

Гусак Юрій Аркадійович (доктор військових наук, професор)¹

Василенко Ольга Анатоліївна²

¹ Національний університет оборони України, Київ, Україна

² Центральний науково-дослідний інститут Збройних Сил України, Київ, Україна

АЛГОРИТМ СТВОРЕННЯ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ КЛАСИФІКАЦІЇ УДАРНИХ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ

Метою статті є розроблення алгоритму створення нейронної мережі для класифікації ударних безпілотних літальних апаратів за їхніми тактико-технічними характеристиками на основі елементів штучного інтелекту. Під час проведення дослідження застосовано основні принципи теорії штучного інтелекту, положення системного підходу, методи формальної логіки, аналізу та порівняння документів. Це поєднання дало змогу розробити алгоритм створення нейронної мережі для класифікації ударних безпілотних літальних апаратів. Запропоновано алгоритм створення нейронної мережі для класифікації ударних безпілотних літальних апаратів за їхніми тактико-технічними характеристиками. Описано підходи до проектування архітектури, методів навчання, підготовки даних для проведення навчання, тренування та тестування нейронної мережі для класифікації ударних безпілотних літальних апаратів. Продемонстровано, що процес створення та навчання нейронної мережі має шість етапів: формування бази даних; вибір архітектури нейронної мережі; навчання нейронної мережі; вибір алгоритму навчання; оцінювання результатів навчання; використання нейронної мережі. Розроблено структурно-функціональну схему нейронної мережі, що складається з вхідного, прихованого та вихідного шарів, кожен окремий нейрон описано відповідною активаційною функцією із підібраними ваговими коефіцієнтами. На основі аналізу різних типів та архітектур нейронних мереж обрано один із типів багатошарової нейронної мережі, а саме згорткову нейронну мережу (Convolutional neural network). Оскільки пакет MATLAB пропонує декілька інструментів для побудови нейронних мереж, то для класифікації ударних безпілотних літальних апаратів вибрано інструмент побудови архітектури нейронної мережі Pattern Recognition app, який формує двошарову нейронну мережу прямого зв'язку із прихованим і вихідним шарами та сигмоїдною функцією активації у прихованому шарі. Для навчання нейронної мережі, базу даних ударних безпілотних літальних апаратів поділено на тренувальний, перевірочний та тестовий набори, у співвідношенні 70% : 15% : 15% відповідно. Навчання мережі здійснюється за алгоритмом зворотного поширення Левенберга–Марквардта, який має найбільшу стійкість і високу швидкість збіжності. Результати оцінювання свідчать, що загальна точність нейронної мережі становить 98,9%, а значення похибки – 0,03868 вказує на високу продуктивність мережі. Наукова новизна полягає у розробленому алгоритмі створення нейронної мережі для класифікації ударних безпілотних літальних апаратів, який узагальнює типові етапи та процеси класифікації рою ударних безпілотних літальних апаратів на основі методів штучного інтелекту. Практичною і теоретичною значущістю викладеного у статті є можливість використання результатів дослідження для створення нейронної мережі залежно від складності й типу завдання, а також для класифікації ударних безпілотних літальних апаратів залежно від їх тактико-технічних характеристик. Напрямом подальшого дослідження може бути створення нейронних мереж для класифікації об'єктів групових цілей, а також – розв'язання задачі розподілу неоднорідного рою ударних безпілотних літальних апаратів по об'єктах нестационарної неоднорідної групової цілі.

Ключові слова: алгоритм створення нейронної мережі, ударні безпілотні літальні апарати, рій, нейронна мережа, класифікація безпілотних літальних апаратів, російсько-українська війна.

Вступ

Постановка проблеми. Під час російсько-української війни відбуваються зміни стратегії, форм та способів застосування військ. Однією з причин цього є розвиток озброєння та військової техніки, а особливо безпілотних систем, застосування яких стало масовим та дає змогу виконувати різноманітні завдання, спектр яких постійно розширюється. На цей час застосування

безпілотних систем є дієвим засобом ураження, що сприятиме переходу від позиційної оборони до маневрених дій.

