



Д. В. Стасенко, Д. В. Островка, В. М. Теслюк

Національний Університет "Львівська політехніка", м. Львів, Україна

РОЗРОБЛЕННЯ АВТОНОМНОЇ СИСТЕМИ КЕРУВАННЯ МОБІЛЬНОЮ РОБОТОТЕХНІЧНОЮ СИСТЕМОЮ З ВИКОРИСТАННЯМ МОДЕЛЕЙ НА ПІДСТАВІ ШТУЧНИХ НЕЙРОМЕРЕЖ

Проаналізовано основні літературні джерела щодо розроблення автономних систем керування на базі штучного інтелекту (ШІ). Після літературного огляду сформовано вимоги до розроблюваної мобільної робототехнічної системи з автономним керуванням та спроектовано її структуру. Розроблено середовище для навчання штучних нейромереж на підставі платформи *Unity*, з використанням об'єктно-орієнтовної мови розроблення – *C#*. Розроблено програмний інтерфейс для роботи зі штучною нейромережею. Обрано для розроблення мобільної робототехнічної системи з автономним керуванням такі принципи: ієрархічності та побудови інтелектуальної системи керування; модульності; змінного складу обладнання; зменшення габаритів, електроспоживання та вартості; автономного пересування в невідомому середовищі. Після аналізу методів та апаратно-програмних засобів функціонування спроектовано та розроблено роботизовану систему на підставі контролера *Raspberry Pi 2B*. Розроблено програмну та апаратну систему збирання даних з використанням інфрачервоних датчиків відстані. Розроблено програмні засоби для автономного керування мобільною системою. Для ефективності тренування моделі створено симульоване середовище з різними типами перешкод. Результатом цих досліджень є автономна мобільна робототехнічна система, що використовує розроблену штучну нейронну мережу прийняття рішень для руху в невідомій місцевості. Розроблена модель ШІ та система успішно пройшли тестування на симуляторі та в реальному середовищі. Розроблені системи придатні до покращень та можуть бути використані в інтересах промисловості чи малого бізнесу, а також здатні до адаптації відповідно до вимог користувачів (замовників).

Ключові слова: машинне навчання; автопілот; штучний інтелект; штучна нейронна мережа.

Вступ / Introduction

Останнім часом автономні машини щораз частіше потрапляють у заголовки новин, що майже перестало дивувати суспільство, яке прийняло таку технологію та активно почало її впроваджувати. Безпілотні автомобілі розглядають як заміну громадському транспорту та перспективу для перевезення вантажів. Для цього є всі підстави в епоху розвитку систем штучного інтелекту, які є запорукою існування автомобілів без водіїв.

Такі компанії, як *Waymo* та *Tesla*, інвестують значні кошти в розвиток безпілотних автомобілів. На сьогодні *Waymo* знову відновив тестування таких машин після зупинки в 2017 р. [5]. Тестування проводять з водіями, що перебувають у транспортних засобах, проте наразі компанія не зможе отримати достатньо даних, щоб перейти до повністю автономного рішення.

Об'єкт дослідження – процес розроблення автономного пересування робототехнічної системи в середовищі з перешкодами.

Предмет дослідження – моделі та засоби розроблення автономної системи керування мобільною робототехнічною системою з використанням штучних нейронних мереж.

Мета роботи – розробити систему керування автономною робототехнічною системою з використанням методів і засобів штучного інтелекту.

Для досягнення зазначеної мети визначено такі основні завдання дослідження:

- проаналізувати основні літературні джерела щодо розроблення автономних систем керування на базі штучного інтелекту (ШІ).
- розробити програмні засоби та алгоритм прийняття рішень на базі ШІ для руху мобільної робототехнічної системи в невідомому середовищі з перешкодами.

Наукова новизна отриманих результатів дослідження – розвинуто модель на підставі штучних нейронних мереж для задач розпізнавання перешкод у невідомій місцевості.

