

АЛГОРИТМ КОМПЛЕКСИРОВАНИЯ СЕНСОРНЫХ ДАННЫХ ДЛЯ ЗАДАЧ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ ПОДВИЖНЫМ СОСТАВОМ

ИВАНОВ Вадим Федорович, начальник отдела математического обеспечения систем управления²; e-mail: v.ivanov@vniias.ru

ОХОТНИКОВ Андрей Леонидович, заместитель начальника Департамента информационных технологий — начальник отдела стратегического развития¹; e-mail: a.ohotnikov@vniias.ru

ГРАДУСОВ Александр Николаевич, ведущий программист отдела математического обеспечения систем управления²; e-mail: a.gradusov@vniias.ru

¹ АО «Научно-исследовательский и проектно-конструкторский институт информатизации, автоматизации и связи на железнодорожном транспорте» (АО «НИИАС»), Москва

² АО «Научно-исследовательский и проектно-конструкторский институт информатизации, автоматизации и связи на железнодорожном транспорте» (АО «НИИАС»), СПб филиал, Санкт-Петербург

В статье дан анализ исследований и работ по обработке сенсорных данных и навигации для подвижных объектов, в том числе с автоматическим управлением. Указано на необходимость интеграции решений на основе технического зрения и нейронных сетей с учетом состояния окружающей среды. Дано описание алгоритма и методов для комплексирования данных, получаемых от сенсорных датчиков бортовой системы технического зрения подвижного состава. В рамках работы алгоритма осуществляется траекторная обработка данных: подтверждение (удаление) наблюдаемых объектов, построение их траекторий, оценка скоростей и координат, формирование списка глобальных объектов. Для расчета модели движения объекта применяется расширенный фильтр Калмана, Венгерский алгоритм и осуществляется расчет расстояния Махаланобиса. Подробно описаны этапы обнаружения, сопровождения и идентификации объектов-препятствий, а также определения их параметров (класса, координат и скорости). Приведены опытные показатели, используемые в данном исследовании. Подтверждена актуальность и показана практическая применимость описанного подхода для задач автоматического управления подвижным составом.

Ключевые слова: система технического зрения, комплексирование данных, идентификация объектов, трекинг объектов, ковариационная матрица, кластеризация, расширенный фильтр Калмана, Венгерский алгоритм, расстояние Махаланобиса.

DOI: 10.20295/2412-9186-2024-10-04-360-371

▼ Введение

Цифровая железная дорога предполагает развитие автономного транспорта [1]. Работа по внедрению беспилотных технологий на железной дороге ведется многими государствами, включая Россию. В августе 2024 года введен в эксплуатацию электропоезд с третьим уровнем автоматизации на Московском центральном кольце. В планах ОАО «РЖД» ввести в опытную эксплуатацию полностью беспилотный скоростной электропоезд «Ласточка» с четвертым уровнем автоматизации (УА4 по ГОСТ Р 70059-2022) к 2026 году [2].

Для автоматического управления подвижным составом (ПС) важной задачей является необходимость определения внешней

ситуации и обнаружения препятствий по ходу движения, особенно в его габаритах. Также перед разработчиками подобных систем стоит задача повышения точности и достоверности обработки данных от датчиков системы технического зрения (СТЗ) для определения, сопровождения и идентификации объектов-препятствий на пути движения ПС, а также их параметров (скорости и координат).

Методам и алгоритмам обработки сенсорных данных посвящено много работ отечественных и зарубежных ученых. Для управления роботами и решения задачи обхода препятствий и планирования пути предложен подход, основанный на обучении с подкреплением [3–5].

Исследования потенциальных методов обработки сенсорных данных и навигации для мобильных роботов в динамической среде, их пригодности и производительности освещены в работах [6–8]. В них исследованы преимущества гибридных методов обработки данных применительно к различным областям применения, в том числе описана интеграция решений на основе нейронных сетей, учитывая состояние окружающей среды, а также их способности устранять основные ограничения по обходу препятствий и планированию пути.

Предмет автономного движения и навигации подвижных объектов включает в себя локализацию, обнаружение препятствий, их идентификацию и принятие эффективного решения по управлению и нахождению оптимального пути от начальной точки до заранее определенного целевого местоположения [9]. Характер встречающихся препятствий может быть статическим или динамическим, в зависимости от структуры окружающей среды и определяемой датчиками (сенсорами) обстановки. Навигация в такой среде может оказаться сложной задачей из-за зависимости от сенсорных возможностей датчиков и возможности анализа огромного количества данных об окружающей среде в режиме реального времени. Некоторые из проблем навигации подвижных объектов включают в себя необходимость точного восприятия, идентификации и реагирования на геометрию окружающей среды, форму и размер обнаруженного объекта, тип и положение препятствия с использованием подходящей модели [10, 11].

Датчики обнаружения препятствий предназначены для взаимодействия с окружающей средой и получения данных об окружающей среде с помощью сенсорных устройств. Затем они используют алгоритмы, основанные на компьютерном зрении и распознавании объектов, для обнаружения, отслеживания и обхода препятствий в навигационной системе.

Недавние исследования внесли значительный вклад в интеллектуальные решения по обнаружению препятствий и обходу препятствий, основанные на использовании сенсорных датчиков и улучшении алгоритмов. В работе [12] представлены различные конфигурации и возможности датчиков машинного зрения, акту-

альные в различных областях. Подробнее в [13] проанализированы датчики систем технического зрения как сложные элементы для понимания особенностей окружающей среды, в том числе с использованием моделей глубокого обучения с целью обнаружения препятствий и навигации подвижных объектов. Повысить качество обнаружения и улучшить методы обнаружения препятствий и их идентификации стало возможно за счет использования лидаров и нейронных сетей [14].

Данные задачи можно решить путем двухэтапной обработки данных от датчиков (сенсоров) СТЗ, основанных на различных физических принципах (камер видимого и инфракрасного диапазонов, лидаров) [15].

На первом этапе (этап первичной обработки) осуществляется обработка изображений камер и облаков точек лидаров с использованием искусственной нейронной сети и/или алгоритмов кластеризации (для данных лидаров). Результатом этого этапа обработки являются обнаруженные объекты (далее — сенсорные объекты, СО) заранее заданных классов.

