РЕАЛІЗАЦІЯ НАВЧАЛЬНОЇ СИСТЕМИ КЕРУВАННЯ РАДІОКЕРОВАНИМ ВІЗКОМ

Ігор Арсенюк[1], Володимир Месюра[2], Юрій Ляшенко[3]

Вінницький національний технічний університет
Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 43-78-80,
E-Mail: air@vstu.vinnica.ua [1], Volodymyr.Mesyura@gmail.com [2], Yuri.Lyashenko@gmail.com [3]

Abstract

The main purpose of the researches represented in the article is to develop a full-functional system for the radio-controlled carriage handling. The system that is being developed has to provide possibilities to implement movement algorithms, recognition algorithms, tuning algorithms. This can be applied for the both research and educational purposes.

Актуальність

Дослідження в галузі робототехніки сприяють автоматизації багатьох процесів повсякденного життя. Безсумнівно, цей напрямок дістає бурхливого розвитку разом із комп'ютеризацією та автоматизацією усіх виробничих процесів. Цим обумовлюється корисність знань у навіть елементарних автоматизованих системах. Система, що пропонується може застосовуватись як для моделювання результатів досліджень, так і з навчальної метою – формування знань по функціонуванню автоматичних систем, тощо. В даний час спостерігається безліч дослідницьких проектів в області інтелектуальних транспортних засобів. Дослідники аналізують можливості застосування автоматично керованих транспортних засобів у багатьох інших застосуваннях в галузях сільського господарства, знешкодження мін, аварійно-рятувальних робіт і т. д. Це дозволило б зменшити рівень ризику людей та підвищити ефективність використання транспортних засобів. Проте в більшості галузей промисловості найвищий інтерес представляє автоматизація транспортних засобів.

Аналіз сучасного стану проблеми

В даній роботі використовуються кілька технологій та розв'язуються декілька задач пов'язаних з алгоритмами керування, навігації, розпізнавання, моделювання тощо. Зробимо короткий огляд досліджень та здобутків у галузі автоматизації навігації.

В кінці 1990-х рр. виконувалися проекти ARGO і Surface Antarctic Robot (RAS) [1]. В рамках проекту ARGO був розроблений один з перших прототипів, що демонструє можливості автоматичного управління транспортним засобом на дорогах. Були розроблені, протестовані і налаштовані декілька рішень для автономного переміщення з вибором маршруту руху і виявленням перешкод. У проекті RAS досліджувалися можливості автоматичного управління транспортними засобами в екстремальних умовах. Останніми роками однією з областей дослідження є захист вразливих учасників дорожнього руху. У співпраці з дослідницьким центром Фольксваген компанія VisLab розробила прототип – систему розпізнавання пішоходів для додатків запобігання аваріям та допомозі водіям.

Проводяться дослідження можливості застосування близьких до інфрачервоного спектру прожекторів і камер для локалізації перешкод. У автомобільній індустрії також виявляється інтерес до додатків допомоги водіям на основі кольорового машинного зору.

У військовій області центром TARDEC (Tank Automotive Research, Development and Engineering Center) збройних сил США розробляється система, оснащена чотирма відеокамерами, призначена для підвищення рівня безпеки робото-технічних систем. Для компанії Oshkosh Truck Corp. розробляється додаток нічного бачення для застосування у військових цілях. Ось уже кілька років останні здобутки у галузі навігації та автоматичного розпізнавання можна побачити у «Великих гонках роботів» (DARPA Grand Challenge), які щорічно відбуваються та являють собою перегони транспортних засобів керованих комп'ютерами. Для реалізації сучасних систем навігації використовуються машинний зір, лазерні сканери, GPS, інерційні датчики і бази даних, що забезпечують розуміння середовища.

Мета досліджень та постановка задачі

Метою даної роботи є розробка повнофункціональної системи для керування радіокерованим візком. Система має забезпечувати можливість підключення самостійно реалізованих алгоритмів керування візком, алгоритмів розпізнавання, калібрування, тощо. Отже, сформулюємо такі задачі дослідження:

- реалізувати гнучку систему керування радіокерованим візком;
- забезпечити чітку і зрозумілу архітектуру програмного продукту для можливості використання системи з навчальною метою;
- реалізувати модульну структуру з можливістю підключення самостійно розроблених модулів.