Практична значущість результатів дослідження –

Інформація про авторів:

Стасенко Данило Володимирович, магістрант, кафедра автоматизованих систем управління. Email: danylo.stasenko@gmail.com; <https://orcid.org/0000-0002-4636-0349>

Островка Дмитро Васильович, аспірант, кафедра автоматизованих систем управління. Email: ostrovdi@gmail.com; <https://orcid.org/0000-0003-4818-3822>

Теслюк Василь Миколайович, д-р техн. наук, професор, завідувач кафедри автоматизованих систем управління. Email: vasyli.m.teslyuk@lpnu.ua; <https://orcid.org/0000-0002-5974-9310>

Цитування за ДСТУ: Стасенко Д. В., Островка Д. В., Теслюк В. М. Розроблення автономної системи керування мобільною робототехнічною системою з використанням моделей на підставі штучних нейромереж. Науковий вісник НЛТУ України. 2021, т. 31, № 6. С. 112–117.

Citation APA: Stasenko, D. V., Ostrovka, D. V., & Teslyuk, V. M. (2021). Development of an autonomous control system for a mobile robotic system using artificial neural network models. *Scientific Bulletin of UNFU*, 31(6), 112–117. <https://doi.org/10.36930/40310617>

розроблено алгоритм прийняття рішення про напрям руху в невідомому середовищі, структуру роботи та засоби мобільної робототехнічної системи.

Аналіз останніх досліджень та публікацій. Ідеї інтелектуального керування мобільною робототехнічною системою стрімко стають популярними та активно розвиваються в таких напрямках, як побутова техніка, військова розвідувальна галузь, галузь інтелектуального транспортування [11, 15, 16]. Проте не всі сучасні рішення можна вважати саме автономним керуванням.

Так, у роботах [2, 4, 9, 19] зазначено, що переважна частина наявних мобільних робототехнічних систем потребує супроводу роботи оператором-людиною, задля швидкого корегування роботи системи. Це призводить до збільшення ймовірності помилкових дій та забезпечення постійного каналу зв'язку між системою та оператором.

З аналізу літератури [3, 7, 16, 17, 18] випливає, що основними шляхами вирішення невизначеності зовнішнього середовища, в якому рухається мобільна робототехнічна платформа, є набір інтелектуальних давачів (інфрачервоні, ультразвукові, лазерні) та засобів опрацювання вхідних даних на підставі штучних нейронних мереж.

Матеріали та методи дослідження. У роботі використано: методи навігації, методи навчання штучних нейронних мереж; сучасні компоненти мобільної робототехнічної платформи, елементну базу для реалізації апаратних комп'ютерних засобів; методи автономного опрацювання та оцінювання даних з давачів; методи та засоби автоматизованого проєктування апаратних і програмних засобів.

Результати дослідження та їх обговорення / Research results and their discussion

З огляду на стрімкий розвиток автономних автомобілів, великий попит на ринку та невеликі пропозиції метою цієї роботи є розроблення макету системи керування автономним роботом, з використанням методів і засобів ІІІ. Результатом цих досліджень є автономна мобільна робототехнічна система, для якої характерні такі вимоги:

- рух прямо з максимальною швидкістю, доки це можливо;
- вибір напрямку руху із врахуванням перешкод;
- рух назад, якщо інших варіантів немає;

Для досягнення цієї мети у роботі поставлено такі основні завдання:

- розроблення середовища для навчання нейромереж;
- тренування нейромережі та її нормалізація для інтегрування в програмну частину системи;
- розроблення програмного інтерфейсу для роботи з нейромережею;
- розроблення програмної та апаратної систем збирання даних;
- проєктування апаратної частини системи;
- створення програмних засобів для керування роботою систем.

Під час виконання поставлених завдань до апаратних частин розроблення систем належали: фізичні пристрої, що створюють середовище для функціонування системи, системи збирання даних, а також система керування переміщенням автономного робота. До програмних частин розроблення входили: керівний алгоритм системи (натренована модель за допомогою штучного інтелекту), а також різноманітні засоби програмного інтерфейсу.

