

$\beta_1 = 0,9$, $\beta_2 = 0,999$ – коэффициенты, используемые для расчёта скользящих средних градиентов и их квадратов;

$\epsilon = 1e-8$ – слагаемое, добавляемое к знаменателю для улучшения числовой стабильности;

$\text{weight_decay} = 0$ – L2-регуляризация.

При обучении датасет был разбит на 2 множества: для обучения (90 % от общего числа изображений) и валидации (10 %). Также в процессе обучения на каждой итерации использовались следующие преобразования изображений с целью аугментации датасета: случайное изменение масштаба от 90 % до 110 %; случайный поворот изображения от -75 до $+75$ градусов; случайное отражение изображения по горизонтали.

Нейронная сеть была реализована на языке Python с использованием фреймворка PyTorch.

Предложенная в настоящей работе нейронная сеть способна обрабатывать 10 изображений в секунду на ноутбуке с графическим ускорителем NVIDIA GeForce GTX 1060 with Max-Q Design, центральным процессором Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz и 16GB оперативной памяти под управлением ОС Ubuntu 16.04. Общее потребление памяти составило не более 2 ГБ.

Проведенные исследования показывают, что совместное использование сети из работы [1] и предложенной дает прирост точности предложенной модификации 1,4% (98,3% против 96,9%) несмотря на то, что вторая реализация обучалась на меньшем общем количестве изображений.

Ключевыми отличиями, повлиявшие на результат, являются: размер входного слоя (512×512 в предлагаемой сети против 256×256 у сети [1]) и глубина нейронной сети (4 пары «энкодер-декодер» против 3 пар у [1]). Они прямо влияют на количество обучаемых параметров и, как следствие, качество сегментации. Из негативных последствий таких изменений можно отметить большее потребление ресурсов.

Список литературы

1. Ganchenko, V. Image Semantic Segmentation Based on Convolutional Neural Networks for Monitoring Agricultural Vegetation / V. Ganchenko, A. Doudkin // Communications in Computer and Information Science, Springer, 2019. – Ch. 5. – V. 1055. – 2019. – P. 52-63.
2. Adam: A Method for Stochastic Optimization [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1412.6980> – Дата доступа : 01.05.2006.

УДК 004.415.25

БЕСПИЛОТНЫЙ ТРАНСПОРТ В АВТОМАТИЗАЦИИ ПРОИЗВОДСТВ

Я. О. Заречный, Е. В. Василюк, А. О. Заречный

Брестский государственный технический университет, Брест

Научный руководитель: В.Н. Шуть, доцент, кандидат технических наук

Роботы – это физические агенты, которые выполняют поставленные перед ними задачи, проводя манипуляции в физическом мире. На данный момент все большую перспективность приобретает мобильная робототехника – область ро-

бототехники, где роботы автономны и способны самостоятельно выполнять широкий спектр различных задач от мониторинга окружающей среды до проведения экскурсий для туристов. Управление такими роботами является сложной технической задачей. Существует большое количество подходов к управлению роботами и робототехническими системами, наиболее перспективным является интеллектуальное управление[1].

Интеллектуальное управление – высшая ступень управления в теории автоматического управления после программного и адаптивного. Оно основано на применении методов искусственного интеллекта.

На данный момент существует большое множество различных средств моделирования мобильных роботов, а также большое количество инструментов продолжает появляться. Большое разнообразие дает много положительных моментов, однако и порождает следующие нерешенные проблемы:

- каждый отдельный инструмент моделирования мобильных роботов направлен на решение своего определенного класса задач, универсальных решений нет;

- различные инструменты предъявляют различные требования к построению моделей мобильных роботов;

- большая часть работ направлена на доработку конкретных недостатков существующих средств моделирования, но практически отсутствуют работы по применению комбинированного подхода и построению комплексных моделей с использованием различных сред моделирования;

- отсутствуют хорошо описанные методики применения различных средств моделирования в разработке прототипов мобильных роботов и анализа достоверности моделей.

Проанализировав ситуацию в области компьютерного моделирования робототехнических систем, можно сделать вывод, что перспективным направлением является разработка комплексных методик моделирования мобильных роботов с использованием различных инструментов моделирования. Этот подход в перспективе даст более точные модели и более гибкий процесс разработки мобильных роботов.

В последние годы наблюдается рост интереса среди ученых и производителей автотранспорта к беспилотным автомобилям, способным перемещаться по дорогам без участия человека. Особое место в развитии беспилотного транспорта занимают так называемые беспилотные тележки. Они уже сейчас используются в складских помещениях, заводских цехах, в некоторых крупных портах для автономного перемещения грузов. Перспектива их применения довольно широка: подвоз комплектующих со склада на сборочный участок, отвоз готовых изделий от металлообрабатывающих станков на промежуточный склад хранения и т. д.

