УДК 004.3

С.О. Цололо, канд. техн. наук,
Д.Ю. Бровкіна, аспірант,
А.О. Кравецький, студент
Донецький національний технічний університет
daniella.brovkina@gmail.com

# Розробка системи керування мобільним роботом з можливістю виявлення перешкод на основі відеоданих

*В роботі запропонована система діалогового керування мобільним роботом, що створений на базі обчислювальної платформи Arduino UNO R3. Розроблене програмне забезпечення вирішує завдання обходу перешкод за допомогою аналізу відеоданих, що надходять з одиночної камери на роботі.*

***Ключові слова: мобільний робот, обхід перешкод, оптичний потік.***

## *Вступ*

На сьогоднішній день автоматизація і використання роботів надійно зайняли нішу у виробництві, проте це повністю не позбавляє від необхідності здійснювати людиною контроль або безпосереднє втручання за повністю або частково автоматизованим процесом. Не у всі місця людина може потрапити, це пов'язано з певними ризиками для здоров'я і життя, а також з фізичними можливостями людського тіла, таким чином виникає необхідність у віддаленому спостереженні та управлінні. 80-90 відсотків інформації людина отримує через органи зору. З цього випливає, що реагувати швидше людина буде, побачивши ситуацію. Звідси постає питання про важливість передачі відеоданих в реальному часі.

З іншого боку сучасні роботи відносяться до нового етапу свого розвитку — створення інтелектуальних роботів. Сучасний робот володіє мобільністю, здатний самостійно досліджувати навколишнє середовище, вирішувати ряд завдань, передбачених розробниками. Проте все це не гарантує його повної самостійності. Саме тому тема віддаленого керування є актуальною.

Таким чином, у роботі в якості об'єкта дослідження розглядається мобільний робот на платформі Arduino UNO, а предметом розробки є програмне забезпечення аналізу відеоданих з камери робота. Практична цінність роботи полягає в реалізації робота та розподіленого програмного забезпечення до нього.

## *Апаратна платформа*

Оскільки вирішено знаходити перешкоди за допомогою аналізу відеоданих, то в якості датчика робота виступає камера. Варіантів обробки відео два:

1. Потік обробляє безпосередньо бортовий комп'ютер робота, тому необхідний досить потужний мікроконтролер або мікрокомп'ютер.
2. Потік транслюється на робочу станцію оператора, і після аналізу робоча станція формує команду і передає її роботу.

Через те, що відеодані в будь-якому випадку повинні передаватися на термінал оператора, вирішено реалізовувати другий варіант. Значною перевагою цього вибору є те, що мобільний робот не потребує потужного бортового комп'ютера, а отже можна обійтися мікроконтролером сімейства ATtiny або ATmega фірми Atmel. Також слід зазначити, що при такій обробці даних не подтрібен зв'язок між камерою та основними модулями самого робота, що робить вибір апаратних складових робота ширшим, а також має пришвидшити роботу системи, тому що дані з камери та команди керування передаються по окремим каналам.

Функціональну схему системи, що була розроблена для реалізації інтерактивної системи керування з передачею відеоданих оператору, відображено на рис. 1.

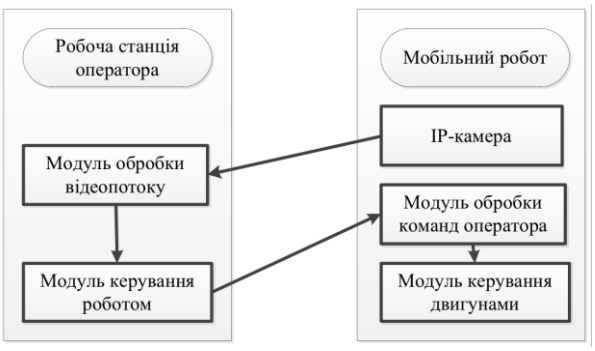


Рисунок 1 – Функціональна схема системи

Згідно зі схемою камера не пов'язана схематично з роботом, тож кріпиться вона механічно, без підключення до інших модулей робота. Безпосередньо з двигунами платформи взаємодіє модуль керування двигунами L298N. На платформі розташовано два двигуни: поворотний двигун та

двигун для руху. Модуль передбачає підключення двох двигунів, при цьому можна керувати кожним двигуном окремо, та задавати напрямок руху кожного з них.