1. Розроблення структури системи. У цій роботі розроблено структуру макету системи керування автономним роботом, яка містить давачі відстані, контролер оброблення даних, систему керування апаратними засобами та натреновану нейромережу (рис. 1).



Рис. 1. Загальна структура макету системи керування автономним роботом / General structure of the mock-up control system for an autonomous robot

У цій структурі модуль сенсорів відповідає за отримання даних про відстань до об'єкта. Основна програма або ж контролер опрацьовує дані та відправляє повідомлення між усіма модулями системи. Модуль нейромережі відповідає за опрацювання даних, отриманих з навколишнього середовища і прийняття рішення про наступні дії усієї системи. Система керування апаратними засобами відповідає за приведення в рух роботизованої системи.

Для функціонування системи розроблено алгоритм взаємодії між всіма модулями, який складається з таких кроків:

- Крок 1. Головний контролер опитує зовнішні сенсори, щоб зібрати інформацію про навколишнє середовище.
- Крок 2. Інформація надходить із сенсорів у головну програму і приводиться до необхідного формату.
- Крок 3. Контролер взаємодіє із нейромережею через програмний інтерфейс, надсилаючи нормалізовані дані.
- Крок 4. Нейромережа відповідає на запити контролера і на базі отриманих даних приймає рішення про траєкторію руху системи.
- Крок 5. Контролер опрацьовує результати нейромережі та відправляє відповідні сигнали керування на привід роботизованої системи.

2. Розроблення моделі на підставі штучної нейроної мережі. Під час виконання роботи розроблено модель на підставі штучної нейроної мережі. Як описано у структурній схемі, у системі повинен існувати модуль, який на підставі даних з навколишнього середовища повинен прийняти рішення про подальший рух робота. Для цього було використано тренування моделі з використанням методів "навчання з підкріпленням" [10].

Засобами розроблення моделі обрано 3D-середовище на платформі *Unity*. Цей програмний продукт уже реалізовує деякий потрібний функціонал, а також підтримує декілька мов програмування високого рівня і є безкоштовним у використанні [12]. Цей продукт має фреймворк для взаємодії із методами машинного навчання – *Unity Machine Learning Agents Toolkit (ML-*

Agents). Це проєкт з відкритим кодом, який дає змогу іграм та симуляціям бути середовищем для навчання інтелектуальних агентів. Він пропонує реалізації (на базі *PyTorch*) найсучасніших алгоритмів, які дають змогу розробникам і любителям ігор легко навчати інтелектуальних агентів для ігор 2D, 3D та особливо VR / AR, який стрімко набуває популярності [13]. Дослідники також можуть використовувати наданий простий у використанні *API Python* для навчання агентів за допомогою навчання з підкріпленням, імітаційного навчання, нейроревольюції або будь-якими іншими методами. Ці навчені агенти можуть бути використані для різних цілей, в т.ч. й контроль поведінки *NPC*, автоматичне тестування збірок ігор та оцінку різних рішень щодо дизайну ігор до випуску. Інструментарій *ML-Agents* є взаємовигідним як для розробників ігор, так і для дослідників ШІ, оскільки він забезпечує центральну платформу, де досягнення ШІ можна оцінити в багатому середовищі *Unity*, а потім зробити доступними для широких спільнот дослідників і розробників ігор.

Оскільки платформою для взаємодії із середовищем обрано *Unity*, то вибір мови програмування зроблено в бік *C#*. Мова програмування *C#* є суворо типізованою

мовою, краще піддається оптимізації, а також підтримується обраною платформою розроблення.