Беспилотная тележка (AGV – Automatic guided vehicle) – транспортер с электроприводом, предназначенный для перемещения грузов. Тележка автоматическая, а это значит, что для ее обслуживания не нужен отдельный оператор - тележки двигаются по заданной траектории в автономном режиме без участия человека.

Беспилотные тележки разделяются на два типа: с компьютерным зрением и без него. Разработка транспорта первого типа довольно затратная и в плане ра-

бочего персонала, и в плане денег. Сами тележки имеют в себе программное обеспечение, в которое входит нейросеть, вычисляющая траекторию пути и обеспечивающая безопасность окружающих.

Беспилотная тележка снабжена всеми необходимыми системами и элементами безопасности, может эксплуатироваться на вредных или опасных производствах, местах скопления людей и других двигающихся тележек.

В зависимости от типа, беспилотная тележка работает:

- 1) буксиром, перевозя другие тележки;
- 2) перевозчиком, поднимая и перевозя на себе грузы (стеллажи, паллеты и т. д.).

Основные направления работы:

- 1) движение по предварительно определенной траектории различной формы, включая развилки;
- 2) поддержание постоянной скорости в случае сопровождения конвейера или других объектов;
- 3) остановки и продолжение движения, как в заранее определенных позициях, так и "по требованию" оператора;
- 4) загрузка и разгрузка перевозимых компонентов;
- 5) беспроводная связь с другими тележками или центральным терминалом для создания сети тележек, движущихся без участия людей;
- 6) картографирование неподвижных препятствий для проезда на минимальном расстоянии;
- 7) распознавание перемещающихся препятствий, ожидание их исчезновения и продолжение работы.

Следуя по маршруту, беспилотная тележка может:

- 1) останавливаться и снова начинать движение;
- 2) сопровождать конвейер на постоянной скорости;
- 3) выполнять другие запрограммированные действия.

В структуру автоматически управляемой тележки входят:

- 1) блок привода, с 2-мя независимыми электромоторами, обеспечивающий тягу и поворот на маршруте следования;
- 2) блок энергообеспечения, содержащий набор герметичных необслуживаемых аккумуляторов (разрешены для применения в помещениях, где находятся люди, не требуют специальной комнаты для зарядки);
- 3) блок управления с программируемым контроллером, отвечающий за процесс движения;
- 4) система безопасности и оповещения, включающая в себя свето-звуковую сигнализацию при движении и сертифицированный ультразвуковой дальномер;
- 5) система навигации, позволяющая реализовать движение по заданной траектории;
6. пульт управления.

Использование этих систем позволяет беспилотной тележке следовать по заданному маршруту, включая развилки и повороты, контролировать препятствия на пути следования, останавливаться при их наличии и продолжать движение при первой возможности.

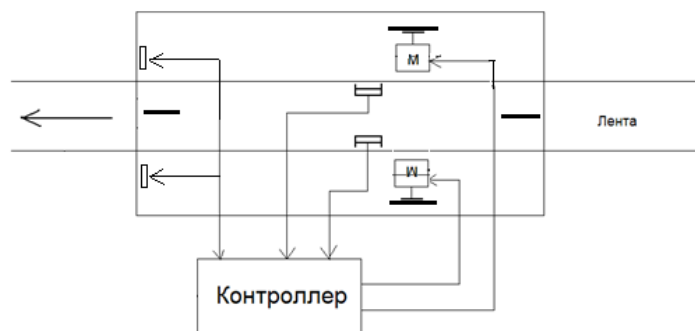


Рисунок 1 – Структура беспилотной тележки

Беспилотная тележка имеет четыре колеса: два из них ведущие, остальные поддерживающие. Поддерживающие колеса вращаются на 360 градусов, чтобы в случае чего тележка могла развернуться вокруг своей оси и продолжить движение без сторонней помощи. Также тележка имеет три типа датчиков. Индуктивные и инфракрасные датчики, расположенные в самом низу тележки, будут считывать траекторию магнитной ленты или чёрной линии. Датчики, расположенные впереди автоматического транспорта, должны предотвращать любое столкновение с препятствием, поэтому применяются ультразвуковые датчики расстояния. Все сигналы, получаемые датчиками, обрабатываются контроллером, который вырабатывает управляющий сигнал для драйверов привода.