На втором этапе (этап вторичной обработки) осуществляется объединение (комплексирование) результатов первого этапа, то есть данных об обнаруженных различными сенсорами объектах с учетом предыстории их наблюдения. Результатом этапа являются объекты (далее — глобальные объекты, ГО), число которых соответствует их реальному количеству в контролируемой области пространства перед ПС независимо от числа сенсоров, их наблюдающих.

В рамках работы алгоритма необходимо осуществлять траекторную обработку данных: подтверждение (удаление) наблюдаемых объектов, построение их траекторий, оценку скоростей и т. д.

Рассмотрим работу алгоритма комплексирования данных для определения и идентификации объектов, относящихся к категории «препятствий» (далее — объектов-препятствий), как наиболее важной категории с точки зрения решения задачи автоматического управления ПС. К данной категории относятся такие объекты, как люди, крупные животные, транспортные средства, ПС и крупногабаритные неклассифицированные объекты на пути следования ПС в пределах его габаритов.

Алгоритм комплексирования выполняет следующие основные функции:

- синхронизация входных данных сенсоров;
- сопровождение глобальных объектов (трекинг);
- классификация глобальных объектов;
- определение состояния глобальных объектов;
- управление глобальными объектами;
- идентификация препятствий.

Рассмотрим каждую функцию алгоритма подробнее.

1. Синхронизация входных данных сенсоров

Для этапа вторичной обработки объекты поступают асинхронно, то есть в различные моменты времени (рис. 1), что обусловлено разной частотой работы сенсоров и разными временными задержками между моментом формирования данных сенсорами и моментом их обработки. Однако с точки зрения системы управления ПС желательным является как стабильное поступление данных, так и регулярная выдача результатов вторичной обработки (на рис. 1 обозначено как выход алгоритма).

Алгоритм запускается строго по событию программируемого таймера с заданным периодом времени, большим, чем максимальное возможное время обработки всех данных об объектах на входе сервиса. Верхнее значение периода повторения цикла обработки ограничивается требованиями по частоте обнов-

ления данных на входе системы управления, являющейся потребителем данных алгоритма комплексирования. Типовое значение периода задается равным 100 мс.

За время периода (T_i) происходит накопление входных данных об сенсорных объектах. В каждом периоде количество пакетов, содержащих некоторое число объектов от сенсоров, может быть различным (см. рис. 1).

Стоит отметить, что возможны ситуации, когда объекты с какого-либо сенсора приходят с опозданием по сравнению с другими сенсорами (например, если время обработки исходных данных на нем больше, чем на других). В этом случае сенсорный объект, пришедший на вход программы, реализующий разработанный алгоритм, будет иметь время, меньшее, чем актуальное время выдачи данных. В этом случае предусмотрен механизм отката на предыдущее состояние, чтобы учесть этот объект в списке входных данных и заново рассчитать результаты работы алгоритма. Максимальное время хранения истории состояний может составлять до 1 с.

2. Сопровождение глобальных объектов (трекинг)

Слежение за объектами, то есть контроль эволюции вектора его кинематических параметров (координат и скорости), основано на базе метода глобального ближайшего соседа (*GNN — Global Nearest-Neighbor*) [16] с использованием Калмановской фильтрации [17].

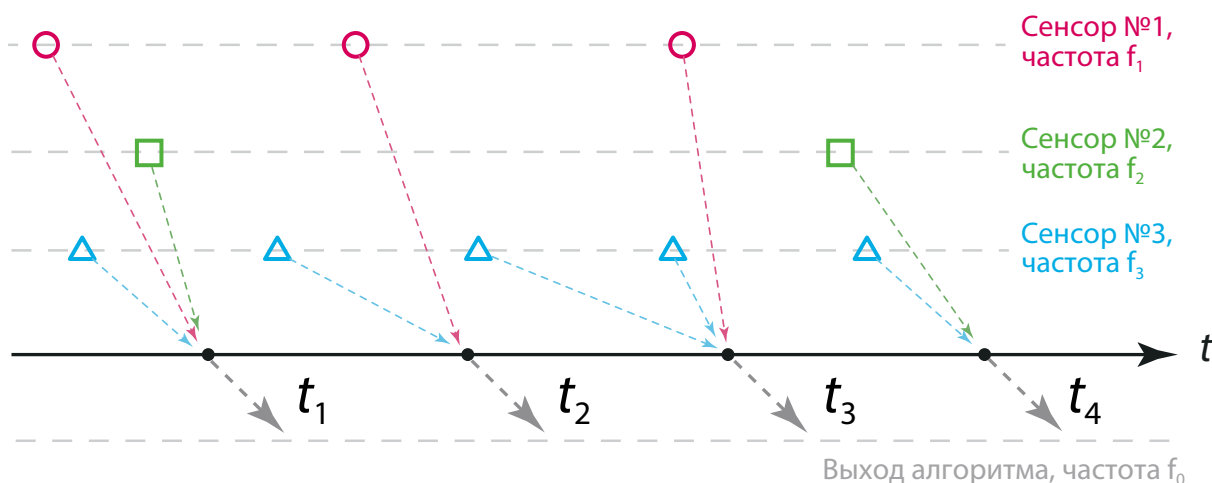


Рис. 1. Синхронизация сенсорных данных от системы технического зрения

В качестве меры для ассоциации между объектами в алгоритме используется расстояние Махаланобиса:

$$C_{ij}^{(dist)} = \sqrt{(x_i - \mu_j)^T S_{ij}^{-1} (x_i - \mu_j)},$$

где x_i и μ_j — векторы декартовых координат i -го глобального объекта и j -го сенсорного объекта, а S_{ij} — матрица ковариации.

Наряду с расстоянием Махаланобиса в пространстве координат вычисляется еще и расстояние в пространстве размеров объектов. Таким образом, объектами, которые могут быть ассоциированы, считаются такие, которые расположены близко друг от друга и имеют схожие размеры. Кроме того, учитываются и классы объектов — ассоциированы могут быть только объекты определенных классов.

