фигур – параллелограммы;
2. Иерархия вложенности фигур друг в друга;
3. Задано соотношение площадей фигур.

Основные требования к создаваемой метке: её точное, быстрое и надёжное распознавание в условиях хаотичных движений снимающей камеры и неблагоприятных погодных условий. Эти требования обеспечиваются следующими решениями:

1. Исключение какой-либо дополнительной информации, например, номера метки. Это упрощение уменьшает время обработки и позволяет избавиться от мелких элементов, требующих большую разрешающую способность камеры. Стоит отметить, что в случае задачи посадки нескольких ЛА на индивидуальную для каждого посадочную поверхность, от этого решения необходимо отказаться;
2. Область поиска метки на изображении, в первую очередь, ищется в окрестности её положения на прошлом изображении (использование так называемого «региона интереса»);
3. Метка имеет вложенность – метка в метке. То есть реализовано два режима распознавания: дальний и ближний. В первом случае опорные точки (координаты углов квадратов) для расчёта положения ищутся на внешних элементах, а во втором на внутренних. Это необходимо для возможности ориентации как на большом удалении ЛА от ВПУ (выбираются крупные внешние элементы), так и в непосредственной близости (когда внешние элементы уже уходят из поля зрения камеры).

Изначально, при заходе на посадку, используется дальний режим, а по мере приближении к метке происходит переключение на ближний режим работы. Критерий переключения с режима на режим – целевая функция в которую входят отношение площади метки на изображении к самому изображению, а также скорость её изменения. Переключение с режима на режим происходит также в случае не обнаружения метки на нескольких изображениях подряд (в случае чрезмерно резкого сближения или удаления).

## 5. Описание математического аппарата расчёта положения

Предложенный ПАК позволяет определить пространственное относительное положение ВПУ посредством обработки снимков ориентационной метки, полученных только от одной камеры. Использование монокулярной системы [3] – с одной камерой, по сравнению с бинокулярной – с двумя камерами, позволяют уменьшить её стоимость и габариты, упростить калибровку, а также увеличить быстродействие т.к. требуется обработать в два раза меньше изображений.

На рисунке 3 ниже представлена аппаратная часть комплекса и с.к., принадлежащие каждому её элементу:

