# Список терминов

Порог вхождения (ПВ) — это минимальное значение, при котором датчик считает, что объект находится в его области обнаружения.

Порог понижения (ПП) — это значение, показание которого служит для определения обнаружения части объекта.

Порог нового объекта (ПНО) — это значение, которое показывает, что обнаружен новый объект.

LEDP – это пара светоизлучатель/детектор.

# 1 Анализ состояния проблемы и постановка задачи исследования

Потоковые данные представляют собой непрерывный поток информации, поступающий в систему из внешнего источника в режиме реального времени. Они могут быть в виде различных типов данных, включая аудио и видеосигналы, сенсорные данные, текстовую информацию и т. д.

Значимость адаптивной обработки потоковых данных заключается в том, что она позволяет системам эффективно обрабатывать и анализировать непрерывно поступающую информацию в режиме реального времени. Это особенно важно в ситуациях, когда данные имеют высокую изменчивость и требуют оперативной обработки.

Адаптивная обработка потоковых данных предоставляет возможность системе адаптироваться к изменениям в потоке данных и оптимизировать процессы обработки на основе текущего состояния данных. Это может включать в себя динамическое изменение параметров алгоритмов обработки, выбор наиболее подходящих методов анализа, управление буферизацией данных и другие техники, которые позволяют достичь оптимальной производительности и результатов обработки.

Адаптивный алгоритм обработки включает в себя несколько этапов:

* предварительная обработка;
* адаптация данных в зависимости от характеристики потока данных;
* детективароние объектов выпуклой формы.

Предобработка данных включает в себя фильтрацию данных. Для выбора оптимального фильтра для поставленной задачи необходимо проанализировать существующие методы фильтрации. Обзор включает различные типы фильтров, такие как скользящее среднее, фильтр Гаусса, медианный фильтр, с их особенностями и ограничениями, которые необходимо учесть при выборе соответствующего фильтра.

При выборе определенного фильтра для поставленной задачи, необходимо учесть такие параметры, как частотный диапазон сигнала, требуемое подавление шума или выбросов, а также особенности конкретной системы или оборудования. Также следует учитывать вычислительные возможности доступных алгоритмов и время обработки данных.

Фильтр Гауса также может быть применен в обработке и фильтрации данных. В этом случае он используется для сглаживания шумов и выбросов в данных. Применение фильтра Гауса к данным основывается на том же математическом принципе, который используется для размытия изображений. Он основан на математической функции Гаусса, которая позволяет уменьшить шумы и удалить выбросы в данных. Передаточная функция гауссова фильтра высоких частот (ФВЧ Гаусса) с частотой среза, расположенной на расстоянии D0 от центра частотного прямоугольника, задается формулой

, (1)

Результат применения фильтра Гаусса представлен на рисунке 5.



Рисунок 5 - До и после применения фильтра Гауса

Медианный фильтр — это тип фильтра, применяемый для обработки сигналов и изображений. Для применения медианного фильтра к сигналу или изображению определяется окно определенного размера, которое перемещается по всему входному сигналу или изображению. В каждом положении окна пиксели (или значения входного сигнала) внутри окна сортируются, а затем выбирается медианное значение. Это медианное значение затем заменяет значение пикселя (или входного сигнала) в центре окна.

Фильтр Гаусса можно быть требовательным к ресурсам, особенно если он используется для обработки больших потоков данных. Это связано с тем, что применение фильтра Гаусса требует выполнения множества математических операций. Поэтому в случае работы с большими объемами данных или приложений реального времени может потребоваться значительное количество вычислительных ресурсов и времени для обработки.

Применение медианного фильтра позволяет эффективно уменьшать влияние выбросов и шумов на сигнал или изображение, сохраняя при этом краевые детали и структуру. Особенно медианный фильтр хорошо работает в случаях, когда есть значительные артефакты или шумы, которые необходимо удалить без размытия изображения. Пример представлен на рисунке 6. Результат фильтрации представлен на рисунке 7.



Рисунок 6 – Пример расчета медианы



Рисунок 7 - До и после применения медианного фильтра

Медианный фильтр является требовательным к ресурсам в силу необходимости сортировки значений в окне фильтрации. При применении медианного фильтра к изображению или сигналу, каждый пиксель или значение проходит через окно фильтрации, в котором выбирается медианное значение из всех значений в окне.