Беспилотная тележка следует по предварительно определенной траектории. Старт движения может осуществляться по нажатию кнопки или событию: начало смены, прибытию груза, сигнал оператора.

Три варианта определения маршрута беспилотной тележкой:

1. По металлической ленте - наклеивается на пол, не мешает, легко создавать развилки, остановки и повороты. При таком режиме работают индуктивные датчики, которые выдают сигнал при наличии магнитного поля.

2. По чёрной линии – дешёвый способ, но при перестройке маршрута придётся стирать линию. При таком режиме работают ИК-датчики (датчики чёрной линии), которые работают по принципу отражения инфракрасного луча от поверхности пола.

3. Комбинированный режим – на пол наклеивается металлическая лента и покрывается чёрным матовым лаком. Более дорогой способ, однако самый надёжный. При этом режиме работают одновременно два типа датчиков.

В настоящее время использование беспилотного транспорта является малоизученным, однако в дальнейшем может быть внедрено для автоматизации различных производств. Использование такой системы позволит увеличить прибыль и ускорить транспортировку груза на предприятии.

В управлении тележки заложена остановка в случае обнаружения препятствия. Для этого в бампере тележки с обеих сторон находится по два ультразвуковых датчика. Такой датчик работает по принципу сонара, посылая пучок ультразвука, и получая его отражение с задержкой. Этим устройством управления определяется наличие объектов и расстояние до них. Ультразвуковые сигналы, генерируемые приёмником, отражаясь от препятствия, возвращаются к нему через определённый промежуток времени. Именно этот временной интервал становится характеристикой, помогающей определить расстояние до объекта.

Список литературы

1. Юревич, Е. И. Основы робототехники / Е. И. Юревич. – 2 –е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2005. – 416 с.
2. Полностью автономные мобильные роботы OMRON LD 60/90 [Электронный ресурс]. – Режим доступа: <https://industrial.omron.ru/ru/products/ld-60-90>. – Дата доступа: 21.08.2021.
3. 3D-технологии Автоматически управляемые тележки [Электронный ресурс]: – Режим доступа: <http://agvrobot.ru>. – Дата доступа: 22.08.2021.

УДК 004.67

РАЗРАБОТКА ФУНКЦИЙ ОБРАБОТКИ ДАТ В C++

Е. А. Кривоносова, Т. Г. Хомицкая

Брестский государственный технический университет, г. Брест

Постановка задачи. Сфера применения языка программирования C++ широка. Даже в сегодняшнем мире, несмотря на всё многообразие современных надёжных инструментов, он остается незаменимым. Поэтому нередко C++ находит свое применение в системном программировании, при создании операционных систем, драйверов, различных утилит, антивирусов и т. д. Также C++ можно использовать в программах любого уровня, где важны скорость работы и производительность. Нередко он применяется для создания графических приложений, различных прикладных программ. Однако, также можно получить интересные результаты для решения прикладных задач используя только условный оператор *if()*.

В силу специфики специальности «экономика электронного бизнеса» часто приходится сталкиваться с обработкой чисел и дат. В стандартных библиотеках C++ не так много встроенных функций для работы с последними. В том числе нет возможности для автоматического нахождения разницы между двумя датами. Поэтому наша задача заключалась в реализации функции поиска разницы между данными, представленными в форме соответствующей структуре «дата» (день / месяц / год).

Целью нашей работы является разработка алгоритма для нахождения разницы между двумя датами и его реализация на C++.

Разработка алгоритма и программы. С первого взгляда задача разработать алгоритм может показаться несложной. Однако, в расчетах необходимо учитывать многие факторы, такие как разнообразное количество дней в каждом месяце, високосные года, а также ситуацию, когда день первой даты больше дня второй даты, т. е. невозможность просто вычесть дни друг от друга. Эти нюансы регулировались с помощью анализа данных и выбора действий в зависимости от полученного результата, т. е. в основу был заложен разветвляющийся алгоритм.

Программа, составленная на основе разработанного алгоритма, работает в двух режимах, в зависимости от выбора пользователя: вычисляется разница между двумя датами (рис. 1) либо вычисляется разница между системной датой и датой, введенной пользователем (рис. 2).

Для определения количества дней в месяце, в том числе в зависимости от того является ли год високосным или нет, была разработана функция *checking()*, которая позволяет регулировать различные нюансы при обработке введенных дат. Например, при анализе дат оказывается, что первая дата наступила позже второй даты или в первой дате число, определяющее день, больше аналогичного числа во второй дате. Программа работает с любыми существующими датами, учитывая все вышесказанные нюансы (рис. 3, 4).