Размеры объектов в процессе траекторного сопровождения фильтруются по аналогии с координатами. При инициализации траектории объектам присваиваются размеры того сенсорного объекта, из которого он был инициализирован. При последующих ассоциациях производится правка размеров объекта и ковариационной матрицы.

На каждом шаге производится расчет расстояний Махаланобиса между размерами сенсорных и глобальных объектов. В результате вычисляется расстояние Махаланобиса в пространстве размеров объектов: $C_{ij}^{(dim)}$. Чем больше отличие в размерах объектов, тем больше будет расстояние Махаланобиса между ними.

Таким образом, имеем две матрицы размером $A \times B$, где A — число глобальных объектов, а B — число сенсорных объектов:

C_{dist} — матрица расстояний Махаланобиса координат объектов;

C_{dim} — матрица расстояний Махаланобиса размеров объектов.

Итоговая матрица весов рассчитывается как взвешенная сумма этих матриц:

$$C = C_{dist} + wC_{dim},$$

где w — весовой коэффициент, подбираемый эмпирически.

Числа, записанные в ячейках матрицы, отражают суммарное расстояние между объ-

ектами — как в декартовом пространстве, так и в пространстве размеров. То есть малые числа соответствуют объектам, которые похожи по размерам и расположены близко друг к другу. Большие числа соответствуют объектам, которые отличаются между собой по размерам или пространственному расположению. Бесконечные значения соответствуют заведомо исключенным случаям — несовместимым классам или сильно отличающимся объектам.

Поскольку в общем случае возможны неоднозначные состояния объектов — нескольким глобальным объектам могут быть потенциально ассоциированы несколько сенсорных (рис. 2), то применяется Венгерский алгоритм. Использование Венгерского алгоритма (также известен как алгоритм Куна–Манкреса или алгоритм Манкреса решения задачи о назначениях) [18] позволяет решить задачу выбора эффективного решения, подобрав каждому сенсорному объекту один глобальный таким образом, чтобы суммарное расстояние между ними было минимальным.

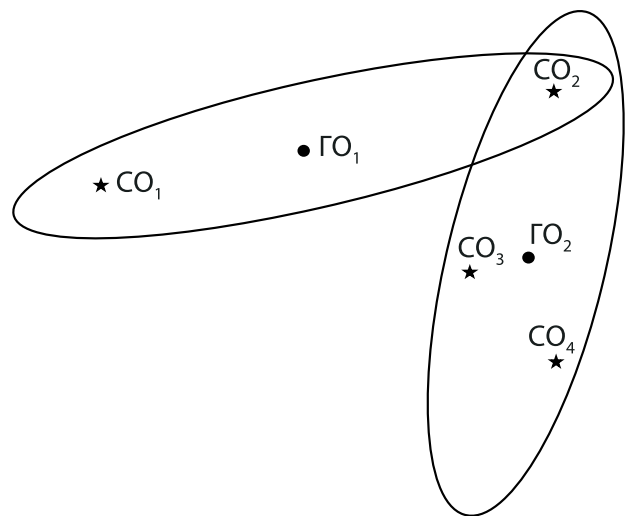


Рис. 2. Неоднозначная ситуация: сенсорный объект CO_2 может быть потенциально ассоциирован двум глобальным объектам (GO_1 и GO_2)

Оценка координат и скоростей, наблюдаемых системой технического зрения объектов, — важная часть в реализации безопасного движения ПС. В частности, зная текущую координату и вектор скорости объекта, можно предсказать его координату в будущем.

Это используется для прогноза столкновений: можно предсказать появление препятствия в «красной» зоне еще до того, как оно там физически окажется («красная» зона соответствует ширине ПС, типовое значение — 3,4 м).

3. Классификация глобальных объектов

Класс объекта определяется на основе анализа истории ассоциаций сенсорных объектов данному глобальному объекту. Такой подход позволяет усреднить показания и принять наиболее точное решение в условиях стохастических значений.

Исходя из физических принципов работы, различные сенсоры по-разному распознают объекты. По кадру с камеры можно распознать человека, но классический алгоритм кластеризации лидарных точек не позволяет это сделать, так как алгоритм оценит только размер объекта, но не сможет распознать класс (среди используемых алгоритмов для обработки лидарных данных есть и нейронная сеть, позволяющая распознать класс объекта, однако для надежности классический алгоритм также работает в параллельном режиме).

Для реализации алгоритма зададим следующие параметры (табл. 1):

Таблица 1. Параметры для классификации глобальных объектов

Параметр	Единицы	Описание
Δt	с	Временной период накопления данных
T_{conf}	%	Порог на подтверждение класса
T_{change}	%	Порог на сброс/смену класса

По мере сопровождения объектов дополняется история (список) классов сенсорных объектов, которые были ассоциированы данному глобальному объекту (то есть для каждого глобального объекта ведется своя история). Длина этого списка при этом варьируется таким образом, чтобы в нем содержались события за последние Δt секунд сопровождения данного объекта. Элементами списка являются индексы классов объектов.

Определяется индекс класса, наиболее часто встречающийся в истории. N_{max} — число событий в истории, соответствующее этому классу.

Тогда доля ассоциаций с выбранным классом:

$$\delta = \frac{N_{max}}{N},$$

где N — число элементов в списке-истории.

Если класс объекта еще не определен, то величина δ сравнивается с порогом T_{conf} . По результатам сравнения класс объекта либо инициализируется, либо остается неопределенным.

Если класс объекта уже был определен, то δ сравнивается с порогом T_{change} (смена одного класса на другой).

4. Определение состояния глобальных объектов

Помимо определения класса, алгоритм также определяет дополнительные характеристики объектов, в частности, показания светофоров и направления движения по стрелочным переводам. Сигнал светофора, а также направление стрелочного перевода определяются аналогично классу объекта (п. 3). Для этого в каждом глобальном объекте хранится история ассоциированных признаков. По результатам анализа истории определяется итоговый признак.

Сигналы светофоров могут быть как постоянными (выключен или включен с постоянным светом), так и мигающими. В алгоритме заложен механизм определения признака мигающего светофора. Проверка на статус мигания производится при каждом обновлении истории объекта в том случае, если сигнал уже подтвержден и для него возможно мигание (желтый или зеленый цвет).