Для ураження нестационарної неоднорідної групової цілі неоднорідним роєм ударних безпілотних літальних апаратів (далі – БпЛА) з максимальною ефективністю необхідно розв'язати оптимізаційну задачу розподілу неоднорідних ударних БпЛА по об'єктах нестационарної

неоднорідної групової цілі. Для цього потрібно здійснити класифікацію як ударних БпЛА, так і об'єктів групових цілей, а потім призначити кожному об'єкту того чи іншого класу групової цілі ударний БпЛА або декілька ударних БпЛА відповідного класу. Зазначене свідчить про актуальність обраної теми.

Одним із перспективних напрямів розв'язання задачі класифікації є застосування методів та алгоритмів штучного інтелекту, а саме нейронних мереж. Разом із тим, створення нейронних мереж під конкретну задачу являє собою складний процес, оскільки необхідно врахувати безліч параметрів цих задач.

Для цього є різні програмні засоби, наприклад, нейропакети SNNS (Stuttgart Neural Network Simulator), JavaNNS (Java Neural Network Simulator), Neural Lab, Neuron, Genesis, XNBC, Braincel, Neural Planner, NeuroPro, Partek Discovery, QwikNet, Statistica Automated Neural Networks, Thinks і ThinksPro тощо. Аналіз окреслених програмних інструментів, призначених для моделювання нейронних мереж, підтвердив, що вони вимагають досвіду розроблення програмних кодів різними мовами програмування, а також уможливають розв'язання вузького кола задач, тобто є вузьконаправленими.

Зважаючи на вищезазначене, варто розглянути програмний пакет для моделювання систем MATLAB, який містить чимало вбудованих бібліотек для створення нейронних мереж, що дають змогу розв'язувати задачі різних класів. Саме Neural Network Toolbox призначений для моделювання архітектур нейромереж.

Аналіз останніх досліджень і публікацій. З огляду на перспективність технологій штучного інтелекту [1–11], напряму розвитку Повітряних Сил Збройних Сил України [12] та розвитку безпілотної авіації [13–18], розроблення нейронних мереж для класифікації БпЛА з метою формування неоднорідних роїв ударних БпЛА в умовах ведення російсько-української війни є важливим завданням.

У [19] висвітлено підхід, який є основою застосування технологій великих даних і штучного інтелекту для підвищення обізнаності про ситуацію або комерційної цінності знань, отриманих із потоку даних від груп БпЛА. Запропоновано моделі оцінювання якості відеоданих із БпЛА, підходи до оснащення різномірних груп БпЛА та показники оцінювання їхніх тактико-технічних характеристик у групі за допомогою

програмного середовища MATLAB.

У [20] використано нейронну мережу до розроблення системи, що здатна навчатися розпізнаванню типу БпЛА за результатами аналізу трафіка, який передається ним на наземну станцію керування. Автором зауважено, що здатність нейронної мережі (далі – НМ) до узагальнення, за достатнього обсягу навчальної вибірки, дає змогу НМ екстраполювати власні знання про відомі види мережевих дій на невідомі види. За таких умов, вирішальну роль відіграє вибір адекватної архітектури НМ згідно поставлених завдань.

Приклади застосування нейронних мереж в задачах розпізнавання образів наведено у праці [21]. У [22–30] розглянуто застосування різних типів нейронних мереж для розпізнавання та класифікації об'єктів. У [31–34] викладено теоретичні матеріали щодо проєктування нейронних мереж та методи пошуку рішень у таких системах для різного класу задач. Разом із тим зауважимо, що в означених наукових працях, питання оптимальних шляхів вибору типів та архітектур нейронної мережі, а також алгоритмів їх створення, наприклад, із застосуванням відомих пакетів прикладних програм, висвітлені не достатньо.