Побудова апаратної частини починається з вибору камери та рухомої платформи [1]. Камера необхідна для передачі відеоданих оператору та виконання на основі цих даних обходу перешкод. На платформі ж буде розташована решта обладнання. При цьому камера повинна відповідати наступним вимогам:

1. Надавати змогу вибирати ключові параметри зображення.
2. Зберігати зображення в форматі, що підтримується бібліотекою з відкритим доступом.
3. Надавати змогу обирати необхідний час між кадрами.

Усім вимогам відповідає телефон з системою Android 4.4.2 зі стороннім контентом «IP Webcam» [2]. Дана програма дозволяє отримувати послідовність кадрів у форматі MJPEG.

В роботі у якості модуля обробки команд та керування роботом використана обчислювальна платформа Arduino UNO R3 (рис. 2) [3]. Ця платформа виконана на базі контролера ATMega328 [4-6].

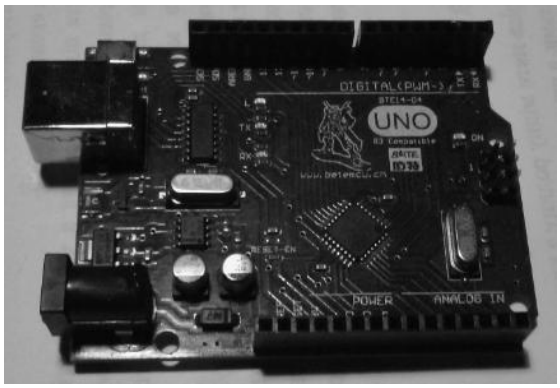


Рисунок 2 – Обчислювальна платформа Arduino UNO R3

В якості передавача було вирішено використати модуль ESP8266 ESP01 (рис. 3) [7] (ESP01 – версія випуску, на даний момент існує 12 версій). Головними критеріями вибору є радіус дії, швидкість передачі, наявність прошивки зі зручною системою команд для налаштування та передачі даних, підтримка інтерфейсу UART, підтримка стеку TCP/IP.

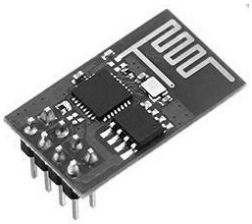

Рисунок 3 – Wi-Fi-модуль ESP8266

В якості модуля керування двигунами було обрано драйвер L298N [8] (рис. 4). Головним критерієм вибору була напруга – до 35 В, яку може витримати даний модуль та сумісність з Arduino.

Рисунок 4 – Драйвер для шагових двигунів L298N

У якості пульта керування виступає робоча станція з програмним забезпеченням, що було розроблено. На рис. 5 відображено повну функціональну схему розробленої системи з урахуванням обраних компонент.

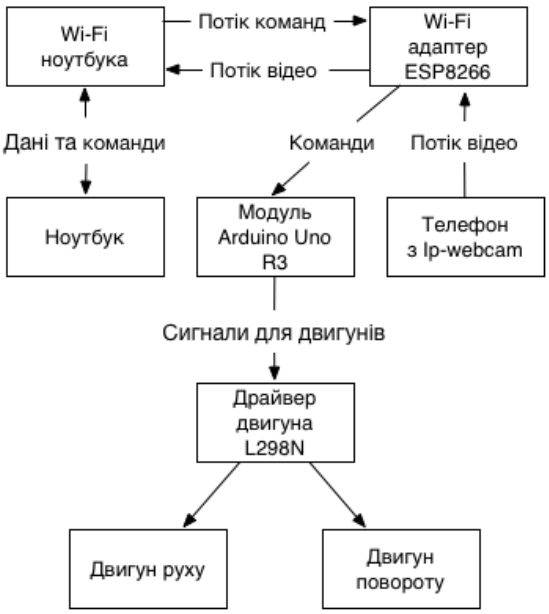


Рисунок 5 – Функціональна схема системи керування з урахуванням обраного обладнання

