

## РОЗРОБКА БЕЗПІЛОТНОГО РОБОТА НА БАЗІ ШТУЧНОГО ІНТЕЛЕКТУ

Розробка безпілотних роботів на основі штучного інтелекту (ШІ) є надзвичайно актуальною темою в сучасному світі, оскільки ШІ застосовується для створення різноманітних автономних систем, здатних виконувати завдання без прямого втручання людини. Одним з ключових аспектів таких систем є здатність робота до захоплення цілей за допомогою ШІ. Розробка безпілотного робота на базі ШІ для захоплення цілі включає кілька основних етапів. Спочатку вивчаються теоретичні основи ШІ та машинного навчання, зокрема алгоритми обробки зображень і розпізнавання об'єктів. Цей етап передбачає аналіз наукової літератури, патентів та технічної документації, щоб зрозуміти поточний стан справ у сфері безпілотних роботів і визначити найефективніші підходи до захоплення цілей. Ця стаття присвячена детальному опису розробки безпілотного робота, що здатний самостійно захоплювати цілі за допомогою ШІ. Після теоретичного аналізу розробляються алгоритми для виявлення та відстеження цілі, використовуючи згорткові нейронні мережі (CNN) та рекурентні нейронні мережі (RNN). Моделі тренуються на великих наборах даних зображень, що містять різноманітні варіанти цілей. ШІ безпілотного робота працюватиме на основі згорткових нейронних мереж (CNN) для розпізнавання об'єктів і рекурентних нейронних мереж (RNN) для відстеження цілей. Модель навчатиметься розпізнавати та захоплювати об'єкти за допомогою великих наборів даних, що включають зображення об'єктів з різних ракурсів та умов освітлення. Проаналізовано програмні інструменти, розроблено архітектуру системи навігації безпілотного робота, включаючи взаємодію сенсорів, алгоритмів планування та контролю.

**Ключові слова:** безпілотний робот, штучний інтелект, захват цілі, розпізнавання об'єктів, машинне навчання, глибинні сенсори, маніпулятор, траєкторія захвату, автономна робототехніка, інтеграція сенсорних даних.

### Вступ

Безпілотні роботи (БПР) стають все більш важливими в різних галузях, включаючи промисловість, медицину, військові технології та сільське господарство. Однією з ключових функцій безпілотних роботів є здатність до захвату і маніпуляції об'єктами. З використанням штучного інтелекту (ШІ) ця функція може бути значно покращена, надаючи роботам можливість автономно розпізнавати, захоплювати і переміщати цілі з високою точністю. У цій статті ми детально розглянемо процес розробки безпілотного робота, здатного до захвату цілі за допомогою ШІ.

### Аналіз проблеми та літературних джерел

Розробка безпілотних роботів на базі ШІ є активною сферою досліджень у сучасній робототехніці. Багато робіт присвячено створенню алгоритмів для автономного управління, обробки сенсорних даних і прийняття рішень в реальному часі. Зокрема, використання методів машинного навчання та глибинного навчання дозволяє значно підвищити ефективність роботи безпілотних систем.

Роботи [1] показали, що інтеграція нейронних мереж з традиційними алгоритмами управління дозволяє покращити точність і швидкість захвату об'єктів. Захват цілі за допомогою ШІ включає використання комп'ютерного зору та алгоритмів обробки зображень для ідентифікації та захвату об'єктів у складних умовах. Дослідження у цій галузі зосереджуються на розробці ефективних методів сегментації зображень, розпізнавання об'єктів і оцінки їхнього положення у просторі. Наприклад, робота [2] описує використання Kinect та ML для швидкого і точного виявлення 3d об'єктів у реальному часі, що є важливим для автономних роботів [1, 2].

Однією з ключових проблем у цій сфері є втрата сигналу та його вплив на точність захвату. Недостатня стійкість до перешкод і варіативність умов освітлення можуть призводити до помилок у роботі алгоритмів комп'ютерного зору.

Проте, наявні дослідження не завжди враховують усі аспекти інтеграції ШІ з робототехнічними системами. Більшість робіт зосереджені на окремих компонентах, таких як обробка зображень або управління рухом, але не розглядають проблему в комплексі. Це

створює прогалини у знаннях, особливо щодо взаємодії різних підсистем і їхньої інтеграції у єдину робочу систему.