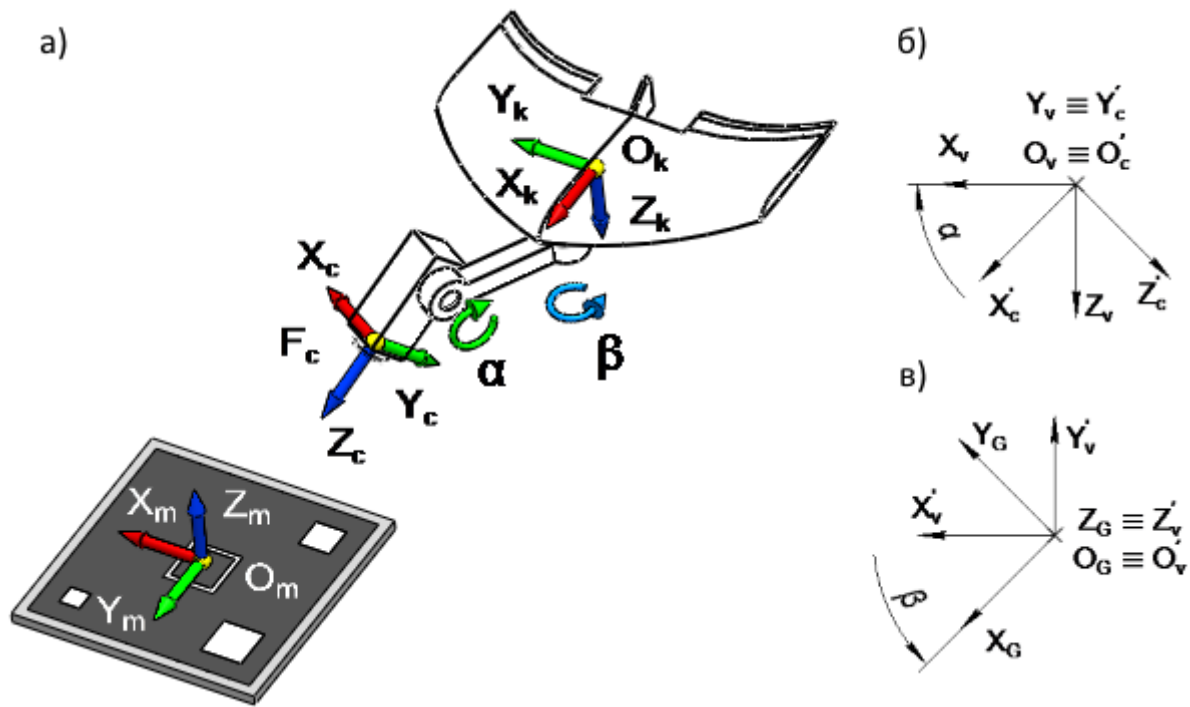


Рисунок 3 - Системы координат элементов комплекса

Введены следующие основные с.к., связанные с элементами ПАК:

1.  $O_m X_m Y_m Z_m$  – с.к. связанная с меткой, начало координат в её центре;
2.  $F_c X_c Y_c Z_c$  – оптическая с.к. камеры, начало координат в фокусной точке;
3.  $O_k X_k Y_k Z_k$  – с.к. связанная с ЛА, начало координат в его центре масс.

Для удержания метки в поле зрения камеры последняя должна иметь определённую степень свободы, относительно ЛА. Вполне достаточно для этого иметь возможность поворота на углы  $\alpha$  и  $\beta$ , обозначенные на рисунке 1а. Поэтому, кроме основных с.к., введены вспомогательные с.к.:

1.  $O'_c X'_c Y'_c Z'_c$  – корпусная с.к. камеры. Начало координат в точке, являющейся базовой на камере и лежащей на оси поворота на угол  $\alpha$ .
2.  $O_v X_v Y_v Z_v$  – с.к. связанная с шарниром поворота на угол  $\alpha$ ;
3.  $O_g X_g Y_g Z_g$  – с.к. связанная с шарниром поворота на угол  $\beta$ ;
4.  $O'_v X'_v Y'_v Z'_v$  – с.к. после смещения  $O_v X_v Y_v Z_v$  в точку  $O_g$ .

Расчёт положения ЛА относительно ВПУ после нахождения опорных точек метки на изображении происходит в три этапа:

1. Численный расчёт пространственного положения опорных точек метки в с.к. камеры;
2. Пересчёт координат точек в с.к. ЛА;
3. Выражение из полученных выражений линейного смещения и набора углов рысканья ( $\psi$  вокруг оси Z), тангажа ( $\vartheta$  вокруг оси Y) и крена ( $\phi$  вокруг оси X). Повороты на эти углы выполняются в порядке их перечисления для совмещения системы координат метки и ЛА.

Полученный набор значений полностью определяет положение ЛА относительно ВПУ.

Для преобразования координат из с.к. метки в с.к. камеры необходимо выполнить последовательный ряд поворотов и переносов.

Эти преобразования описывается следующим матричным уравнением (обобщённый вид):

$$P_c = [R_m^k | t_m^k] \cdot P_m \quad (1)$$

где  $P_c$  – координаты точки метки в с.к. камеры;

$R_m^k$  – матрица поворота;

$t_m^k$  – вектор переноса;

$P_m$  – координаты точки метки в с.к. метки.

Это же самое уравнение в подробном виде:

$$\begin{bmatrix} P_c^x \\ P_c^y \\ P_c^z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \\ 1 \end{bmatrix} + \begin{bmatrix} P_m^x \\ P_m^y \\ P_m^z \\ 1 \end{bmatrix} \quad (2)$$

Имея априорные координаты 4 точек в с.к. метки и определённые их пиксельные координаты в с.к. изображения, матрицу внутренних параметров камеры (геометрия оптики), а также набор коэффициентов коррекции оптических искажений (дисторсий) можно вычислить матрицу аффинного преобразования  $[R_m^k | t_m^k]$ , позволяющую установить положение с.к. камеры относительно с.к. метки.

Элементы матрицы внутренних параметров и коэффициенты коррекции можно получить при помощи калибровки по шаблону. В качестве шаблона выбрана прямоугольная шахматная доска с известными размерами и количеством клеток. Сама калибровка камеры представляет собой получение серии в несколько десятков снимков камерой шаблона с различных ракурсов и последующим запуском программного численного решателя. В качестве исходных данных расчёта для калибровки используются пары координат угловых точек клеток в с.к. шаблона и с.к. изображения.