### *Програмне забезпечення*

Першим кроком у розробці ПЗ є написання програми контролера. Дана програма дозволить перевірити працездатність та правильність зібраної схеми. Перш ніж описати алгоритм роботи, необхідно визначити команди, які буде виконувати контролерна плата Arduino. Перелік необхідних команд наведений у табл. 1. Також в цій таблиці наведена необхідна кількість даних для виконання конкретної команди. Також необхідно привести розрахунки щодо допустимої довжини пакета команд.

Таблиця 1 – Опис команд робота

| Команда | Код команди | Довжина (байт) | Пояснення |
|---|---|---|---|
| Рух вперед | "W" | 2 | Другий байт для ШІМ |
| Рух назад | "S" | 2 | Другий байт для ШІМ |
| Зупинка | "P" | 1 | Вимкнення двигуна руху |
| Змінити швидкість | "K" | 2 | Другий байт для ШІМ |
| Поворот ліворуч | "A" | 1 | – |
| Поворот праворуч | "D" | 1 | – |
| Кінець повороту | "T" | 1 | Вимкнення двигуна повороту |
| Перевірка зв'язку | "R" | 1 | Отримується кожні 2 с |

Розрахунок допустимої довжини пакету команд виконується наступним чином. Необхідно розглянути випадок виявлення роботом перешкоди. Перед зіткненням робот може зробити дві дії:

– зупинитися;

– повернути (поворот ліворуч чи праворуч слід вважати рівноправними з приводу симетрії робота).

До початку процесу гальмування контролер повинен отримати команду на відстані S з запасом в 1 корпус машинки (L = 26см) та обробити її за час, коли робот пройде пів-корпусу (13 см), це вважається нормою реагування [9]. Процес зображено на рис. 6.

При зупинці прискорення роботу а становить 0,4 м/с$^2$, а максимальна швидкість руху робота – 0,6 м/с. Таким чином, робот проходить відстань 45 см за 2 с. На швидкості 0,6 м/с робот проходить 13 см за 0,2 с. За даний час з моменту прийому контролер повинен прочитати X байтів команди та обробити їх, вимкнувши двигун. Процедура читання байту з UART на конролері сімейства ATMega займає до 19 тактів, розпізнавання символу з масиву – 6 тактів, а подача сигналу на двигун з використанням ШІМ – 6 тактів.

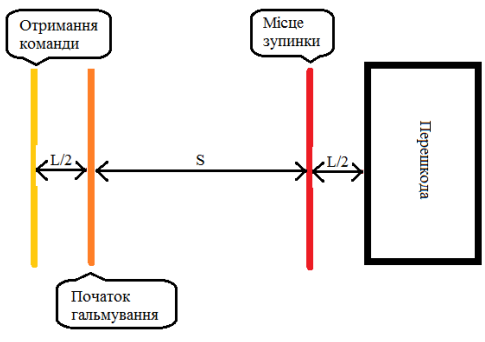


Рисунок 6 – Процес зупинки перед перешкодою

Таким чином, контролер повинен затратити T тактів для зупинки, T розраховується як

$$T = X*(19+6)+6 \quad (1)$$

Це становитиме 0,2 с. При швидкості тактування 16 МГц, контролер зможе обробити X=127999 байт. Під час розвороту контролер повинен відреагувати за той самий час (час проходження відстані, що дорівнює половині довжини корпуса) і подати сигнал на двигун повороту за 2 такти. За таких умов контролер зможе обробити команду тієї ж довжини, що й в попередніх розрахунках.