3. Вибір апаратних засобів для системи збирання даних. Для розв'язання поставленої задачі потрібні апаратні засоби для отримання відстані до об'єктів, а саме давачі відстані. Є дві групи давачів, які можуть бути використані в проєктованій системі: інфрачервоні та ультразвукові. У ході досліджень було обрано саме інфрачервоні давачі, оскільки інфрачервоний давач не має проблеми із відбиванням та має набагато менші розміри, але варто зазначити і вищу ціну [1]. При куті огляду $\sim 30^\circ$ достатньо 6 давачів для розробленої системи: 5 спереду і одна ззаду (рис. 2).

Середовищем для запуску було обрано *Raspberry Pi 2B+*. Це серія невеликих одноплатних комп'ютерів, розроблена у Великобританії Фондом *Raspberry Pi* спільно з *Broadcom*. Драйвер двигуна було обрано *L298N*. Це драйвер колекторних двигунів на 2 канали, який може також застосовуватися для керування одним кроковим двигуном. Максимальна напруга живлення моторів – 46 В, струм на канал – 2 А. Цей драйвер обрано для контролю приводів машинки: привід для повороту передніх коліс та привід задніх коліс.

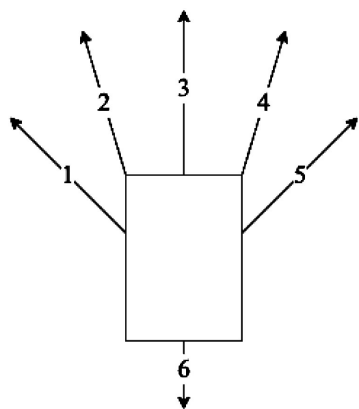


Рис. 2. Схематичне розміщення шести давачів у системі / Schematic layout of the 6 sensors in the system

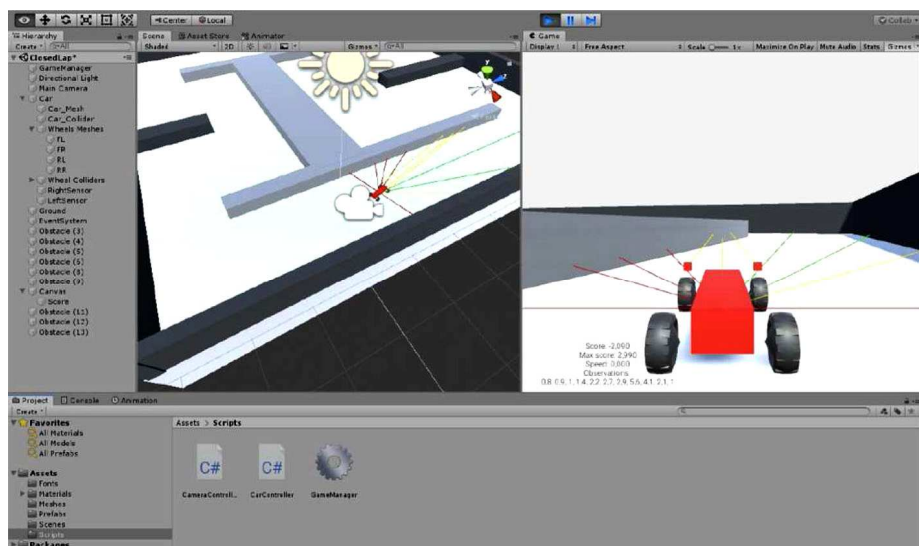


Рис. 3. Моделювання середовища в Unity / Environment modeling in Unity

Оскільки інфрачервоні давачі використовують протокол *I2C*, потрібно реалізувати його у проєктованій системі. Для вирішення проблеми підключення декількох *I2C* пристроїв з однаковими адресами до одного мікроконтролера обрано *I2C* мультиплексор (комутатор, розширювач). Цей пристрій дає змогу повністю контролювати процес доступу ведучого пристрою до ведених по *I2C* шині. Мультиплексор під'єднують по *I2C* шині до мікроконтролера (ведучого пристрою), а давачі та модулі (ведені пристрої) під'єднують до портів мультиплексора. Тому мультиплексор є посередником в обміні даними між ведучим і веденими пристроями.