Для определения мигающего светофора необходимо проверить историю сигналов по ряду критериев, в частности:

- минимальное число событий в истории для собственно сигналов и их отсутствия (то есть должно набраться необходимое число детекций как включенного светофора, так и выключенного — необходимое условие мигания);
- минимальное число смен сигналов за анализируемый отрезок истории, при превышении которого считается, что светофор стабильно мигает;

- минимальная необходимая доля событий включенного и выключенного светофора в истории (для борьбы с разовыми выбросами);
- границы периода включенного и выключенного светофора (борьба со слишком короткими или длинными сигналами, то есть невозможная частота).

Сигнал светофора считается мигающим, если в результате анализа истории ассоциированных ему сигналов были пройдены все вышеперечисленные пороговые проверки.

5. Управление глобальными объектами

Поскольку в общем случае объекты от различных сенсоров поступают с различной частотой, применяется следующий подход к управлению циклом жизни глобальных объектов.

Вводятся пороги подтверждения и удаления объектов:

T_c — максимальное время, отведенное для подтверждения ГО;

T_d — временной порог на удаление ГО: максимальное время, за которое не произошло ни одной ассоциации глобального объекта с сенсорными;

$T_{uc}^{(i)}$ — количественный порог для перевода ГО из кандидата в подтвержденные (i — номер сенсора).

Кандидатами в глобальные объекты (КГО) считаются объекты, которые еще не подтверждены.

Для каждого глобального объекта ведется счетчик ассоциаций с сенсорными объектами от каждого сенсора — L_i . Изначально счетчик инициализируется нулем и инкрементируется при каждой ассоциации.

Подтверждение объекта осуществляется в период времени T_c : глобальный объект считается подтвержденным, если за это время счетчик превысил порог хотя бы по одному из сенсоров:

$$\exists i \in [1; K] : L_i > T_{uc}^{(i)},$$

где K — число сенсоров; L_i — счетчик ассоциаций для i -го сенсора.

Если порог был превышен раньше, чем за время T_c , то объект переходит в список под-

твержденных в этот момент времени. Если же за отведенное время T_c счетчик ни по одному из сенсоров не превысил порог:

$$\forall i \in [1; K] : L_i < T_{uc}^{(i)},$$

то объект удаляется как неподтвержденный.

После перевода глобального объекта в статус подтвержденного производится контроль удаления.

Если у глобального объекта не было ассоциаций с сенсорными в течение последних T_d секунд ни по одному из сенсоров, он удаляется:

$$\forall i \in [1; K] : t_c - t_{uc}^{(i)} > T_d,$$

где t_c — время текущего шага обновления глобального объекта;

$t_{uc}^{(i)}$ — время последнего обновления i -го сенсора для данного глобального объекта.

Выбор величин T_c , T_d и $T_{uc}^{(i)}$ осуществляется таким образом, чтобы частота обнаружения (истинные положительные и ложные положительные значения) на пороговых значениях гарантировала определенный желаемый уровень [19].

Например, при частоте работы условного сенсора 10 Гц порог на подтверждение $T_{uc}^{(i)}$ может быть выбран равным 16, а временной порог на подтверждение $T_c = 2$ с. Это будет означать, что если из максимально возможных 20 ассоциаций случилось только 16, то этого будет достаточно для перевода кандидата в статус подтвержденных глобальных объектов (подтвержденный глобальный объект — это объект, который считается оценкой реальной цели, в то время как кандидат может быть ложной целью). Если же за отведенное время не набралось необходимое число ассоциаций, то данный кандидат в глобальные объекты удаляется.

6. Идентификация объектов-препятствий

Входные данные о координатах объектов сравниваются с данными о координатах рельсового пути, и в случае попадания глобального объекта в габариты ПС, находящегося на этих рельсовых путях, объект идентифицируется как препятствие. Для получения стабильных значений используется история расстояний

до колеи от сенсорных объектов, которые участвовали в формировании данного глобального объекта. Каждому созданному глобальному объекту присваивается признак, является ли данный объект препятствием на пути следования ПС или нет.

Признаки препятствий определяются на уровне каждого сенсора в отдельности, и принимается итоговое решение на основе анализа истории признаков с разных сенсоров. Такой подход видится наиболее точным, поскольку каждая камера в отдельности может распознать и колее, и объекты, после чего сопоставить одни с другими, что будет достаточно точным определением препятствий. Объекты с лидара сопоставляются с колеей на цифровой модели пути [20].

Для каждого глобального объекта ведется список расстояний до колеи за последние Δt секунд сопровождения. Длина списка меняется с течением времени в зависимости от количества сенсоров, в поле видимости которых находится объект. Для каждого сенсорного объекта сохраняется и его время.

Поскольку более поздние сенсорные объекты имеют большую ценность, чем ранние, вводится весовая функция (вес данного i -го сенсорного объекта):

$$w_i = t_i - t_N + \Delta t,$$

где Δt — временной период накопления данных;
 t_N — время последнего ассоциированного сенсорного объекта;
 t_i — время i -го сенсорного объекта в списке.

На каждом шаге работы алгоритма комплексирования итоговое расстояние от глобального объекта до колеи определяется как средневзвешенное значение истории расстояний:

$$d_{mean} = \frac{\sum_{i=1}^N w_i d_i}{\sum_{i=1}^N w_i},$$

где d_i — расстояние i -го сенсорного объекта до колеи в момент времени t_i .

В результате работы алгоритма комплексирования данных формируется массив (список) объектов-препятствий, содержащий инфор-

мацию о координатах, скорости, расстоянии и классе объекта-препятствия, необходимый для принятия решений при управлении ПС для его безопасного следования по железнодорожному пути.