Підхід до розв'язання поставленої задачі

Для розв'язання задачі насамперед виділимо основні функціональні модулі та наведемо загальний алгоритм роботи програмного продукту. Вхідними даними для системи є зображення, яке сприймається за допомогою Web-камери. Перетворення знятих кадрів у внутрішні об'єкти програми виконується завдяки можливостям JMF 2.0 (Java Media Framework) [2, 3]. Далі відбувається перетворення зображення у внутрішню мапу світу. Для цього можуть застосовуватись різні алгоритми розпізнавання, тому цей модуль повинен бути чітко стандартизований та реалізований так, щоб його можна було замінювати самостійно розробленими іншими модулями, і при цьому на втрачалась загальна функціональність системи.

Наступним кроком є реалізація зворотної відповідності між внутрішньою мапою світу, реальними об'єктами та реальним простором в якому рухатиметься радіокерований візок. Необхідно реалізувати модуль для налаштування швидкості руху радіокерованого візка. Цей модуль використовуватиметься щоразу, як виконуватиметься задача у нових умовах. Рух візка відбувається шляхом подачі сигналу на LPT порт комп'ютера. Тривалість сигналу визначає тривалість руху, але в залежності від відсотку заряду накопиченого в акумуляторних батареях радіокерованого візка, від ступеня тертя шин візка з поверхнею, від рельєфності поверхні, відстань пройдена за один і той же час, так само як і радіус повороту можуть, і будуть відрізнятись. Тому цей модуль також повинен бути стандартизований з можливістю заміни.

Після того як етап налаштування завершено, необхідно реалізувати оптимальний алгоритм руху по карті світу. Тут можуть бути застосовані різноманітні алгоритми пошуку оптимального шляху, в залежності від конкретної поставленої задачі. Таким чином цей модуль також має бути реалізований з можливістю заміни та з відсутністю прямого впливу на функціональність інших модулів.

На рисунку 1 наведено схему пакетів та інтерфейсів які пропонуються для реалізації поставленої задачі. При створенні моделі програмного засобу було дотримано таких основних аспектів:

- розмежування інтерфейсної, функціональної та логічної частин;
- відокремлення модулів, що мають різні функціональні призначення;
- відокремлення даних від методів обробки;
- запровадження чіткої системи наслідування класів;
- організація рівнів доступу до методів класів (приватні, захищені чи загальні).

На даній схемі можна бачити як відбувається передача основних даних та відокремлюється функціональність. Можливість впровадження окремої реалізації кожного модуля реалізовано з використанням технології Pattern Provider [5] на платформі Java.

Отже, інтерфейс IRecoznizer вимагає самостійної реалізації методів recognize(), який відповідає за перетворення відзнятого знімку у внутрішню карту світу, та методу getCarPosition(), що має повертати координати радіокерованого візка у поточний момент часу.

Інтерфейс IWayFinder вимагає реалізації методу find(), який відшукує оптимальний шлях у внутрішній карті світу та формує його у вигляд послідовності координат точок. Всі точки із знайденими координатами надалі оброблятимуться наступними модулями та становлять шлях.

Інтерфейс ISpeedTunning вимагає реалізації метода tuneCar(). Цей метод налаштовує внутрішні параметри системи такі як швидкість, реальна швидкість, радіуси поворотів тощо. В процесі моделювання цей метод використовуватиметься декілька разів перед наступним кроком, для найбільш адекватного налаштування системи.

І, нарешті, останнім кроком є реалізація методу drive() інтерфейсу IRacer. Цей метод, використовуючи попередньо налаштовані параметри, визначає тривалість сигналів які необхідно подати для забезпечення руху радіокерованого візка у потрібному напрямку.

Зауважимо, що реалізована система не подає сигнали на LPT порт безпосередньо. Сигнали подаються спеціальним C++ класом, який працює із драйвером вказаного порту. Необхідність використання драйвера обумовлена тим, що програмний продукт орієнтовано для роботи на платформі операційних система Windows XP та вище. Особливістю таких систем є те, що користувацькі додатки виконуються з рівнем доступу «3». Для подачі сигналів на LPT порт та зчитування сигналів з нього необхідний рівень доступу «0». Досягти цього можна двома шляхами:

- використовуючи недокументовані можливості ОС змінювати рівні доступу запущених процесів;
- використовуючи стандартні або самостійно розроблені драйвери під відповідну версію ОС керувати сигналами на потрібних портах.

