# 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 Исследование алгоритмов обработки потоковых данных

В данной главе производится общее описание задачи и проводится анализ методов решения этой задачи

# 1.1 Общая характеристика проблемы

# 1.1.1 Определение адаптивной обработки потоковых данных

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.

Примеры приложений адаптивной обработки потоковых данных включают в себя системы распознавания речи, видеонаблюдение, медицинскую диагностику, финансовый анализ и другие области, где требуется быстрая и точная обработка информации в режиме реального времени.

# 1.1.2 Описание проблемы обработки потоковых данных

Проблема адаптивной обработки потоковых данных заключается в необходимости эффективной обработки данных, поступающих в режиме реального времени из различных источников, таких как датчики, сенсоры, устройства IoT и другие. Особенностью потоковых данных является их непрерывный характер и возможность изменения характеристик данных со временем.

В контексте данной задачи, данные с датчиков представляют собой карту вершин, на которой находятся яйца. Основной целью является разработка алгоритма, способного эффективно и точно обнаруживать эти яйца на карте вершин в реальном времени.

Одной из основных проблем является нестабильность и непостоянство данных, поскольку потоковые данные могут быть подвержены шуму, аномалиям и изменениям во времени. Это может привести к ложным срабатываниям или пропускам в обнаружении яиц, что снижает надежность и точность алгоритма.

Другой важной проблемой является необходимость адаптивности алгоритма к изменяющимся условиям работы и требованиям. В контексте данной задачи, например, ситуация может измениться, и новые типы яиц могут быть добавлены на карту вершин. Адаптивный алгоритм должен быть способен автоматически определять скорость потока данных и адаптироваться к ней без необходимости переписывания или переобучения всего алгоритма.

Таким образом, разработка адаптивного алгоритма обработки потоковых данных представляет собой сложную задачу, требующую учета нестабильности данных и способности к адаптации к изменяющимся условиям. Это обеспечит точность, надежность и эффективность обработки потоковых данных и обнаружения яиц на карте вершин.

# 1.2. Обзор существующих подходов к обработке потоковых данных

# 1.2.1 Описание традиционных методов обработки потоков данных их преимущества и недостатки

Традиционные методы обработки потоковых данных включают в себя различные техники и подходы, разработанные для эффективной обработки и анализа непрерывно поступающей информации. Примеры методов обработки потоковых данных:

* буферизация данных;
* потоковая обработка;
* алгоритмы обратной связи;
* методы семплирования и прогнозирования.

Буферизация данных включает в себя использование буферов для хранения и организации потоковых данных. Буферы могут использоваться для временного хранения данных перед их обработкой, а также для сглаживания колебаний в потоке данных. Буферы с разными размерами и стратегиями управления могут быть использованы для обеспечения оптимальной производительности и минимизации задержек в обработке.

Потоковая обработка включает выполнение операций обработки данных непосредственно по мере их поступления в систему. Вместо того, чтобы обрабатывать все данные одновременно, как в пакетной обработке, потоковая обработка позволяет анализировать и обрабатывать данные в реальном времени. Это позволяет системе реагировать на изменения в потоке данных и своевременно принимать решения. Этот подход особенно полезен для систем, где требуется непрерывная обработка данных, например, в финансовых системах, мониторинге сетей или системах реального времени.

Методы семплирования и прогнозирования в обработке потоков данных — это способы анализа и обработки непрерывно поступающей информации для выявления общих тенденций, проведения агрегации и прогнозирования будущих значений.

Семплирование в потоковых данных представляет собой метод, при котором из потока данных выбирается определенное подмножество элементов для анализа. Такой подход позволяет уменьшить объем данных, сохраняя при этом основные характеристики потока. Существуют различные методы семплирования, такие как случайное семплирование, семплирование с перекрытием, равномерное семплирование и другие.

Прогнозирование в потоковых данных — это процесс оценки будущих значений на основе анализа прошлых и текущих данных. Этот метод позволяет предсказать будущее состояние потока данных и использовать эту информацию для принятия решений. Прогнозирование может выполняться с использованием различных статистических и машинно-обучаемых методов, таких как методы временных рядов, регрессионный анализ, нейронные сети и др.

Эти методы позволяют анализировать и использовать только выборочные данные из потоков, что может значительно снизить требования к вычислительным ресурсам. Они также позволяют выполнять прогнозирование и предсказание на основе исторических данных, что может быть полезно для принятия оперативных решений.

Ограничения традиционных методов обработки потоковых данных:

* ограниченные вычислительные ресурсы;
* ограниченная масштабируемость;
* недостаток гибкости.

Ограниченные вычислительные ресурсы. Традиционные методы обработки могут столкнуться с ограничениями вычислительных ресурсов при обработке больших объемов данных или в условиях высокой скорости потока. Это может привести к задержкам в обработке и недостаточной пропускной способности системы;

Ограниченная масштабируемость. Некоторые традиционные методы обработки потоковых данных могут иметь ограничения на масштабируемость. При увеличении объема данных или потоков может потребоваться дополнительная архитектура или оптимизация для обеспечения эффективной обработки;

Недостаток гибкости. Традиционные методы могут быть ограничены в своей способности адаптироваться к изменяющимся требованиям и динамическим ситуациям. В случае изменения формата данных или требований к обработке.

Каждый из этих методов имеет свои преимущества и ограничения, и выбор конкретного метода зависит от требуемых результатов и характеристик потока данных. Однако, в целом, эти традиционные методы обработки потоковых данных являются основными строительными блоками для обработки и анализа непрерывно поступающей информации.