7. Практические аспекты реализации алгоритма комплексирования

7.1. Фильтрация объектов искусственной нейронной сети

Для фильтрации глобальных и сенсорных объектов при наличии вагона в поле зрения системы технического зрения в алгоритме используется *Car*-фильтр. Он обеспечивает устранение артефакта при кластеризации лидарных точек, который проявляется в виде распада одного кластера на несколько для одного физического объекта (рис. 3). Это воспринимается алгоритмом комплексирования данных как множество независимых объектов. Это приводит к тому, что алгоритм будет принимать обоснованное положительное решение о существовании нескольких объектов по отношению к одному физическому объекту. Соответственно, такую ситуацию допустимо рассматривать как ошибку первого рода, то есть ложную тревогу. Таким образом, можно говорить о том, что функциональное назначение *Car*-фильтра — это уменьшение ложных тревог в принятии решений о наличии объектов в ограниченной по пространству области.

Car-фильтр активируется только при наличии вагона в красной зоне, то есть в габарите ПС. Требованием к фильтру является то, что в результате его работы на выходе алгоритма комплексирования данных в пределах ограниченной области *Car*-фильтра могут существовать объекты только типа *person* и только один глобальный объект типа *car* (*вагон*) — одновременное появление двух объектов типа *car* невозможно.

Последовательность действий следующая:

1. Определяются текущие координаты ГО (передний край вагона), который распознается датчиками СТЗ (лидарами).
2. Относительно этих координат формируется область пространства, соответствующая размерам вагона (с учетом формы колеи, на которой он стоит, например в повороте).

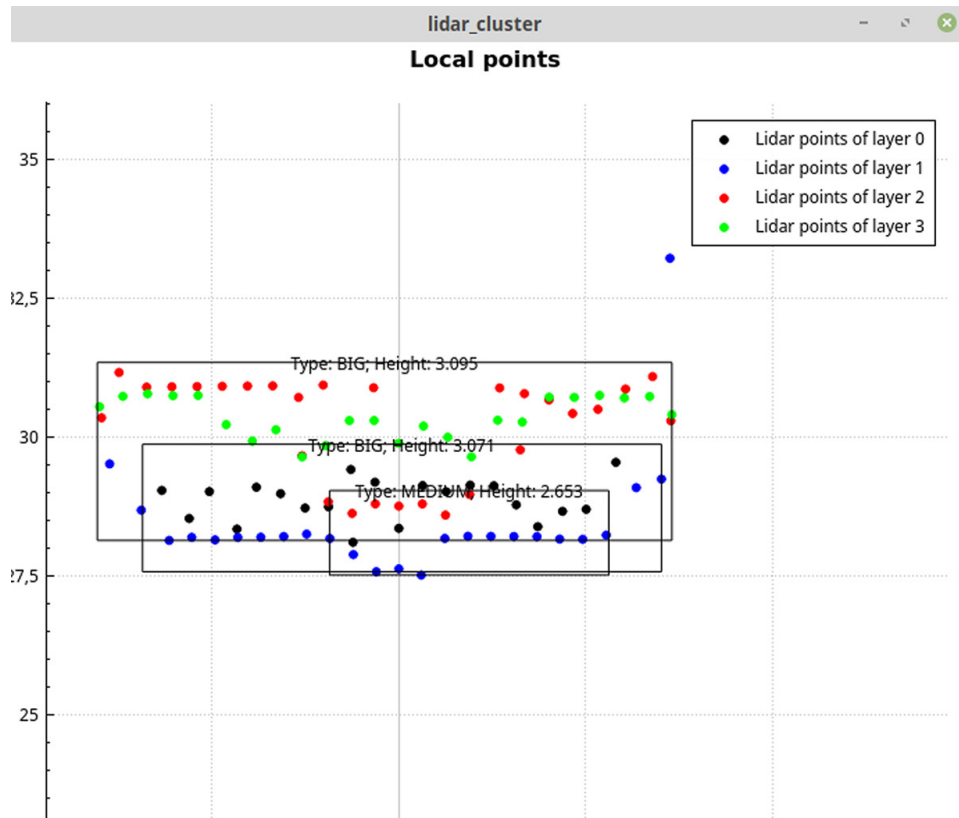


Рис. 3. Кластеризация лидарных точек от вагона типа хоппер (четыре уровня)

3. Из данной области пространства удаляются все объекты неопределенного типа (объекты от лидара) в пределах заданных размеров (например, все объекты шире 1 м).
4. Объекты, размеры которых не превышают заданного ограничения, сопровождаются в фоновом режиме, без выдачи на выход алгоритма. Это необходимо на случай их ассоциации с объектом типа *person* — тогда эти объекты будут участвовать в сопровождении человека.

7.2. Фильтры объектов на платформах и внутри тормозных позиций

В процессе опытной эксплуатации блока обнаружения препятствий, установленного на маневровом локомотиве на станции Лужская, а также на электропоезде «Ласточка» на Московском центральном кольце, были выявлены трудности в обработке следующих ситуаций, приводящие к ложным тревогам:

1. Проезд тормозной позиции. В этом случае лидар распознает ряд объектов малого размера, расположенных в красной

зоне. Однако в данном случае это часть инфраструктуры станции, и они не должны распознаваться как ложная тревога.

2. Подъезд к платформе с людьми на ней (рис. 4). В случае большого числа людей на платформе алгоритм траекторной обработки и комплексирования данных испытывает ряд трудностей: увеличивается процессорное время на обработку массива объектов, а также происходит «перемешивание» ГО, что может приводить к ложным ассоциациям и, как следствие, ложным прогнозам координат объектов в рамках Калмановской фильтрации.