4. Розроблення апаратно-програмних засобів. Наступним етапом проєктування мобільного робототехнічного засобу є моделювання тестового 3D-середовища у платформі *Unity*. Для цього за допомогою засобів платформи створюється навчальне поле та макет автономного роботизованого пристрою. Після чого запускається програмний код на *C#*, який регулює винагородження тренуваної моделі за прийняті рішення (напрямок руху роботизованої системи) на підставі даних з макетів

давачів відстаней до перешкод. Результати моделювання та вікна інтерфейсу програми зображено на рис. 3.

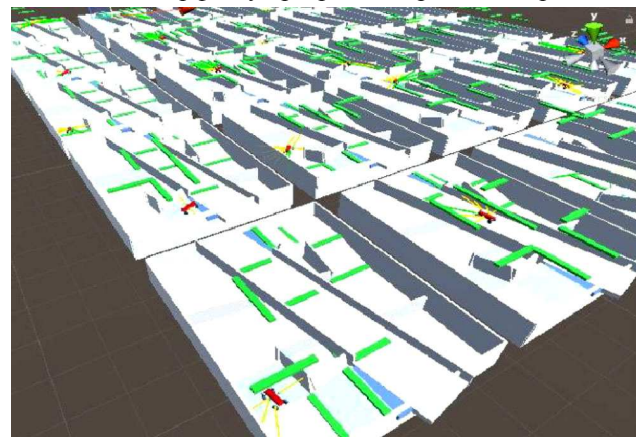


Рис. 4. Копії навчального середовища для пришвидшення навчання / Copies of the training environment to accelerate learning

Оскільки в реальних умовах роботизований пристрій може стикнутися з великою кількістю різних сце-

наріїв, тренування відбувається шляхом симуляції різних станів середовища (рис. 4). Після тренування – представлення нейромережі у *.nn* форматі. Нижче наведе-

```
public override void AgentAction (float[] vectorAction, string textAction) {
    float w = vectorAction[0]; float s = vectorAction[1]; float a = vectorAction[2]; float d = vectorAction[3];
    // Moves: Forward - Backward float motor = 0; if (w == 1 && s == 0) {
    motor = maxMotorTorque; score += FORWARD_REWARD; AddReward (FORWARD_REWARD);
    }
    if (w == 0 && s == 1) {
    motor = -maxMotorTorque;
    score -= FORWARD_REWARD * 2;
    AddReward (-FORWARD_REWARD * 2); }
    if (w == 1 && s == 1) {
    score -= 0.01 f;
    AddReward (-0.01 f); }
    // Moves: Turn left - right
    float steering = 0; if (a == 1 && d == 0) {
    steering = -maxSteeringAngle; }
    if (a == 0 && d == 1) {
    steering = maxSteeringAngle; }
    if (a == 1 && d == 1) {
    score -= 0.01 f;
    AddReward (-0.01 f); }
    foreach (WheelPair pair in wheelPairs) {
    if (pair.steering == true) { pair.leftWheel.steerAngle = pair.rightWheel.steerAngle =
    ((pair.reverseTurn)? -1: 1) * steering;
    }
    if (pair.motor == true) {
    pair.leftWheel.motorTorque = motor;
    pair.rightWheel.motorTorque = motor; }
    VisualizeWheel (pair); }
    detectObstacles ();
    measureSpeed (); float speedReward = speed * observations[4] * observations[5]; AddReward
    (speedReward);
    // logWithColor ("Speed reward: " + speedReward, "red"); String array = "Observations:
    \n" + string.Join (" ", observations) + "\n"; logWithColor (array, "red");
    }
```

5. Результати застосування розробленої системи на реальному прототипі. Запускаємо систему попередньо встановивши перешкоди на шляху та аналізуємо програмний звіт системи під час руху. Як бачимо на рис. 5, система звітує про відстань найближчого об'єкта зі шести інфрачервоних давачів, де від 1 до 6 – лівий, лівий передній, передній, передній правий та правий давачі відповідно. Після чого натренована модель нормалізує ці дані (рядок *In:*) та виводить прийняте рішення про наступну траєкторію руху рядку (*Out:*), де зліва направо – це значення в напрямку вперед, назад, вліво вправо. Відповідні до рис. 5, перша прийнята траєкторія, це назад і вправо, що відповідає вимогам до системи оминати поставлені перешкоди.