Метою статті є розроблення алгоритму створення нейронної мережі для класифікації ударних безпілотної літальних апаратів за їхніми тактико-технічними характеристиками на основі положень теорії штучного інтелекту.

Виклад основного матеріалу дослідження

Завдання класифікації може бути вирішено із застосуванням методів та алгоритмів штучного інтелекту, а саме нейронних мереж. Створення та використання нейронних мереж для класифікації ударних БпЛА передбачає:

- формування бази даних БпЛА;
- вибір архітектури нейронної мережі;
- вибір алгоритму навчання нейронної мережі;
- навчання нейронної мережі;
- оцінювання результатів навчання нейронної мережі;
- використання нейронної мережі класифікації ударних БпЛА.

На рис. 1 зображено етапи створення та використання нейронної мережі для класифікації ударних БпЛА.

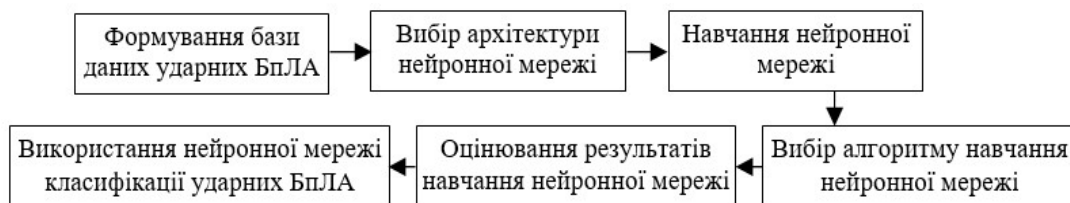


Рисунок 1 – Етапи створення та використання нейронної мережі для класифікації ударних безпілотної літальних апаратів

Розглянемо процес створення нейронної мережі для класифікації ударних БпЛА (далі – НМК УБпЛА).

1. *Формування бази даних ударних БпЛА.* Першим етапом створення НМК УБпЛА є формування бази даних ударних БпЛА для навчання нейронної мережі. На цьому етапі, насамперед, здійснюємо збір даних про ударні БпЛА, які застосовуються Збройними Силами України або є у перспективній розробці. Якість розробленої нейронної мережі буде залежати від ретельно підготовлених даних для її навчання. Навчальні дані мають охоплювати повний діапазон входних параметрів об'єкта, системи або процесу, для якого нейронна мережа буде використовуватись. Необхідно враховувати, що за великої розмірності входної множини даних ускладнюється процес абстрактного представлення даних. На основі аналізу тактико-технічних характеристик ударних БпЛА виділяємо ті характеристики, які дають можливість здійснити класифікацію саме ударних БпЛА. Такими характеристиками є маса бойової частини і тактичний радіус дії. Тоді у створюваній нейронній мережі має бути два входи. Отже, на етапі формування бази даних мають бути сформовані множини входних (Inputs) та вихідних даних (Targets).

2. *Вибір архітектури нейронної мережі.* Архітектура нейронних мереж являє собою структуру та організацію нейронів, а також зв'язки між ними. Щоб задати архітектуру нейронної мережі, необхідно вибрати кількість шарів у мережі, кількість нейронів у кожному шарі, типи зв'язків між нейронами (наприклад, повнозв'язні, зважені, рекурентні тощо), а також тип функції активації для кожного шару. Можна виділити три класи архітектур нейронних мереж.

Перший, одношарові нейронні мережі прямого поширення (*Feedforward Neural Network, FNN*). У найпростішому випадку в такій мережі є входний шар (*input layer*), інформація від якого передається на вихідний шар (*output layer*) нейронів, але не навпаки. Така мережа називається мережею прямого поширення (*feed forward*) (рис. 2).