Обе проблемы решаются схожим образом:

- 1) данные объекты инфраструктуры (вагонные замедлители и платформы) заносятся в цифровую модель пути;
- 2) в процессе движения ПС алгоритм комплексирования получает актуальные координаты инфраструктурных объектов по направлению движения;
- 3) в пределах области пространства, занимаемой данными объектами инфраструктуры,

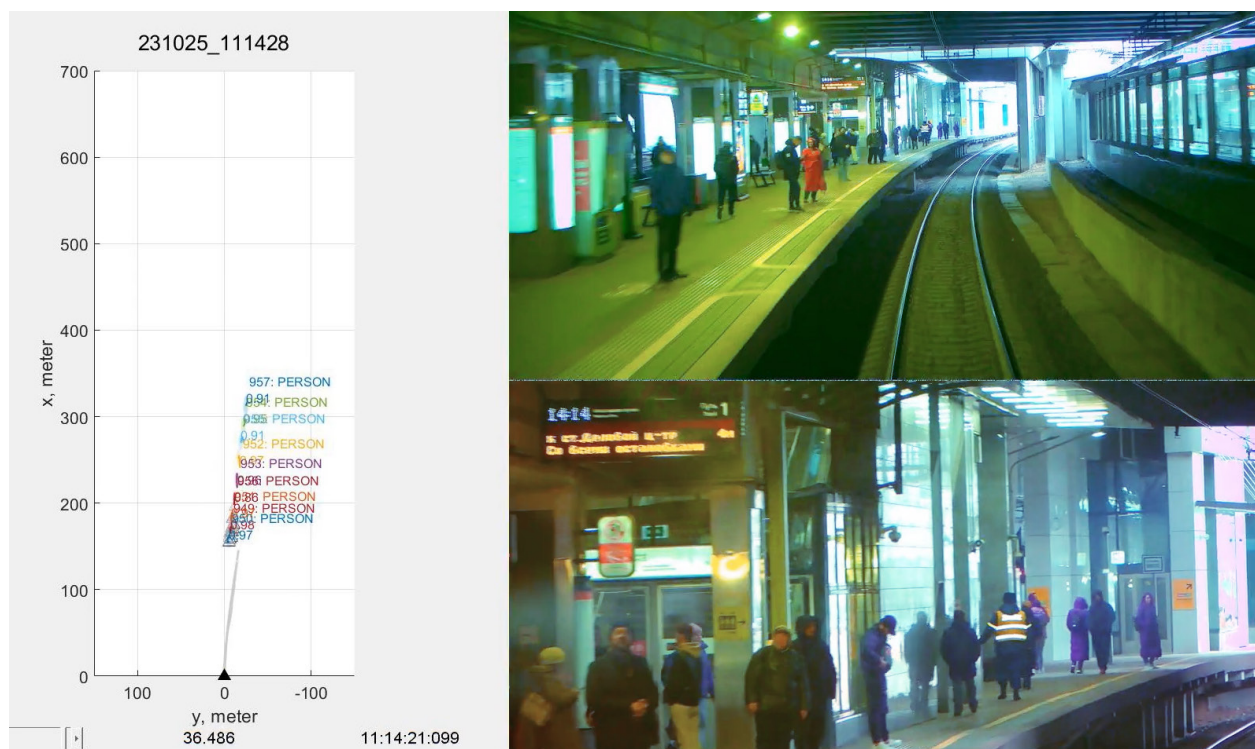


Рис. 4. Пример определения и сопровождения объектов (людей) на платформе

ко всем обнаруженным сенсорами объектам применяется заданное правило (например, из области тормозных позиций удаляются все объекты малого размера, а из области платформ — все объекты типа *person*, расположенные за пределами «красной» зоны).

Таким образом, и *Car*-фильтр, и инфраструктурные фильтры работают по схожему принципу. Отличие заключается только в иницирующем объекте (вагон, вагонный замедлитель, платформа) и правиле, которое применяется к обнаруженным в области фильтра объектам.

7.3. Выбор приоритетных глобальных объектов

В целях оптимизации времени выполнения алгоритма введено ограничение на количество ГО сопровождаемых объектов. Наименее значимые объекты — не подтвержденные, расположенные далеко от опасной зоны — удаляются в случае превышения общего числа объектов над заданным порогом.

Для этого формируется статистика по всем текущим ГО: они распределяются в группы,

соответствующие их классам. Это нужно для того, чтобы в работе остались ГО всех классов и чтобы избежать ситуации, когда будут удалены все ГО одного класса.

В каждой группе рассчитываются веса входящих в нее ГО. Вес отдельно взятого *k*-го глобального объекта рассчитывается по формуле:

$$Q_k = \frac{W_{conf}}{D_L \cdot D_{mean_k}^2},$$

где W_{conf} — вес подтвержденного ГО (например, для подтвержденного $W_{conf} = 1,0$; для неподтвержденного $W_{conf} = 0,2$);

D_{mean_k} — поперечное расстояние от глобального объекта до колеи (п. 6);

D_L — продольное расстояние от глобального объекта до ПС:

$$D_L = \sqrt{X^2 + Y^2},$$

где (X, Y) — координаты объекта в декартовой системе координат подвижного состава [21].

Таким образом, большим значением веса будут обладать самые опасные для ПС объекты: подтвержденные и находящиеся ближе всего

к ПС. Формула носит эмпирический характер, полученные веса используются для сортировки объектов по приоритету для удаления. При обработке большого числа объектов одновременно ГО удаляются таким образом, чтобы в каждой группе-классе осталось одинаковое число этих объектов.

Заключение

Алгоритм комплексирования данных и траекторного сопровождения обнаруженных объектов показал работоспособность, что подтверждено при проведении испытаний на станции Лужская (на маневровом локомотиве) и на Московском центральном кольце (на электропоезде «Ласточка»). Алгоритм позволяет на основе разрозненных данных с различных сенсоров составить общую актуальную картину из реальных объектов, расположенных в поле зрения системы технического зрения. Методы траекторного сопровождения позволяют прогнозировать положение объектов в будущем, а также обеспечивать высокую надежность полученных характеристик объектов-препятствий.

Стоит отдельно отметить, что алгоритм предъявляет определенные требования к качеству входных данных (калибровке камер и лидаров). Соответственно, дальнейшее развитие разработанных алгоритмов СТЗ, траекторной обработки также тесно связано с валидацией входных данных и автоматизацией процесса калибровки сенсоров [22]. Это позволяет минимизировать участие технических специалистов в рамках пуско-наладочных работ и делает систему в целом более устойчивой. Кроме того, необходимо улучшать робастность алгоритма для обеспечения стабильности, предсказуемости работы в нестандартных ситуациях.