Обговорення результатів дослідження. Наявні системи [14] використовують *SLAM* алгоритми для автономного керування робототехнічною системою в невідомому середовищі. Використання такого підходу дає змогу створити карту місцевості, але і збільшує час для прийняття рішення про траєкторію руху мобільної системи. Розроблена в цій роботі нейромережа не потребує попереднього опрацювання всіх сторін системи перед початком її руху.

Для збільшення точності відстані до об'єктів у наявних системах [4, 14] використовують *LIDAR* давачі, що значно підвищує вартість роботизованої системи порівняно з розробленою системою на інфрачервоних давачах. Нагадаємо, що Лідар (транслітерація *LIDAR* англ. *Light Identification, Detection and Ranging*) – технологія отримання та оброблення інформації про віддалені об'єкти за допомогою активних оптичних систем, що використовують явища відбиття світла і його розсіювання в прозорих і напівпрозорих середовищах.

мання та оброблення інформації про віддалені об'єкти за допомогою активних оптичних систем, що використовують явища відбиття світла і його розсіювання в прозорих і напівпрозорих середовищах.

```
Automated system output:
Out: 0 1 0 1
1: 8190 mm, 819 cm, 56
2: 725 mm, 72 cm, 56
3: 95 mm, 9 cm, 56
4: 698 mm, 69 cm, 56
5: 337 mm, 32 cm, 56
6: 430 mm, 42 cm, 56
In: 1.0 0.362500003%5003395 0.1585 0.2185
Out: 0 1 0 1
1: 8199 mm, 819 cm, 55
2: 738 mm, 73 cm, 55
3: 96 mm, 9 cm, 55
4: 705 mm, 70 cm, 55
5: 294 mm, 29 cm, 55
6: 502 mm, 50 cm, 55
In: 1.0 0.368 0.0475 0.3525 0.147 0.251
Out: 0 1 0 1
1: 8190 mm, 819 cm, 56
2: 725 mm, 72 cm, 56
3: 96 mm, 9 cm, 56
4: 698 mm, 69 cm, 56
5: 317 mm, 31 cm, 56
6: 437 mm, 43 cm, 56
In: 1.0 0.3625 0.0475 0.349 0.1585 0.2185
Out: 0 1 0 1
1: 8190 mm, 819 cm, 57
2: 742 mm, 74 cm, 57
3: 96 mm, 9 cm, 57
4: 723 mm, 72 cm, 57
5: 320 mm, 32 cm, 57
6: 478 mm, 47 cm, 57
In: 1.0 0.371 0.048 0.3615 0.16 0.239
Out: 0 1 0 1
```

Рис. 5. Результати запуску програми на реальному прототипі / Results of starting the program on a real prototype

Поширеним рішенням навігації мобільної роботизованої системи на підставі штучних нейромереж є вико-

ристання набору фотографій потенційної місцевості, як вхідних даних для навчання [6]. Недоліком використання таких методів є необхідність створення та сортування великих масивів даних задля тренування моделей, що можуть приймати рішення в різних середовищах. Методи навчання, використані в цій роботі, дають змогу симулювати різноманітні середовища та масштабовано тренувати моделі без потреби створення набору вхідних даних.