Враховуючи небезпеку і незручність використання першого підходу реалізовано було другий. Відповідний C++ клас було зібрано у DLL, та з використання JNDI [4] зв'язано із системою, що розробляється.

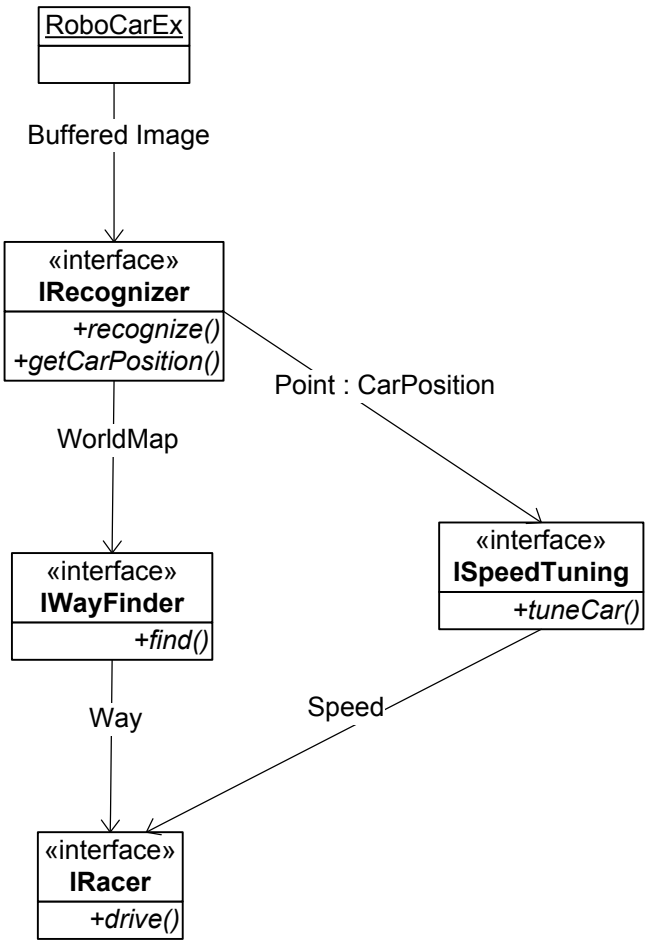

Рис.1.Схема пакетів та інтерфейсів системи керування радіокерованим візком

Висновки

При виконанні даної роботи було розроблено і реалізовано модель програмного продукту для керування радіокерованим візком. Архітектура задовольняє основним вимогам поставленим у завданні – гнучкості та можливості розширення. Враховуючи рівень розвитку сучасних навігаційних систем, використання ними лазерних технологій та інше, можна визначити основний недолік розробленого продукту – складність визначення реальної відстані до об'єкта. Власне, використовуючи тільки одну камеру неможливо визначити реальну відстань до об'єкта. Щонайменше потрібно дві камери, та шляхом імітації бінокулярного машинного зору, що передбачає складні математичні обрахунки, можна визначити відстань до об'єкта. Але метою роботи є забезпечити можливість моделювання результатів досліджень, зазвичай у лабораторних умовах. Тому запропонована програма повністю відповідає поставленій меті та дозволяє реалізувати обчислювальні та розпізнавальні алгоритми будь-якої складності та степені точності.

Зазначимо, що запропонована система експериментально застосовується з навчальної метою для вивчення, дослідження і реалізації студентами різних алгоритмів розпізнавання образів та раціонального подолання перешкод тощо.

Література

[1] Computer // IEEE Computer Society, V. 39, No 12, Грудень, 2006.
[2] Ноутон П., Шилдт Г. JAVA 2, 2007.
[3] Арнольд К., Гослинг Д., Холмс Д. Язык программирования Java, 4-е издание. – 2004.
[4] Java Data Access: JDBC, JNDI, and JAXP by Todd M. Thomas. – 2007.
[5] Robert C. Martin – UML for Java Programmers. – 2003.
[6] Foley, Van Dam, Feiner, and Hughes Computer Graphics // Principles and Practice. Addison Wesley. – Ch. 16. – 1996. – pp. 800-870.