**Мета роботи** дослідити методи та підходи до розробки безпілотного робота, що використовує штучний інтелект для ефективного виявлення та захоплення цілей.

#### **Методи дослідження**

Розробка безпілотного робота на базі ШІ для захоплення цілі включає кілька основних етапів. Спочатку вивчаються теоретичні основи ШІ та машинного навчання, зокрема алгоритми обробки зображень і розпізнавання об'єктів. Цей етап передбачає аналіз наукової літератури, патентів та технічної документації, щоб зрозуміти поточний стан справ у сфері безпілотних роботів і визначити найефективніші підходи до захоплення цілей.

Після теоретичного аналізу розробляються алгоритми для виявлення та відстеження цілі, використовуючи згорткові нейронні мережі (CNN) та рекурентні нейронні мережі (RNN). Моделі тренуються на великих наборах даних зображень, що містять різноманітні варіанти цілей [4].

#### **Результати досліджень**

Для створення безпілотного робота на базі ШІ, здатного захоплювати цілі, використовуються високоякісні камери для отримання зображень об'єктів, глибинні сенсори (LiDAR або стереокамери) для отримання тривимірної інформації про навколишнє середовище, та сенсори наближення для точного визначення положення об'єкта перед захватом. Обчислювальний модуль, наприклад, потужний процесор NVIDIA Jetson, обробляє дані в реальному часі. ШІ моделі забезпечують розпізнавання об'єктів та обчислення траєкторії захвату. Роботизована рука з кількома ступенями свободи та захватний механізм з можливістю регулювання сили стискання дозволяють гнучко та ефективно маніпулювати різними типами об'єктів.

Для розробки робота використовуються сучасні програмні інструменти та мови програмування. Python застосовується для реалізації алгоритмів машинного навчання та обробки даних. Бібліотеки, такі як NumPy та Pandas, використовуються для обробки сенсорних даних, а TensorFlow та Scikit-learn допомагають у створенні та навчанні моделей машинного навчання для розпізнавання та захвату цілей. C++ використовується для написання низькорівневого коду, що керує апаратними компонентами робота, такими як сенсори та актуатори, забезпечуючи високу продуктивність і ефективність, необхідні для швидкого та точного захвату цілей [3, 7].

#### *Сенсори і камери:*

- 1) камери високої роздільної здатності для отримання зображень об'єктів;
- 2) глибинні сенсори (наприклад, LiDAR або стереокамери) для отримання тривимірної інформації про навколишнє середовище [5, 6];
- 3) сенсори наближення для точного визначення положення об'єкта перед захватом.

#### *Обчислювальний модуль:*

- 1) потужний процесор (наприклад, NVIDIA Jetson) для обробки даних в реальному часі [7];
- 2) ШІ моделі для розпізнавання об'єктів і обчислення траєкторії захвату.

#### *Маніпулятори:*

- 1) роботизована рука з кількома ступенями свободи для гнучкого переміщення;
- 2) захватний механізм (грифер) з можливістю регулювання сили стискання для різних типів об'єктів.

### **Використання сучасних програмних інструментів та мов програмування для розробки робота**

Python є однією з основних мов програмування, що використовується для розробки безпілотного робота на базі ШІ. Вона забезпечує ефективну реалізацію алгоритмів машинного навчання та обробки даних. Зокрема, бібліотеки, такі як NumPy та Pandas, використовуються

для обробки та аналізу сенсорних даних, тоді як TensorFlow та Scikit-learn допомагають у створенні та навчанні моделей машинного навчання для розпізнавання та захвату цілей [3].

C++ використовується для написання низькорівневого коду, що керує апаратними компонентами робота, такими як сенсори та актуатори. Ця мова програмування забезпечує високу продуктивність і ефективність, що є критичним для обробки даних у реальному часі та виконання складних обчислень, необхідних для швидкого та точного захвату цілей [3].

OpenCV є бібліотекою комп'ютерного зору, яка використовується для обробки зображень та відео в реальному часі. Вона дозволяє реалізувати алгоритми розпізнавання об'єктів, оцінки їхнього положення та орієнтації у просторі. Це важливо для точного визначення та захвату цілей роботом.

ROS (Robot Operating System) використовується як фреймворк для розробки програмного забезпечення для робототехнічних систем. Він забезпечує інструменти для розробки, тестування та налагодження програм для безпілотних роботів. ROS також підтримує інтеграцію з різними мовами програмування та апаратними платформами, що робить його ідеальним для побудови комплексних систем на базі ШІ.