Тож виявлено, що процесор зможе обробити команду довжиною 127 тисяч символів, тож довжина команди навіть у декілька десятків символів є допустимою для уникнення зіткнення до завершення обробки команди від оператора.

### *Задача обходу перешкод*

Для розв'язання задачі обходу перешкод на основі відеоданих необхідно визначити наявність об'єкту на кадрі. Існують різноманітні засоби та методи для виявлення об'єктів на фоні, котрі базуються на сегментації зображення. Оскільки з камери можна отримувати ряд послідовних зображень, то розпізнавання перешкод стає легшим, за рахунок того, що об'єкти, які знаходяться ближче, змінюються на зображеннях швидше, аніж фон та віддалені об'єкти. Таким чином, для визначення, чи є перешкода перед роботом, достатньо знати, наскільки відрізняються два сусідні кадри. Більш детальну інформацію дозволяє отримати оптичний потік.

Оптичний потік – це зображення видимого руху об'єктів, поверхонь або країв сцени, що отримується в результаті переміщення спостерігача (очей або камери) щодо сцени [10]. Розрахунок оптичного потоку виконується на основі двох припущень:

а) інтенсивність освітлення піксела на сусідніх кадрах є незмінною.

б) переміщення піксела є нескінченно малим. Виведена формула має наступний вигляд:

$$I_xV_x + I_yV_y = -I_t, \quad (2)$$

де $I_x$, $I_y$, $I_t$ – похідні від інтенсивності освітлення піксела з координатами x та y в момент часу t; $V_x$, Vy – швидкість (дискретне зміщення) піксела, невідомі у вказаному рівнянні.

Для виявлення оптичного потоку оптимальним за часом є алгоритм Лукаса-Канаде, який реалізовано у бібліотеці OpenCV [11].

Особливістю оптичного потоку є те, що чим більше зміщення відносно попереднього кадру, тим більшим є потік. Запропонований метод виявлення та уникнення перешкоди оснований на даній властивості.

Кадр з камери необхідно розділити вздовж на дві рівні половини. Для ключових точок зображення кожної половини розраховуються век-

тори оптичного потоку та розраховується сумарний вектор кожної половини. Якщо напрямки векторів вздовж осі абсцис співпадають, то однозначно необхідно рухатися в протилежному напрямі. Якщо ж напрямки протилежні, то необхідно рухатися в бік найменшого вектора до моменту, поки вектори не стануть однаковими. Даний підхід має два недоліки:

1. Якщо у послідовних кадрах на фоні є багато маленьких деталей, то разом вони забезпечать достатню величину для того, щоб робот вирішив повертати незалежно від наявності перешкоди.
2. Оцінка кадру розміром 640x480 пікселів займає приблизно 450 мс, за цей час робот може проїхати відстань S/2 та при наступній оцінці виявити перешкоду, вже зіткнувшись з нею.

Ці недоліки можна усунути модифікацією процедури обробки для зменшення впливу деталей фону та збільшенням швидкодії програми.

Розглянемо схематично кріплення телефона з камерою на платформі та відповідність зображення з камери (рис. 4).

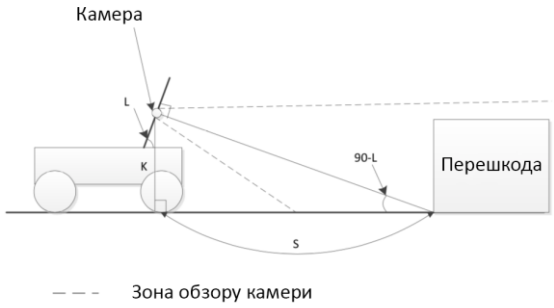


Рисунок 4 – Схематичне зображення видимої зони при закріпленні камери

Робот повинен реагувати на перешкоду на відстані S. Розрахунки виконуються з урахуванням припущення, що проекція вздовж перпендикуляру з камери – це проекція в центр кадру, а S – це відстань від камери до перешкоди. При закріплені планшету під кутом α на відстані S, перпендикуляр камери буде направлений у точку дотику перешкоди до площини поверхні землі. Кут α можна розрахувати за формулою:

$$\alpha = \arctan(S/K), \quad (3)$$

де S – це відстань від камери до перешкоди, К – це відстань від камери до землі.