Последовательной преобразований из одной с.к. в другую представлена ниже:

$$P_c \rightarrow P'_c \rightarrow P_v \rightarrow P'_v \rightarrow P_G \rightarrow P_k \quad (3)$$

Сначала производится пересчёт координат из оптической с.к. камеры в корпусную с.к. камеры. Матрица преобразования находится посредством калибровки. Это необходимо сделать поскольку достаточно сложно определить положение точки фокуса камеры прямыми измерениями. Связь координат точки в этих двух с.к. выражается следующим матричным уравнением, аналогичным (1):

$$P_c = [R'_c | t'_c] \cdot P'_c \quad (4)$$

Координаты точки  $P$  известны в двух системах координат:  $P'_c$  выставляется, а  $P_c$  вычисляется точно так же как и в рабочем режиме (также вычисляется  $[R'_c | t'_c]$ ). Но имеет интерес обратная, а точнее псевдообратная матрица преобразования  $[R'_c | t'_c]$  матрица  $[R'_c | t'_c]^+$ :

$$P'_c = [R'_c | t'_c]^+ \cdot P_c \quad (5)$$

Псевдообратную матрицу можно вычислить, например, посредством сингулярного разложения. Матрица преобразования  $[R'_c | t'_c]^+$  неизменна в процессе дальнейшей работы

системы наведения. Матрица по своей структуре подобна аналогичной матрице в уравнении (4).

Далее выполняется поворот на первый угол наведения  $\alpha$ , показанный на рисунке 3б (вид сбоку), выполняемый в положительном направлении вокруг совпадающих осей  $Y_v \equiv Y'_c$ .

Общий вид:

$$P_v = R_c^v(\alpha) \cdot P'_c \quad (6)$$

Подробный вид:

$$\begin{bmatrix} P_v^x \\ P_v^y \\ P_v^z \\ 1 \end{bmatrix} = \begin{bmatrix} C\alpha & 0 & -S\alpha & 0 \\ 0 & 1 & 0 & 0 \\ S\alpha & 0 & C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} P'_c{}^x \\ P'_c{}^y \\ P'_c{}^z \\ 1 \end{bmatrix} \quad (7)$$

После выполняется пересчёт координат в с.к.  $O_G X_G Y_G Z_G$ , включающее в себя линейное смещение на расстояние между двумя шарнирами, а также поворот на второй угол наведения  $\beta$  вокруг совпадающих осей  $Z_G \equiv Z'_v$ , показанный на рисунке 3в (вид сверху) в отрицательном направлении.

Общий вид:

$$P_G = [R_v^G(\beta)|t_v^G] \cdot P_v \quad (8)$$

Подробный вид:

$$\begin{bmatrix} P_G^x \\ P_G^y \\ P_G^z \\ 1 \end{bmatrix} = \begin{bmatrix} C\beta & S\beta & 0 \\ -S\beta & C\beta & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t_v^{Gx} \\ t_v^{Gy} \\ t_v^{Gz} \\ 1 \end{bmatrix} \cdot \begin{bmatrix} P_m^x \\ P_m^y \\ P_m^z \\ 1 \end{bmatrix} \quad (9)$$

Часть матрицы преобразования отвечающая за поворот  $R_v^G(\beta)$  является переменным в процессе наведения камеры, а столбец  $t_v^G$  остаётся постоянным и определяется конструкцией.

Последнее преобразование координат в с.к. ЛА фиксировано и определяется конструкцией.

Общий вид:

$$P_k = [R_G^k(\alpha_G^k, \beta_G^k, \gamma_G^k)|t_G^k] \cdot P_G \quad (10)$$

Подробный вид:

$$P_k = \begin{bmatrix} C\beta_G^k \cdot C\gamma_G^k & C\beta_G^k \cdot S\gamma_G^k & -S\beta_G^k & t_G^{kx} \\ S\alpha_G^k \cdot S\beta_G^k \cdot C\gamma_G^k - C\alpha_G^k \cdot S\gamma_G^k & S\alpha_G^k \cdot S\beta_G^k \cdot S\gamma_G^k + C\alpha_G^k \cdot C\gamma_G^k & S\alpha_G^k \cdot C\beta_G^k & t_G^{ky} \\ C\alpha_G^k \cdot S\beta_G^k \cdot C\gamma_G^k + S\alpha_G^k \cdot S\gamma_G^k & C\alpha_G^k \cdot S\beta_G^k \cdot S\gamma_G^k - S\alpha_G^k \cdot C\gamma_G^k & C\alpha_G^k \cdot C\beta_G^k & t_G^{kz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot P_G \quad (11)$$