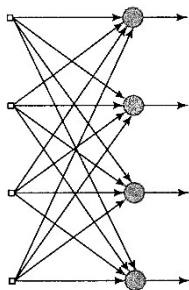


Рисунок 2 – Нейронна мережа прямого поширення з одним шаром нейронів

На рис. 2 зображено структуру такої мережі для випадку чотирьох нейронів у кожному з шарів (вхідному та вихідному). Така нейронна мережа називається одношаровою (*single-layer network*). Водночас, під єдиним шаром мається на увазі шар

нейронів.

Другий, багатошарові нейронні мережі прямого поширення (*Multilayer perceptron, MLP*). Нейронна мережа прямого поширення характеризується наявністю одного або кількох прихованих шарів (*hidden layer*), нейрони яких називаються прихованими нейронами (*hidden neuron*). Їхня функція проявляється у посередництві між зовнішнім входним сигналом і виходом нейронної мережі. Додаючи один або кілька прихованих шарів, ми збільшуємо порядок функції, що описує залежність вектора результатів функціонування нейронної мережі від вектора входних даних. Наприклад, для одношарової нейронної мережі ця функція буде першого порядку, для двошарової нейронної мережі – другого порядку і т. ін. Така мережа має додаткові синаптичні зв'язки та має більшу взаємодію між нейронами.

Нейрони входного шару мережі формують відповідні елементи шаблону активації (вхідний вектор), які становлять вхідний сигнал, що надходить на нейрони другого шару (тобто першого прихованого шару). Вихідні сигнали другого шару використовуються як вхідні для третього шару і т. ін. Зазвичай нейрони кожного із шарів мережі використовують як вхідні сигнали, вихідні сигнали нейронів попереднього шару. Набір вихідних сигналів нейронів вихідного (останнього) шару мережі визначає загальний відгук мережі на даний вхідний сигнал, сформований нейронами вхідного (першого) шару.

Мережа, яку показано на рис. 3, називається мережею 10-4-2, оскільки вона має 10 входних, 4 прихованих і 2 вихідних нейрона.

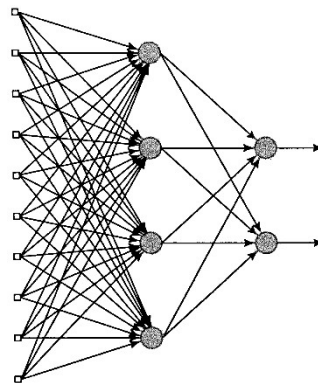


Рисунок 3 – Нейронна мережа прямого поширення 10-4-2

Нейронна мережа прямого поширення з m входами, h_1 нейронами першого прихованого шару, h_2 нейронами другого прихованого шару і q нейронами вихідного шару називається мережею $m-h_1-h_2-q$. Нейронна мережа, яку представлено на рис. 3, вважається повнозв'язною (*fully connected*) у тому сенсі, що всі нейрони кожного конкретного шару з'єднані з усіма нейронами суміжних шарів. Якщо деяких із синаптичних зв'язків немає, така мережа називається неповнозв'язною (*partially*

connected). Отже, багатошарові нейронні мережі прямого поширення складаються із вхідного шару, одного або декількох прихованих шарів (hidden layer) і одного вихідного шару (output layer) нейронів.

Вхідний сигнал поширюється нейронною мережею в одному напрямку, від шару до шару. Такі мережі зазвичай називають багатошаровими перцептронами. Отже, багатошарові нейронні мережі прямого поширення мають такі особливості:

кожен нейрон мережі має нелінійну функцію активації (*nonlinear activation function*). Важливо підкреслити, що ця нелінійна функція є гладкою (тобто такою, що диференціюється всюди), на відміну від порогової функції. Форма функції, що задовольняє цій умові, є сигмоїдальна (*sigmoidal nonlinearity*), яка визначається логістичною функцією (logistic function): $y_j = \frac{1}{1 + \exp(-v_j)}$, де

$$y_j = \frac{1}{1 + \exp(-v_j)}$$

v_j – індуковане локальне поле (тобто зважена сума всіх синаптичних входів плюс порогове значення) нейрона j ; y_j – вихід нейрона;

мережа містить один або декілька прихованих шарів нейронів. Ці нейрони дають змогу мережі навчатися розв'язанню складних завдань, послідовно враховуючи найважливіші ознаки із вхідного вектора;

мережа має високий ступінь зв'язності (*connectivity*), що реалізується за допомогою синаптичних зв'язків. Зміна ступеня зв'язності мережі потребує зміни множини синаптичних зв'язків або їхніх вагових коефіцієнтів.