Разработка алгоритмов комплексирования данных и траекторного сопровождения — один из необходимых шагов на пути полноценной автоматизации движения как на железной дороге [23], так и на других видах транспорта. Текущий уровень разработок уже позволяет использовать их для автоматизированного управления ПС. Дальнейшие исследования и разработки будут направлены на повышение качества работы алгоритмов и их универсальности. ▲

Библиографический список

1. Розенберг Е. Н., Дзюба Ю. В., Батраев В. В. О направлениях развития цифровой железной дороги // Автоматика, связь, информатика. 2018. № 1. С. 9–13.
2. Охотников А. Л., Попов П. А. Беспилотное управление локомотивом: вчера, сегодня и завтра // Автоматика, связь, информатика. 2019. № 8. С. 12–17. DOI: 10.34649/AT.2019.8.8.002
3. Otte M. W. A Survey of Machine Learning Approaches to Robotic Path-Planning. University of Colorado at Boulder: Boulder, CO, USA, 2009.
4. Dulac-Arnold G., Mankowitz D., Hester T. Challenges of Real-World Reinforcement Learning // arXiv. 2019. URL: arXiv:1904.12901
5. Zhao Y., Zhang Y., Wang S. A Review of Mobile Robot Path Planning Based on Deep Reinforcement Learning Algorithm // J. Phys. Conf. Ser. 2021. Vol. 2138. P. 012011.
6. Deshpande S., Kashyap A. K., Patle B. K. A review on path planning AI techniques for mobile robots // Robot. Syst. Appl. 2023. Vol. 3. P. 27–46.
7. Janji S., Kliks A. Neural Networks for Path Planning // arXiv. 2022. URL: arXiv:2207.00874
8. Bharadwaj H., Kumar E. V. Comparative study of neural networks in path planning for catering robots // Procedia Comput. Sci. 2018. Vol. 133. P. 417–423.
9. Obstacle detection in a field environment based on a convolutional neural network security / T. Li [et al.] // Enterp. Inf. Syst. 2022. Vol. 16. P. 472–493.
10. Nowakowski M., Kurylo J. Usability of Perception Sensors to Determine the Obstacles of Unmanned Ground Vehicles Operating in Off-Road Environments // Appl. Sci. 2023. Vol. 13. P. 4892.
11. Deep Multi-modal Object Detection for Autonomous Driving / A. Ennajar [et al.] // Proceedings of the 2021 18th International Multi-Conference on Systems, Signals & Devices (SSD) (Monastir, Tunisia, 22–25 March 2021). P. 10.
12. Reinoso O., Payá L. Special Issue on Visual Sensors // Sensors. 2020. Vol. 20. P. 910.
13. 3D Recognition Based on Sensor Modalities for Robotic Systems: A Survey / S. Manzoor [et al.] // Sensors. 2021. Vol. 21. P. 7120.
14. Xie D., Xu Y., Wang R. Obstacle detection and tracking method for autonomous vehicle based on three-dimensional LiDAR // Int. J. Adv. Robot. Syst. 2019. Vol. 16. P. 1729881419831587.
15. Сравнение оценок качества комплексирования разнотипных изображений и алгоритмы

- улучшения качества низкоконтрастных изображений для систем видеонаблюдения / С. Н. Кириллов [и др.] // Цифровая обработка сигналов и ее применение — DSPA-2019: доклады 21-й Международной конференции. 2019. С. 403–408.
16. Журавлев Ю. И., Рязанов В. В., Сенько О. В. «Распознавание». Математические методы. Программная система. Практические применения. М.: ФАЗИС, 2006. 176 с.
 17. Bar-Shalom Y., Li X.-R. Multitarget-multisensor tracking: principles and techniques. Vol. 19. Storrs, CT: YBs, 1995.
 18. Эддоус М., Стэнсфилд Р. Методы принятия решений / пер. с англ. под ред. член-корр. РАН И. И. Елисеевой. М.: Аудит, ЮНИТИ, 1997. 590 с.
 19. Охотников А. Л. Ситуационное беспилотное управление // ИТНОУ: Информационные технологии в науке, образовании и управлении. 2019. № 3(13). С. 96–107.
 20. Гиззатов А. С., Иванов В. Ф. Привязка координат подвижного состава к цифровой модели пути // Навигация и управление движением: материалы XXV Юбилейной конференции молодых ученых (Санкт-Петербург, 21–24 марта 2023 года). СПб.: Концерн «Центральный научно-исследовательский институт «Электроприбор», 2023. С. 32–34.
 21. ISO 8855:2011 — Road vehicles — Vehicle dynamics and road-holding ability — Vocabulary. URL: <https://www.iso.org/standard/51180.html>
 22. Способ автоматической калибровки датчиков машинного зрения рельсового транспортного средства: патент № 2811766 С1 Российская Федерация, МПК G06Т 7/80 / И. А. Дейлид [и др.]; заявл. 20.04.2023; опубл. 17.01.2024; заявитель ОАО «РЖД».
 23. Устройство комплексирования данных от обнаруженных препятствий для скоростного электропоезда: патент № 042050 / В. Ф. Иванов [и др.]; заявл. 04.02.2021; опубл. 29.12.2022; заявитель ОАО «РЖД».

TRANSPORT AUTOMATION RESEARCH. 2024. Vol. 10, no. 4. P. 360–371
 DOI: 10.20295/2412-9186-2024-10-04-360-371

Algorithm of complexing sensor data for tasks of automatic control of rolling stock

Information about authors

Ivanov V. F., Head of the Mathematical Support Department of Control Systems². E-mail: v.ivanov@vniias.ru

Okhotnikov A. L., Deputy Head of the Information Technology Department — Head of the Strategic Development Department¹. E-mail: a.okhotnikov@vniias.ru

Gradusov A. N., Developer of the Mathematical Support Department of Control Systems². E-mail: a.gradusov@vniias.ru

¹ JSC “NIIAS”, Moscow

² JSC “NIIAS”, St. Petersburg branch, Saint-Petersburg

Abstract: the article analyzes the research and works on processing Sensor data processing and navigation for mobile objects, including those with automatic control. automatic control. It is pointed out the necessity of integration of solutions based on vision and neural networks, taking into account the state of the environment. on the basis of vision and neural networks, taking into account the state of the environment. The description of the algorithm and methods for complexing data received from sensor sensors of the onboard vision system of the rolling stock. Within the framework of the algorithm work the trajectory data processing: confirmation (removal) of the observed objects, construction of their trajectories, estimation of velocities and coordinates, formation of the list of global objects. To calculate the object motion model, an extended Kalman filter is applied Kalman filter, Hungarian algorithm and calculation of the Mahalanobis distance. Detailed The stages of detection, tracking and identification of obstacle objects are described in detail, as well as determination of their parameters (class, coordinates and velocity). The Experimental indicators used in this study. Relevance relevance and shown practical applicability of the described approach for the tasks of automatic control of rolling stock in the conditions of a digital railroad.