Результирующая матрица преобразования получается перемножениями промежуточных матриц преобразования:

$$[R_c^k|t_c^k] = [R_G^k(\alpha_G^k, \beta_G^k, \gamma_G^k)|t_G^k] \cdot [R_v^G(\beta)|t_v^G] \cdot R_c^v(\alpha) \cdot [R'_c|t'_c]^+ \quad (12)$$

Эта матрица имеет следующий вид:

$$[R_c^k|t_c^k] = \begin{bmatrix} C\theta \cdot C\psi & C\theta \cdot S\psi & -S\theta & t_c^{kx} \\ S\varphi \cdot S\theta \cdot C\psi - C\varphi \cdot S\psi & S\varphi \cdot S\theta \cdot S\psi + C\varphi \cdot C\psi & S\varphi \cdot C\theta & t_c^{ky} \\ C\varphi \cdot S\theta \cdot C\psi + S\varphi \cdot S\psi & C\varphi \cdot S\theta \cdot S\psi - S\varphi \cdot C\psi & C\varphi \cdot C\theta & t_c^{kz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

Столбец  $t_c^k$  определяет позиционирование ЛА относительно метки, а подматрица  $R_c^k$  его угловую ориентацию.

## **6. Алгоритм обработки изображений и поиск опорных точек на них**

Начальным звеном алгоритма является инициализация. В процессе неё производится подключение камеры, чтения из файла её параметров, определённых ранее в процессе калибровки (геометрия оптики и коэффициенты коррекции оптических искажений) и физических размеров метки, а также другие подготовительные операции.

Дальнейшие звенья алгоритма выполняются в цикле вплоть до отключения всей системы в целом. С камеры считывается (захватывается) изображение с меткой, преобразуется из цветного в оттенки серого (имеет меньший размер, а значит ускоряет последующую обработку) и применяется гауссово размытие, уменьшающее случайные шумы. На этом предобработка изображения заканчивается.

Поиск метки основан на контурном анализе изображения – замкнутых линий, ограничивающих однородные области на изображении. Создание иерархического массива контуров происходит в 2 этапа. На первом этапе выделяются границы между областями различных яркостей при помощи градиентного детектора границ Кэнни [4-6]. Далее в двухуровневый массив контуров наполняется с учётом их вложенности друг в друга с помощью метода цепного кода Фримена [4-6].

В зависимости от режима работы ищется либо контур с 4 вложенными непосредственно контурами (дальний), либо 1 (ближний). Эти контуры аппроксимируются алгоритмом Рамера-Дугласа-Пекера [6] – сводятся к контурам имеющих меньшее число точек, но повторяющих очертание исходных. Аппроксимированные контуры проверяются на принадлежность к параллелограммам: должны иметь по 4 точки каждый и не превышать допуск непараллельности противоположных сторон.

Точки найденных контуров сортируются, на основании их ориентации. Затем их положение субпиксельно уточняется [4-6]. Полученные пиксельные координаты точек служат опорными точками для запуска расчёта ориентации [6-7].

Смена режима происходит либо при переходе порогового значения целевой функции, аргументами которой являются величина площади изображения метки и скорость её изменения, либо при не обнаружении метки на нескольких изображениях подряд (возможно при временной потере метки из поля зрения камеры, либо при чрезмерно резком удалении/сближении с ней).

Графическая блок-схема описанного алгоритма представлена ниже:

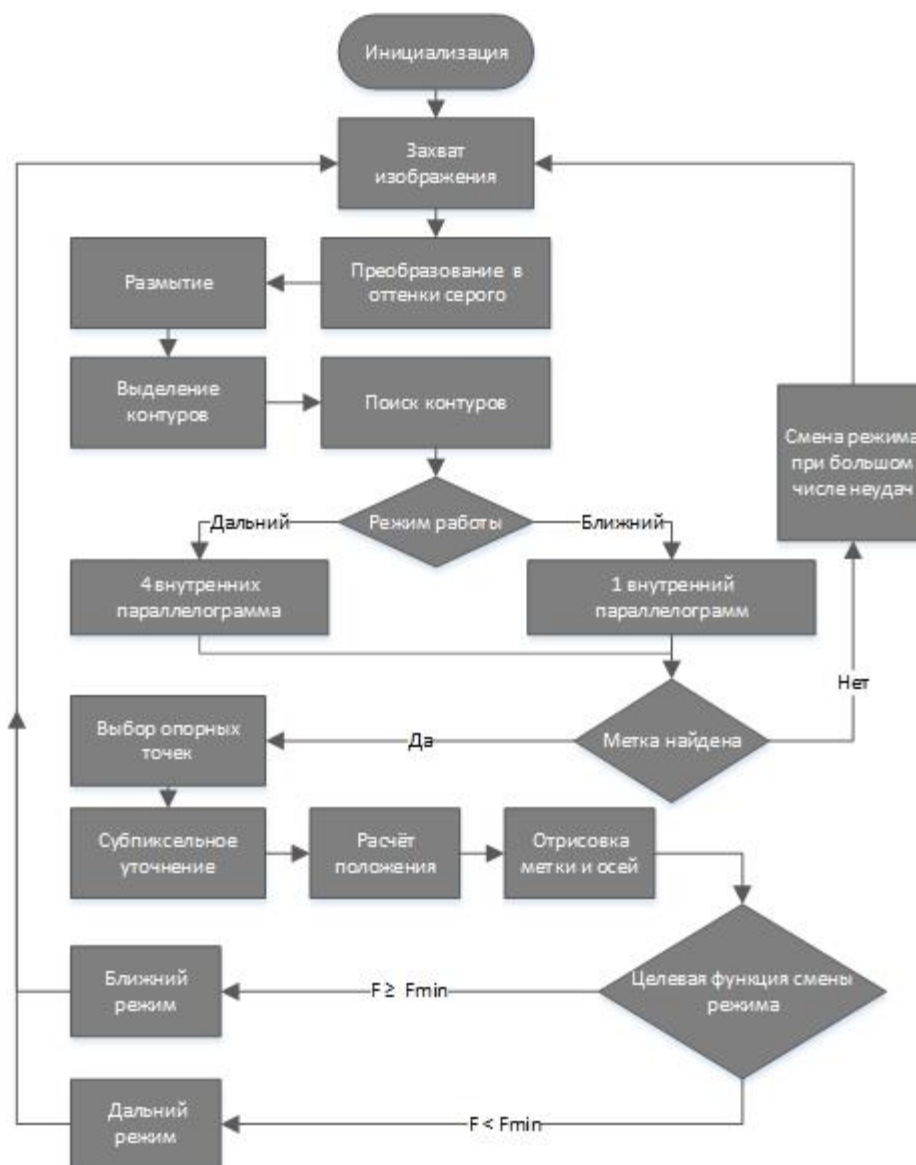


Рисунок 4 - Алгоритм работы программы СТЗ

## 7. Компьютерная программа реализующая разработанные алгоритмы

Разработанный алгоритм был реализован в виде компьютерной программы на языке C++ для семейства операционных систем Microsoft Windows, запуск программы производился на обычном персональном компьютере. Программа создана в интегрированной среде разработки Microsoft Visual Studio 2017 с применением открытой библиотеки компьютерного зрения OpenCV. Для получения изображений была использована web камера бытового назначения, подключаемая к компьютеру при помощи USB кабеля.

На рисунке 5 ниже представлен результат работы программы в виде её окон вывода полученных с камеры изображений, дополненных с.к. связанной с меткой (5а и 5б), а также результаты детектирования контуров методом Кэнни (5в и 5г) в эти же моменты времени. Рисунки 5а и 5в соответствуют дальнему режиму работы, а 5б и 5г ближнему.

Исходные полученные изображения (рисунки 5а и 5б) дополнены окрестностями поиска метки (позиция 1), выделением контуров поиска опорных точек (позиция 2) и рассчитанной с.к. метки.