Gazebo є симулятором робототехнічних систем, який використовується для тестування та валідації алгоритмів захвату цілей у віртуальному середовищі. Це дозволяє перевіряти роботу алгоритмів та їхню взаємодію з різними об'єктами без необхідності використання реального апаратного забезпечення.

Таким чином, для розробки безпілотного робота на базі ШІ з функцією захвату цілі використовуються Python для алгоритмів машинного навчання, C++ для низькорівневого керування, OpenCV для комп'ютерного зору, ROS для інтеграції систем та Gazebo для симуляції та тестування.

Рисунок 1 ілюструє архітектуру системи навігації безпілотного робота, включаючи взаємодію сенсорів, алгоритмів планування та контролю.

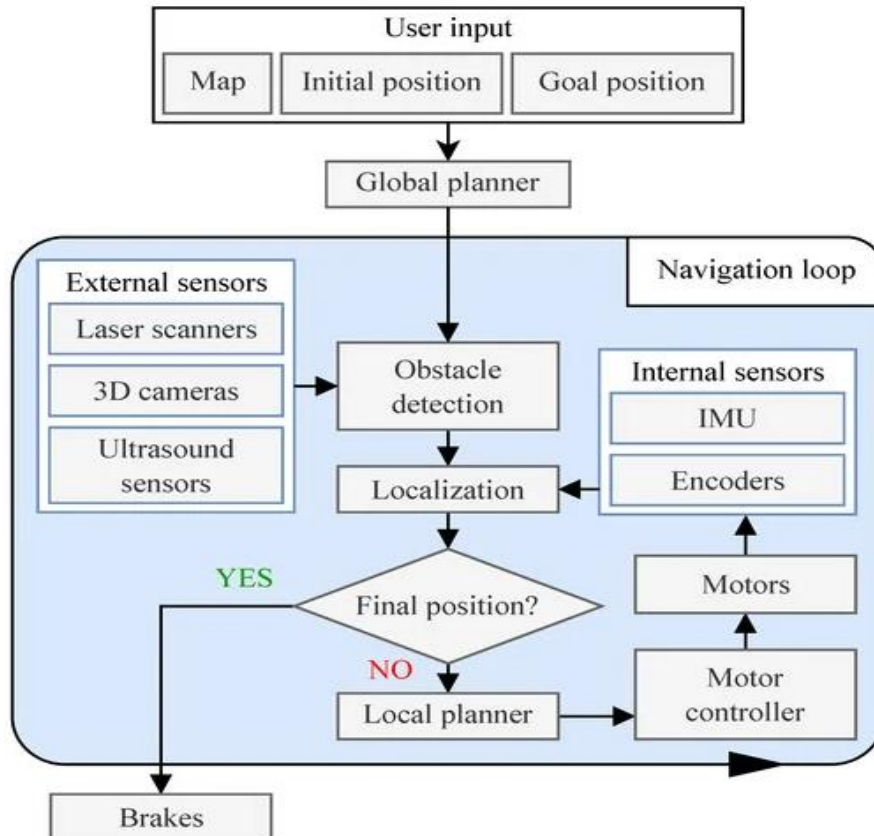


Рис. 1. Взаємодія сенсорів робота з навігацією

*Введення користувача*

- 1) користувач надає вхідні дані, включаючи карту, початкове положення та кінцеву точку;
- 2) дані використовуються глобальним планувальником для створення маршруту.

*Глобальний планувальник*

- 1) глобальний планувальник розраховує оптимальний шлях від початкової до кінцевої точки на основі введених даних;
- 2) цей план передається в навігаційний цикл для подальшої обробки.

*Зовнішні сенсори*

- 1) лазерні сканери, 3D-камери та ультразвукові сенсори збирають інформацію про навколишнє середовище;
- 2) ці дані використовуються для виявлення перешкод та локалізації робота.

*Внутрішні сенсори*

- 1) інерційні вимірювальні одиниці (IMU) та енкодери збирають інформацію про внутрішній стан робота, включаючи його швидкість та орієнтацію;
- 2) ця інформація допомагає в локалізації та контролі руху робота.

*Виявлення перешкод*

- 1) дані з зовнішніх сенсорів передаються в систему виявлення перешкод, яка аналізує їх для визначення потенційних загроз на шляху робота.