Для сортировки значений в окне фильтрации требуется проход по всему окну и выполнение операции сортировки. Это означает, что при каждом применении медианного фильтра к изображению или сигналу требуется большое количество вычислительных ресурсов, особенно при работе с большими окнами фильтрации или изображениями высокого разрешения.

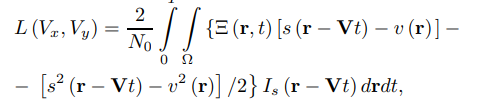
Кроме того, медианный фильтр также требователен к памяти, так как для хранения всех значений в окне фильтрации необходимо выделить достаточно большой объем памяти.

Таким образом, медианный фильтр является требовательным к ресурсам из-за необходимости выполнения операции сортировки значений в окне фильтрации и выделения достаточного объема памяти для хранения этих значений.

После фильтрации необходимо адаптировать данные в зависимости от скорости прохождения объектов, для этого необходимо вывести метод оценки скорости объектов. Пример такого метода в работе [n1].

В данной работе рассмотрены максимально правдоподобные алгоритмы обнаружения и оценки вектора скорости движения изображения неоднородного пространственно протяженного объекта для аппликативной модели взаимодействия полезного изображения и фона. С использованием метода локально-аддитивной аппроксимации логарифма функционала отношения правдоподобия получены асимптотические выражения для характеристик синтезированных алгоритмов.

Для решения задачи проверки гипотезы H1 против альтернативы H0 необходимо формировать логарифм функционала отношения правдоподобия [n2].

 (2)



Если скорость движения объекта априори известна, то решение о наличии или отсутствии изображения объекта в области наблюдения выносится путем сравнения величины L = L(V0x, V0y) с порогом h, определяемым выбранным критерием оптимальности [11–13], в соответствии с правилом (3).

 (3)

Положим теперь, что объект движется из заданного положения с априори неизвестным вектором скорости V0, компоненты которого принимают значения из априорных интервалов Wx = [−Vx max/2; Vx max/2], Wy = [−Vy max/2; Vy max/2]. Обозначим через W двумерную область, в пределах которой V0x ∈ Wx, V0y ∈ Wy. Для исключения влияния неизвестных компонент вектора скорости заменим в (2) их значения на оценки максимального правдоподобия Vbx, Vby [12, 13]. Алгоритм максимально правдоподобного обнаружения изображения движущегося с неизвестным вектором скорости объекта формирует величину (4).

 (4)

Несмотря на сложность и математическую обоснованность предложенного алгоритма оценки вектора скорости движения объекта, он также имеет некоторые недостатки:

1. Сложность вычислений: Для вычисления оценки вектора скорости требуется вычисление функционала отношения правдоподобия для различных значений компонент вектора скорости в заданной области W. Это может потребовать значительных вычислительных ресурсов и времени.

2. Чувствительность к параметрам: Алгоритм может быть чувствителен к выбору параметров, таким как порог h или интервалы Wx и Wy. Неправильный выбор этих параметров может привести к неверной оценке скорости.

3. Зависимость от априорных данных: Алгоритм предполагает использование априорных интервалов для компонент вектора скорости. Однако, если априорные данные недостаточно точны или неточны, это может повлиять на качество оценки скорости.

4. Ограничения на скорость объекта: Алгоритм предполагает, что скорость объекта ограничена интервалами Wx и Wy. Это может быть недостаточно для оценки высоких скоростей объекта.

5. Возможность ложных срабатываний: Использование порогового значения h может привести к ложным срабатываниям в случае наличия шума или других помех в данных.

После фильтрации данных и адаптации данных в зависимости от характеристики потока данных, осуществляется обнаружение объектов выпуклой формы.

Пример алгоритма детектирования выпуклых объектов данных представлен в патенте США №4,276,983. Данный патент описывает устройство и способ для обнаружения контура или формы одного или нескольких изделий с помощью отраженного света.