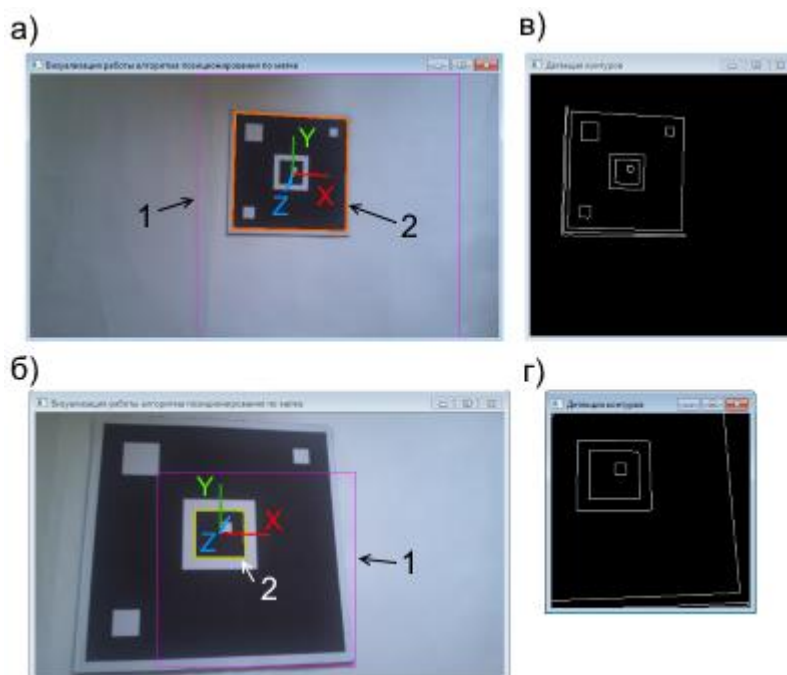


Рисунок 5 - Результат работы разработанной программы

## 8. Выводы

Использование одной только СТЗ в качестве системы позиционирования и ориентации недостаточно поскольку она не обеспечивает должного быстродействия (созданный прототип ПАК обеспечил частоту измерений около 10 Гц, а отдельные акселерометры позволяют добиться частоты измерения в тысячи раз больше), кроме того различные помехи (погодные условия, размытость изображений и вероятность перекрытия метки сторонними объектами) снижают надёжность. Дублирование датчиками другого принципа действия (лидары, акселерометры и др.) позволят решить эту проблему.

Однако, применение технического зрения для решения поставленной задачи показало несколько неповторимых, по сравнению с другими датчиками, преимуществ:

1. Большая степень автономности – возможность работы без наладки радиосвязи ЛА-ВПУ;
2. Подход хорошо зарекомендовал себя для нахождения первоначального положения ВПУ и с последующим уточнением другими датчиками. СТЗ также, после захвата ВПУ, может применяться для периодической коррекции ошибок интегрирования от инерционных датчиков;
3. Применение оптической системы позволяет обнаруживать возможные препятствия на пути к ВПУ и коррекцией траектории избежать столкновения с ними.

Стоит отметить, что СТЗ создана на базе обычного ПК и веб-камеры бытового назначения. Применение профессиональной камеры, мощного вычислительного оборудования, а также дополнение алгоритмами параллельных вычислений существенно повысит ключевые характеристики системы: точность, быстродействие и надёжность.

## 9. Список используемых источников

1. Распознавание маркера дополненной реальности // Geektimes: сообщество гиков. 2012. URL: <https://geektimes.ru/post/135659/> (дата обращения 09.02.2018).
2. Сравнение характеристик штрихкодов // Свободная энциклопедия Википедия. 2016. URL: [https://ru.wikipedia.org/wiki/Сравнение\\_характеристик\\_штрихкодов](https://ru.wikipedia.org/wiki/Сравнение_характеристик_штрихкодов) (дата обращения 12.02.2018).

3. Калибровка Kinect v2 с помощью OpenCV на Python // Хабрахабр: социальное СМИ об ИТ. 2015. URL: <https://habrahabr.ru/post/272629/> (дата обращения 20.02.2018).
4. Фисенко В.Т., Фисенко Т.Ю. Компьютерная обработка и распознавание изображений: учеб. пособие. - СПб: СПбГУ ИТМО, 2008. – 192 с.
5. Шапиро Л. Компьютерное зрение – М.: БИНОМ. Лаборатория знаний, 2006. – 752 с.
6. Adrian Kaehler, Gary Bradski. Learning OpenCV 3. O'Reilly Media, 2016. – 826 с.
7. Camera Calibration and 3D Reconstruction // Open Source Computer Vision. 2018. URL: [https://docs.opencv.org/3.4/d9/d0c/group\\_calib3d.html](https://docs.opencv.org/3.4/d9/d0c/group_calib3d.html) (дата обращения 29.02.2018)