*Локалізація*

- 1) процес локалізації використовує дані з внутрішніх і зовнішніх сенсорів для визначення точного положення робота в просторі;
- 2) локалізація є ключовим етапом для точного слідування маршруту та уникнення перешкод.

*Планування локального маршруту*

- 1) якщо робот ще не досяг кінцевої точки, використовується локальний планувальник для обчислення найкращого маршруту до наступної проміжної точки, враховуючи виявлені перешкоди;
- 2) локальний планувальник оновлює траєкторію в реальному часі, щоб уникнути зіткнень.

*Кінцеве положення*

- 1) система перевіряє, чи досяг робот кінцевого положення. Якщо так, то робот зупиняється;
- 2) якщо ні, то навігаційний цикл продовжується.

*Контроль руху*

- 1) дані з локального планувальника передаються на контролер моторів, який керує рухом роботизованих моторів;
- 2) контролер моторів забезпечує точне виконання команди руху.

*Гальмування*

- 1) якщо робот досягає кінцевої точки або виявляється необхідність екстреної зупинки, то активується система гальмування.

Ця архітектура забезпечує автономну навігацію робота, дозволяючи йому орієнтуватися в складних середовищах, уникати перешкод та досягати заданих цілей з високою точністю.

**Розробка III моделі для захвату цілі**

Алгоритми обробки зображень є основою для багатьох застосувань у робототехніці, особливо в сфері автономних безпілотних роботів, що використовують штучний інтелект для захоплення цілей. До основних алгоритмів обробки зображень належать сегментація, розпізнавання об'єктів та оцінка їхнього положення у просторі. Сегментація зображень — це процес розділення зображення на кілька сегментів або областей для подальшого аналізу.

Використовуються такі методи, як порогове значення, кластеризація (наприклад, алгоритм k-середніх) та згорткові нейронні мережі (CNN). Розпізнавання об'єктів включає виявлення та класифікацію об'єктів на зображеннях. Сучасні методи включають використання глибоких нейронних мереж, таких як YOLO (You Only Look Once), R-CNN (Region-based CNN) та їхні варіанти. Наприклад, алгоритм YOLO виконує розпізнавання об'єктів у реальному часі, що є важливим для автономних систем. Оцінка положення та орієнтації об'єктів включає визначення тривимірного положення та орієнтації об'єктів у просторі. Використання стереозору, LiDAR та глибоких нейронних мереж дозволяє відновлювати 3D-інформацію з 2D-зображень. Ці алгоритми забезпечують точне розпізнавання та відстеження об'єктів у реальному часі, що є критичним для успішного захоплення цілей автономними роботами [6].

ШІ безпілотної робота працюватиме на основі згорткових нейронних мереж (CNN) для розпізнавання об'єктів і рекурентних нейронних мереж (RNN) для відстеження цілей. Модель навчатиметься розпізнавати та захоплювати об'єкти за допомогою великих наборів даних, що включають зображення об'єктів з різних ракурсів та умов освітлення. Навчання моделі здійснюватиметься протягом 100 000 ітерацій для досягнення високої точності розпізнавання та ефективності захвату цілей [4, 5].

### **1. Збір та підготовка даних**

Перед тим як навчати ШІ модель, необхідно зібрати великий набір даних, який включає зображення та тривимірні моделі об'єктів, які потрібно захоплювати. Дані мають включати різні ракурси, умови освітлення та варіації розташування об'єктів.

### **2. Навчання моделі розпізнавання об'єктів**

Для розпізнавання об'єктів можна використовувати глибокі нейронні мережі, такі як Convolutional Neural Networks (CNN) [4]. Процес включає кілька етапів:

- 1) анотація даних (маркування об'єктів на зображеннях для створення навчального набору);
- 2) навчання моделі (використання фреймворків, таких як TensorFlow або PyTorch, для навчання моделі на анотованих даних);
- 3) оцінка точності (перевірка моделі на тестових даних і налаштування гіпер параметрів для досягнення максимальної точності).

### **3. Інтеграція тривимірної інформації**

Щоб робот міг точно визначити положення об'єкта у просторі, необхідно інтегрувати дані з глибинних сенсорів. Це дозволить створити тривимірну карту навколишнього середовища і точно обчислити координати об'єкта.