В настоящем изобретении используется множество пар светоизлучатель/детектор. Каждый детектор света определяет уровень света, отраженного от соответствующего светоизлучателя. Чем ближе объект находится к LEDP, тем больше света отражается от него и регистрируется детектором. Количество отраженного света преобразуется в сигнал и сохраняется. Набор сигналов от всех LEDP дает представление о контуре одного "среза" сканируемой области. Считывание нескольких срезов позволяет получить трехмерное представление о контуре сканируемой области. Изобретение может быть использовано для подсчета количества объектов в сканируемой области, а также для вычисления объема объектов. Когда изобретение используется для подсчета объектов на поверхности, каждый объект должен быть окружен областью, которая отражает меньше света, чем сами объекты. Объекты могут касаться друг друга, если область, где они соприкасаются, отражает меньше света, чем остальные объекты.[1]

Другой задачей изобретения является обеспечение способа подсчета объектов с использованием показаний света, который не требует калибровки или сравнения показаний света от соседних световых детекторов.

На рисунке 3 представлен вид сверху типичного пути, содержащего объекты.



Рисунок 3 - Вид сверху типичного пути

Рисунок 4 представляет собой репрезентативный массив значений сигналов, указывающих на обнаруженный контур траектории.



Рисунок 4 - Репрезентативный массив значений сигнала

Важным аспектом изобретения является то, что LEDP не нужно калибровать, поскольку соседние показания освещенности не нужно сравнивать. Обнаружение объектов осуществляется путем обнаружения изменений в отдельных датчиках и группировки смежных датчиков. Фактически, каждый LEDP может иметь порог вхождения, который отличается от других LEDP.

Следует также отметить, что могут наблюдаться незначительные колебания в показаниях LEDP даже при отсутствии значительных изменений в сканируемой области. Например, в показаниях 22-24 датчик 1 изменяет показания с 91 на 89 и обратно на 91, хотя в этот период под датчиком не проходит никакой объект. Эти незначительные колебания могут быть вызваны рядом факторов, например, вибрацией или загрязнением. Однако такие колебания не влияют на способ изобретения, поскольку они незначительны.

В примере, показанном на рисунке. 3 и 4, для всех датчиков предполагается порог вхождения, равный 99. Это означает, что, когда датчик достигает значения 99, под датчиком находится объект. Объект может находиться или не находиться под датчиком, когда показания света ниже порога вхождения, но, когда показания света находятся на уровне или выше ворога вхождения, под датчиком определенно находится объект. порог понижения равен 9 что означает, что часть объекта прошла мимо датчика, когда показания света, которые были выше порога вхождения, падают на 9.

Порог нового объекта равен 4, что означает, что, если показания света падают на 9, вызывая подсчет, но все еще выше порога вхождения, а затем поднимаются по крайней мере на 4, будет считаться, что обнаружен новый объект.

Группа формируется из включенных смежных датчиков. При обнаружении перепада равного порогу понижения (9) генерируется счет, при этом у всей группы отключается возможность генерации счета.

При падении показателей датчиков ниже порога вхождения, датчики отключаются.

Однако этот алгоритм обладает рядом недостатков, которые необходимо учесть при его применении.

Во-первых, алгоритм не учитывает шум, который может быть обнаружен датчиками. Он предполагает, что различия между датчиками обусловлены только различной калибровкой. Однако в реальных условиях существует возможность появления помех и шумов, которые могут искажать получаемые датчиками данные. Это может приводить к неточным результатам определения контура и подсчета объектов.

Во-вторых, алгоритм не учитывает большое расстояние между датчиками. Получаемые сигналы от датчиков могут иметь слишком большое друг к другу расстояние, что может затруднить точное определение контура объектов. Необходимо разработать специальные методы для учета этого малого расстояния между датчиками и минимизации возможных ошибок.

В-третьих, алгоритм не учитывает разную скорость движения объектов по конвейерной ленте. Это также может привести к неточным результатам при определении контура и подсчете объектов.

На основе проведенного анализа состояния проблемы, разработка комплексного метода себя включает в себя:

* выбор оптимального метода фильтрации и модификации его в соответствии с предметной областью;
* разработка алгоритма адаптации данных в зависимости от характеристики потока данных;
* модификация алгоритма детективарония объектов выпуклой формы;