# 1.2.2 Анализ существующих решений по обработке потоковых данных

Пример алгоритма обработки потоковых данных представлен в патенте №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 1 изображен вид в вертикальном разрезе светоизлучающего/детекторного элемента изобретения, расположенного над лентой для переноса яиц.

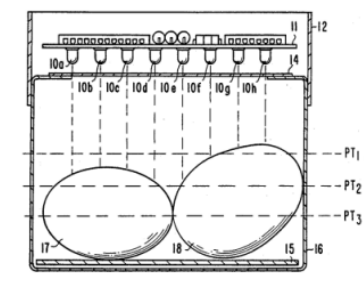


Рисунок 1 - Вид в вертикальном разрезе светоизлучающего/детекторного элемента изобретения

На рисунке 2 представлен график, показывающий показания обнаружения света для одного опроса каждого светового детектора для ленты для переноса яиц. По оси X представлены светодиодные датчики 10a - 10h, а по оси Y - количество обнаруженного света. Рисунок 2 демонстрирует важность ПВ. PT2 является показателем подходящего ПВ, поскольку он представляет собой уровень света, превышающий менее отражающие части периметра яиц, показанных на рисунке 1. Теоретический ПВ на уровне, указанном PT3, был бы слишком низким, поскольку он был бы меньше, чем наименьшее количество света, отраженного областью между объектами, как показано для сканера 5. Таким образом, при таком уровне ПВ невозможно обнаружить "края" объектов. Теоретический ПВ на уровне, обозначенном PT1, будет слишком высоким, поскольку он превышает наибольшее количество отраженного света от сканируемого объекта, а именно яйца 17 на рисунке 1. Поэтому это яйцо не будет подсчитано, если ПВ будет установлен на уровне PT3.

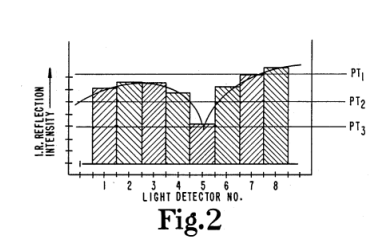


Рисунок 2 - График, показывающий показания обнаружения света

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 расположение, 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.

Разработанный алгоритм направлен на исправление вышеуказанных недостатков, с целью обеспечить более точное определение контура и счет объектов.

Первым значимым улучшением нового алгоритма является его способность учитывать шум, возникающий на датчиках. Путем применения оптимальных алгоритмов фильтрации, таких как метод скользящего среднего, шум на датчиках эффективно подавляется, что способствует повышению точности определения контура и счета объектов.

Вторым улучшением нового алгоритма является его способность учитывать большое расстояние между датчиками. С использованием методов обработки данных и интерполяции, определяется точное положение объекта между датчиками, что позволяет более точно определить контур объекта и выполнить счет.

Третьим значимым улучшением нового алгоритма является его способность адаптироваться к различным скоростям прохождения объектов под датчиками. Он использует динамическую настройку параметров для учета временных задержек и изменений скорости объектов в процессе движения. Это позволяет более точно определить местоположение и скорость объектов, а также предсказать их будущие положения на основе анализа исторических данных и текущих тенденций. Данная функциональность значительно улучшает общую надежность и точность алгоритма при работе с датчиками.

# 1.2.3 Анализ методов фильтрации

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.

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. Недостатком медианного фильтра является его вычислительная сложность, особенно при обработке больших окон или изображений. Пример представлен на рисунке 6. Результат фильтрации представлен на рисунке 7.



Рисунок 6 – Пример расчета медианы



Рисунок 7 - До и после применения медианного фильтра

Скользящее среднее - это метод фильтрации сигналов, который используется для сглаживания данных путем вычисления среднего значения по подмножеству точек данных, перемещающемуся по всему набору данных. Размер этого подмножества, называемого окном, определяется заранее.

Для применения скользящего среднего определяется размер окна, например, 5 точек данных для сигнала. Окно перемещается по всему сигналу, и для каждого положения окна вычисляется среднее значение точек данных внутри окна. Это среднее значение затем заменяет значение в центре окна.

Скользящее среднее полезно для сглаживания шумов и колебаний в сигнале, позволяя выявить общие тенденции и уменьшить влияние случайных изменений. Однако следует помнить, что скользящее среднее также может сглаживать резкие изменения и детали сигнала, что может быть нежелательным в некоторых ситуациях.

На рисунке 8 представлена матица весов. Результат фильтрации представлен на рисунке 9.



Рисунок 8 – Матрица весов



Рисунок 9 – До и после фильтрации

Необходимо учитывать, что выбор конкретного фильтра должен основываться на анализе сигнала и его особенностях. Для каждой ситуации требуется провести анализ спектра частот шумов и артефактов, а также принять во внимание желаемые параметры сигнала, такие как разрешение, стабильность и сохранение деталей.

Таким образом, для эффективной фильтрации данных с датчиков необходимо выбирать соответствующий фильтр, учитывая специфику сигнала и требования к итоговым данным. Подходящий выбор фильтра обеспечит эффективную фильтрацию шумов и артефактов, а также сохранение важных деталей сигнала, что в конечном итоге повысит качество и надежность результатов обработки сигналов.

При наличии шума или непостоянной скорости движения объектов, стандартные методы фильтрации могут давать неточные результаты.

# 1.3 Выводы по главе

- Сделать ссылку на Гонсалеса в обработке методов фильтрации