Висновки / Conclusions

Розроблено структуру та алгоритми функціонування мобільної робототехнічної системи. Ця структура ґрунтується на модульному принципі та дає змогу швидко модернізувати пристрій. Розроблено модель на підставі штучної нейронної мережі з використанням методу "навчання з нагородою". Тренування моделі відбувалося в генерованому середовищі на платформі *Unity*. Після чого розроблено макет автоматизованої системи на базі *raspberry PI* з використанням створеної моделі. Наведено результати тестування проектованої системи в реальних умовах, під час яких було підтверджено виконання поставленої задачі, а саме рух роботизованої системи в середовищі з перешкодами. Перспективою подальших досліджень є розроблення моделі, що здатна швидше приймати рішення в замкненому просторі, або ж з більшою кількістю перешкод.

Підтримка дослідження. Роботу виконано за підтримки проекту № БФ/ІНФОТЕХ-2021 (Національний університет "Львівська політехніка").

References

- Adarsh, S., Kaleemuddin, S. M., Bose, D., & Ramachandran, K. I. (2016). Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/robot navigation applications. In *IOP Conference Series: Materials Science and Engineering*. Vol. 149, no. 1, 12141. IOP publishing.
- Buckley, Lisa, Sherrie-Anne Kaye, & Anuj K. Pradhan. (2018). A qualitative examination of drivers responses to partially automated vehicles. *Transportation research part F: traffic psychology and behaviour* 56, 167–175.
- Chen, X., Huang, F., Zhang, Y., Chen, Z., Liu, S., Nie, Y., Tang, J., & Zhu, S. (2020). A Novel Virtual-Structure Formation Control Design for Mobile Robots with Obstacle Avoidance. *Applied Sciences*, 10(17), 5807. <https://doi.org/10.3390/app10175807>
- Denysyuk, P., Teslyuk, V., & Chorna, I. (2018). Development of mobile robot using LIDAR technology based on Arduino controller, 2018 XIV-th International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH), 240–244. <https://doi.org/10.1109/MEMSTECH.2018.8365742>
- Dikmen, M., & Burns, C. M. (2016). Autonomous Driving in the Real World. *Proceedings of the 8th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. <https://doi.org/10.1145/3003715.3005465>
- Foroughi, Farzin, Zonghai, Chen, & Jikai, Wang. (2021). A CNN-Based System for Mobile Robot Navigation in Indoor Environments via Visual Localization with a Small Dataset. *World Electric Vehicle Journal*, 12(3), 134. <https://doi.org/10.3390/wevj12030134>
- Guanetti, J., Kim, Y., & Borrelli, F. (2018). Control of connected and automated vehicles: State of the art and future challenges. *Annual Reviews in Control*, 45, 18–40. <https://doi.org/10.1016/j.arcontrol.2018.04.011>
- Hrytsiuk, Yu. I., & Dalyavskyy, V. S. (2018). Using Petal Diagram for Visualizing the Results of Expert Evaluation of Software Quality. *Scientific Bulletin of UNFU*, 28(9), 97–106. <https://doi.org/10.15421/411832>
- Kakani, V., Kim, H., Kumbham, M., Park, D., Jin, C. B., & Nguyen, V. H. (2019). Feasible Self-Calibration of Larger Field-of-View (FOV) Camera Sensors for the Advanced Driver-Assistance System (ADAS). *Sensors*, 19(15), 3369. <https://doi.org/10.3390/s19153369>
- Kiran, B. R., Sobh, I., Talpaert, V., Mannion, P., Al Sallab, A. A., Yogamani, S., & Pérez, P. (2021). Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*.
- Matviichuk, K. V., Teslyuk, V. M., & Zelinskyy, A. Ya. (2016). Programming Model of Control Subsystem for Mobile Robotic Technical System. *Scientific Bulletin of UNFU*, 26(5), 325–333. <https://doi.org/10.15421/40260551>
- Meng, W., Hu, Y., Lin, J., Lin, F., & Teo, R. (2015). ROS+ unity: An efficient high-fidelity 3D multi-UAV navigation and control simulator in GPS-denied environments. In *IECON 2015-41st Annual Conference of the IEEE Industrial Electronics Society*, 2562–2567.
- Ostrovka, D. V., & Teslyuk, V. M. (2021). The current state and prospects of advanced reality technology in smart mobile devices. Selected aspects of digital society development: monograph 45. Katowice: Publishing House of University of Technology, 179–190.
- Qi, Xianyu, Wei, Wang, Ziwei, Liao, Xiaoyu, Zhang, Dongsheng, Yang, & Ran, Wei. (2020). Object Semantic Grid Mapping with 2D LiDAR and RGB-D Camera for Domestic Robot Navigation. *Applied Sciences*, 10(17), 5782. <https://doi.org/10.3390/app10175782>
- Tsmots, I. G., Teslyuk, V. M., & Vavruk, I. P. (2013). Hardware and software for controlling the movement of a mobile robotic system, in mater. 12th International Conference. *The Experience of Designing and Application of CAD Systems in Microelectronics*, CADSM2013, Lviv-Polyana, Ukraine, 368. [In Ukrainian].
- Tsmots, I. G., Teslyuk, V. M., Opotiak, Yu. V., Parcei, R. V., & Zinko, R. V. (2021). The basic architecture of mobile robotic platform with intelligent motion control system and data transmission protection. *Ukrainian Journal of Information Technology*, 3(2), 74–80. <https://doi.org/10.23939/ujit2021.02.074>
- Xuexi, Z., Guokun, L., Genping, F., Dongliang, X., & Shiliu, L. (2019). SLAM Algorithm Analysis of Mobile Robot Based on Lidar, 2019 Chinese Control Conference (CCC), 4739–4745. <https://doi.org/10.23919/ChiCC.2019.8866200>
- Yusof, Y., Mansor, H. M., & Ahmad, A. (2016). Formulation of a lightweight hybrid AI algorithm towards self-learning autonomous systems, 2016 IEEE Conference on Systems, Process and Control (ICSPC), 142–147. <https://doi.org/10.1109/SPC.2016.7920719>
- Zhang, Jian. (2021). AI based Algorithms of Path Planning, Navigation and Control for Mobile Ground Robots and UAVs. arXiv preprint arXiv:2110.00910