На рис. 7, 8 та 9 наведено результати роботи програми.

За цими результатами можна зробити висновки щодо роботи алгоритму обходу перешкод. Система забезпечує необхідну функціональність, але при цьому коректно не відпрацьовуються крайні випадки – такі, як відсутність кутів у перешкоди, невеликий розмір перешкоди, висока текстурність поверхні підлоги та мала на перешкоді. Ці випадки не дозволяють отримати ключових точок перешкоди, а отже і не враховують її в оптичному потоці.

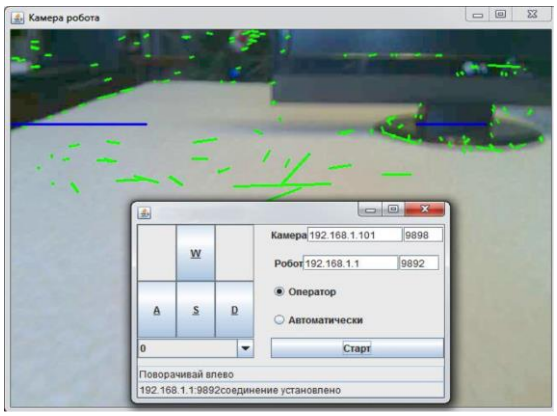


Рисунок 7– Результати роботи програми про виявленні перешкоди праворуч

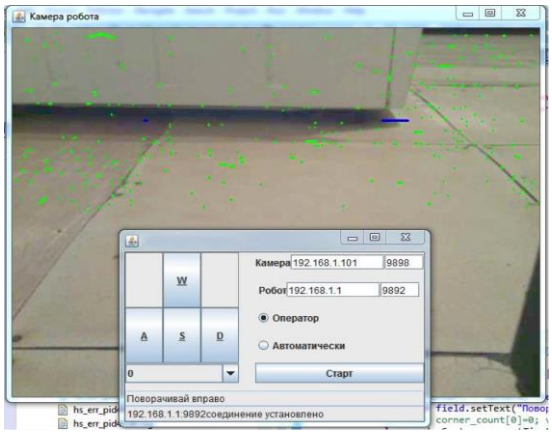


Рисунок 8 – Результати роботи програми про виявленні перешкоди ліворуч

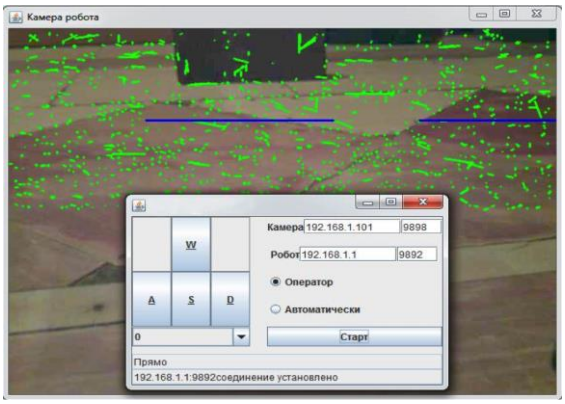


Рисунок 9 – Результати роботи програми на текстурній підлозі

З цього приводу можна зробити висновок, що система повністю працездатна лише в умовах, коли перешкоди є достатньо вираженими для формування ключових точок.

## Висновки

Таким чином, виконано розробку мобільного робота на базі обчислювальної платформи Arduino UNO та мобільного телефону з камерою. Під час проектування найбільша увага була приділена розробці алгоритму обходу перешкод на основі відеоданих.