### **4. Планування траєкторії захвату**

На основі даних з камер і глибинних сенсорів, ШІ модель обчислює оптимальну траєкторію для захвату об'єкта.

*Це включає:*

- 1) визначення найкращого підходу до об'єкта;
- 2) обчислення кутів повороту і переміщення маніпулятора;
- 3) урахування перешкод у навколишньому середовищі.

### **Тестування та валідація**

Після завершення розробки системи, необхідно провести ретельне тестування в реальних умовах:

- 1) тестування на різних типах об'єктів (перевірка здатності робота захоплювати об'єкти різних форм, розмірів і матеріалів);
- 2) оцінка продуктивності в різних умовах освітлення (перевірка роботи камери і алгоритмів розпізнавання об'єктів в умовах змінного освітлення);
- 3) тестування на стійкість до перешкод (перевірка здатності робота обминати перешкоди і коригувати траєкторію захвату).

## Висновки

Розробка безпілотного робота з функцією захвату цілі на базі ШІ є складним, але дуже перспективним завданням. Використання сучасних технологій машинного навчання і сенсорних систем дозволяє створювати роботів, здатних автономно розпізнавати і маніпулювати об'єктами з високою точністю. Така технологія має величезний потенціал для застосування в різних галузях, відкриваючи нові можливості для автоматизації та підвищення ефективності робочих процесів.

Програмування безпілотних роботів із застосуванням штучного інтелекту є однією з найбільш перспективних технологій майбутнього, оскільки дозволяє автоматизувати складні операції в різних сферах — від промислового виробництва до сільського господарства та охорони здоров'я. ШІ забезпечує роботам здатність до самонавчання, адаптації до нових умов та самостійного ухвалення рішень. Наприклад, у виробничих процесах безпілотні роботи можуть оптимізувати свої дії в режимі реального часу, аналізуючи величезні обсяги даних, що підвищує ефективність і знижує витрати. В поєднанні зі здатністю взаємодіяти з іншими машинами та системами через Інтернет речей (ІоТ), такі роботи мають потенціал кардинально змінити індустрію, роблячи їх більш автономними та стійкими.

## Перелік посилань

1. Жеребух, О., Фармага, І. Використання нейронних мереж для визначення об'єктів на зображенні. *Computer Design Systems. Theory And Practice*. Vol. 6, No. 1, 2024. p. 232-240. <https://science.lpnu.ua/sites/default/files/journal-paper/2024/apr/34361/42.pdf>
2. Коваль, О., Сарибога, Г. (2023). Система розпізнавання 3D об'єктів для безпілотних літальних апаратів на базі KINECT та ML. *Measuring And Computing Devices In Technological Processes*, (4), 74–81. <https://doi.org/10.31891/2219-9365-2023-76-9>
3. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, 2019, p. 1-400. [https://powerunit-ju.com/wp-content/uploads/2021/04/Aurelien-Geron-Hands-On-Machine-Learning-with-Scikit-Learn-Keras-and-Tensorflow\\_-\\_Concepts-Tools-and-Techniques-to-Build-Intelligent-Systems-OReilly-Media-2019.pdf](https://powerunit-ju.com/wp-content/uploads/2021/04/Aurelien-Geron-Hands-On-Machine-Learning-with-Scikit-Learn-Keras-and-Tensorflow_-_Concepts-Tools-and-Techniques-to-Build-Intelligent-Systems-OReilly-Media-2019.pdf)
4. Aggarwal, C. C. *Neural Networks and Deep Learning*: Springer, 2018. p. 120-300. [https://warin.ca/ressources/books/2018\\_Book\\_NeuralNetworksAndDeepLearning.pdf](https://warin.ca/ressources/books/2018_Book_NeuralNetworksAndDeepLearning.pdf)
5. Sutton, R. S., Barto A. G. *Reinforcement Learning: An Introduction*, 2nd Edition. 2018. p. 100-450. <http://incompleteideas.net/book/bookdraft2017nov5.pdf>
6. Krizhevsky, A., Sutskever, I., Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, 2019. [Електронний ресурс]. URL: <https://dl.acm.org/doi/10.1145/3065386> .
7. NVIDIA Corporation. *NVIDIA Jetson Platform for AI* [Електронний ресурс]. URL: <https://developer.nvidia.com/embedded-computing>

Надійшла 21.07.2024