D. V. Stasenko, D. V. Ostrovka, V. M. Teslyuk

Lviv Polytechnic National University, Lviv, Ukraine

DEVELOPMENT OF AN AUTONOMOUS CONTROL SYSTEM FOR A MOBILE ROBOTIC SYSTEM USING ARTIFICIAL NEURAL NETWORK MODELS

Modern tools and software development methodologies have begun to be actively deployed in cars ranging from handheld computers to smartphones. Cars that can brake for an impediment on their own, signal the driver to move out of the lane, or make a turn on their own are just a few examples of these developments. All of this is the product of artificial intelligence (AI) technologies being

developed and optimized. The problem of completely autonomous cars, primarily for business solutions such as the transportation of products or people, has arisen as a result of such breakthroughs in consumer solutions. In this context, the goal of this research is to create a model that makes decisions using an artificial neural network. This research yielded an autonomous mobile robot system that moved across unknown terrain utilizing the proposed decision-making model. In order to achieve the goals of the research goals, neural network training methods were investigated and a reward learning method was chosen. The requirements for the produced system were formed and its structure was designed after conducting a literature review. The Unity Platform was chosen as the model learning environment, and the C# programming language was employed. For the creation of the mobile robotic system, the following concepts were chosen: hierarchy and intelligent control system construction; modularity; replaceable equipment. The robotic system was built using a Raspberry PI 2B controller and infrared distance sensors after the methodologies and hardware were analyzed. To adequately train the model, researchers generated a simulated environment with various sorts of disturbance. The created concept and technology have been successfully tested in both a simulator and a real-world setting. The developed solutions can be improved and utilized by industry or small businesses.

Keywords: machine learning; autopilot; artificial intelligence; artificial neural network.