Перевагами є застосування сучасних апаратних засобів та технологій, що знаходяться у широкому доступі, для реалізації робота, застосування кросплатформенної мови програмування, що дозволяє керувати мобільним роботом з будь-якої машини, а також алгоритм обходу перешкод на основі аналізу відеоданих з камери робота.

Головними недоліками запропонованого підходу є невизначена реакція системи на нечітковиражені перешкоди та в випадках, коли текстурність поверхні, по якій переміщується робот, більша за текстурність перешкоди, а також те, що система не передбачає реалізацію поведінки робота при втраті зв'язку з оператором. Перспективи подальшого розвитку рішення, що пропонується у роботі, стосуються модернізації програмного забезпечення оператора та в усуненні недоліків алгоритму для покращення реакції на перешкоди.

## Список використаної літератури

1. Ф. Жимарши. Сборка и программирование мобильных роботов в домашних условиях / Ф. Жимарши. - М.: НТ Пресс, 2007 – 288 с.
2. IP Webcam [електронний ресурс]. – Режим доступу: http://ip- webcam.appspot.com.
3. Апаратна платформа Arduino [електронний ресурс]. – Режим доступу: http://arduino.ru.
4. О. Бишоп. Настольная книга разработчика роботов / О. Бишоп. – К.: «МК Пресс» СПб «КОРОНА-ВЕК», 2010 –404 с.
5. Дж. Ловин. Создаем робота-андроида своими руками / Дж. Ловин. – М.: Издательский дом «ДМК-Пресс», 2007 – 312 с.
6. М. Предко. 123 Эксперимента по робототехнике / М. Предко. – М.: НТ Пресс. 2007 – 544 с.
7. ESP8266 [електронний ресурс]. – Режим доступу: http://esp8266.ru.
8. Motor Shield - Arduino motor [електронний ресурс]. – Режим доступу: http://www.ladyada.net/make/mshield/index.html.
9. С.С. Бехемір, Дж. Л. Баррон. The computation of optical flow / С.С. Бехемір, Дж. Л. Баррон // журнал ACM Computing Surveys (CSUR) – ч. 27 вер. 1995 – с. 433-466.
10. А. Пью. Техническое зрение роботов / А. Пью // М.: Машиностроение, 1987. — 320 с.
11. Optical Flow — OpenCV 3.0.0-dev documentation [електронний ресурс]. – Режим доступу: http://docs.opencv.org/trunk/doc/py_tutorials/py_video/py_lucas_kanade/py_lucas_kanade.html

*Надійшла до редакції 12.03.2015*

**S.O. TSOLOLO, D.Yu. BROVKINA, A.O. KRAVETSKIY**
Donetsk National Technical University

**DEVELOPMENT OF MOBILE ROBOT CONTROL SYSTEM WITH THE ABILITY TO IDENTIFY OBSTACLES FROM VIDEO DATA**

In this paper the dialog control system for a mobile robot, which was based on the computing platform Arduino UNO R3, was suggested. The developed mobile robot consists of moving platform, control module for engines, videocamera, Arduino Uno R3 as a main robot's system module, which implements commands from the user and sends commands to engine's control module. The developed software solves the problem of obstacle avoidance by analyzing video data coming from the robot's single camera. The software of developed system consists of client software, which provides user with control panel to move robot, receives videodata from the robot, analyzes it and gives the user tips, where to move robot to avoid collision with an obstacle, and mobile robot software, which receives commands from the user and implements them. To implement obstacle avoidance a modification of Lucas Kanade's algorithm was used. The main idea of algorithm is to use optical flow to identify obstacles on the background. The results of software testing showed that system and the algorithm work well, but only in conditions, when obstacles are well distinguishable from the background. The main advantage of the developed system is the use of modern hardware and technologies that are broadly available.

***Keywords: mobile robot, obstacle avoidance, optical flow.***