Keywords: vision system, sensor fusion, object identification, object tracking, covariance matrix, clustering, extended Kalman filter, Hungarian algorithm, Mahalanobis distance.

References

1. Rozenberg E. N., Dzyuba Yu. V., Batraev V. V. O napravleniyakh razvitiya Cifrovoy zheleznoj dorogi // Avtomatika, svyaz', informatika. 2018. No. 1. S. 9–13. (In Russian)
2. Okhotnikov A. L., Popov P. A. Bespilotnoe upravlenie lokomotivom: vchera, segodnya i zavtra // Avtomatika, svyaz', informatika. 2019. No. 8. S. 12–17. DOI: 10.34649/AT.2019.8.8.002 (In Russian)
3. Otte M. W. A Survey of Machine Learning Approaches to Robotic Path-Planning. University of Colorado at Boulder: Boulder, CO, USA, 2009.
4. Dulac-Arnold G., Mankowitz D., Hester T. Challenges of Real-World Reinforcement Learning // arXiv. 2019. URL: arXiv:1904.12901
5. Zhao Y., Zhang Y., Wang S. A Review of Mobile Robot Path Planning Based on Deep Reinforcement Learning Algorithm // J. Phys. Conf. Ser. 2021. Vol. 2138. P. 012011.
6. Deshpande S., Kashyap A. K., Patle B. K. A review on path planning AI techniques for mobile robots // Robot. Syst. Appl. 2023. Vol. 3. P. 27–46.
7. Janji S., Kliks A. Neural Networks for Path Planning // arXiv. 2022. URL: arXiv:2207.00874
8. Bharadwaj H., Kumar E. V. Comparative study of neural networks in path planning for catering robots // Procedia Comput. Sci. 2018. Vol. 133. P. 417–423.
9. Obstacle detection in a field environment based on a convolutional neural network security / T. Li [et al.] // Enterp. Inf. Syst. 2022. Vol. 16. P. 472–493.
10. Nowakowski M., Kurylo J. Usability of Perception Sensors to Determine the Obstacles of Unmanned Ground Vehicles Operating in Off-Road Environments // Appl. Sci. 2023. Vol. 13. P. 4892.
11. Deep Multi-modal Object Detection for Autonomous Driving / A. Ennajar [et al.] // Proceedings of the 2021 18th International Multi-Conference on Systems, Signals & Devices (SSD) (Monastir, Tunisia, 22–25 March 2021). P. 10.
12. Reinoso O., Payá L. Special Issue on Visual Sensors // Sensors. 2020. Vol. 20. P. 910.

13. 3D Recognition Based on Sensor Modalities for Robotic Systems: A Survey / S. Manzoor [et al.] // *Sensors*. 2021. Vol. 21. P. 7120.
14. Xie D., Xu Y., Wang R. Obstacle detection and tracking method for autonomous vehicle based on three-dimensional LiDAR // *Int. J. Adv. Robot. Syst.* 2019. Vol. 16. P. 1729881419831587.
15. Svravnenie ocenok kachestva kompleksirovaniya raznospektral'nykh izobrazhenij i algoritmy uluchsheniya kachestva nizkokontrastnykh izobrazhenij dlya sistem videonablyudeniya / S. N. Kirillov [i dr.] // *Cifrovaya obrabotka signalov i eyo primeneniye* — DSPA-2019: doklady 21-j Mezhdunarodnoj konferencii. 2019. S. 403–408. (In Russian)
16. Zhuravlev Yu. I., Ryazanov V. V., Sen'ko O. V. "Raspoznavanie". *Matematicheskie metody. Programmaya sistema. Prakticheskie primeneniya*. M.: FAZIS, 2006. 176 s. (In Russian)
17. Bar-Shalom Y., Li X.-R. *Multitarget-multisensor tracking: principles and techniques*. Vol. 19. Storrs, CT: YBS, 1995.
18. Ehddous M., Stehnsfeld R. *Metody prinyatiya reshenij / per. s angl. pod red. chlen-korr. RAN I. I. Eliseevoy*. M.: Audit; YUNITI, 1997. 590 s.
19. Okhotnikov A. L. *Situacionnoe bespilotnoe upravlenie* // ITNOU: Informacionnye tekhnologii v nauke, obrazovanii i upravlenii. 2019. No. 3(13). S. 96–107. (In Russian)
20. Gizzatov A. S., Ivanov V. F. *Privyazka koordinat podvizhnogo sostava k cifrovoy modeli puti // Navigaciya i upravlenie dvizheniem: materialy XXV Yubilejnoj konferencii molodykh uchenykh (Sankt-Peterburg, 21–24 marta 2023 goda)*. SPb.: Koncern "Central'nyj nauchno-issledovatel'skij institut "Ehlektronpribor", 2023. S. 32–34. (In Russian)
21. ISO 8855:2011 — Road vehicles — Vehicle dynamics and road-holding ability — Vocabulary. URL: / <https://www.iso.org/standard/51180.html>
22. *Sposob avtomaticheskoy kalibrovki datchikov mashinnogo zreniya rel'sovogo transportnogo sredstva*: patent № 2811766 C1 Rossijskaya Federaciya, MPK G06T 7/80 / I. A. Dejlid [i dr.]; zayavl. 20.04.2023; opubl. 17.01.2024; zayavitel' OAO "RZD". (In Russian)
23. *Ustrojstvo kompleksirovaniya dannykh ot obnaruzhennykh prepyatstvij dlya skorostnogo ehlektropoezda*: patent № 042050 / V. F. Ivanov [i dr.]; zayavl. 04.02.2021; opubl. 29.12.2022; zayavitel' OAO "RZD". (In Russian)