Одним із типів багатошарової нейронної мережі є згорткові нейронні мережі (Convolutional neural network (далі – CNN)). Ці нейронні мережі являють собою основний інструмент для класифікації і розпізнавання об'єктів, осіб на фото, розпізнавання мови тощо. Також їх використовують для оброблення аудіосигналів, оброблення часових рядів, аналізу змісту текстів тощо. Є багато варіантів застосування CNN, зокрема Deep Convolutional Neural Network (DCNN), Region-CNN (R-CNN), Fully Convolutional Neural Networks (FCNN), Mask R-CNN та ін.

Третій, рекурентні нейронні мережі (Recurrent Neural Networks (далі – RNN)). Такі нейронні мережі відрізняються від мережі прямого поширення наявністю, принаймні, одного зворотного зв'язку (*feedback loop*). Наприклад, рекурентна мережа може складатися з єдиного шару нейронів, кожен з яких спрямовує свій вихідний сигнал на входи решти нейронів шару. Архітектуру нейронної мережі без прихованих нейронів і зворотних зв'язків нейронів із самими собою показано на рис. 4. У наведеної архітектурі немає зворотних зв'язків нейронів із самими собою, крім того, немає прихованих нейронів.

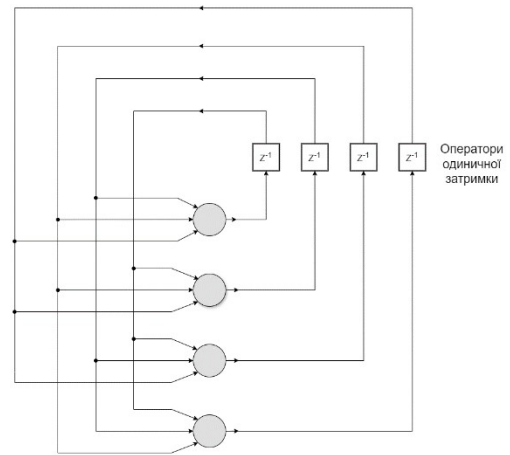


Рисунок 4 – Рекурентна мережа без прихованих нейронів і зворотних зв'язків нейронів із самими собою

На рис. 5 наведено рекурентну мережу із прихованими нейронами, в якій зворотні зв'язки виходять як із прихованих, так і з вихідних нейронів. Зворотний зв'язок означає використання елементів одиначної затримки (unit-delay element). Їх позначено як z^{-1} – оператори одиначної затримки.

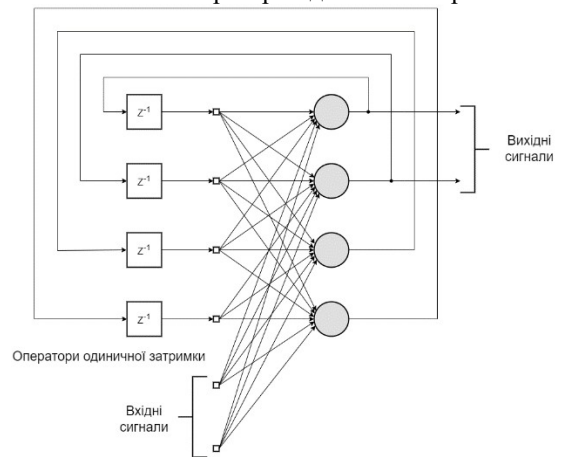


Рисунок 5 – Рекурентна мережа з прихованими нейронами

На відміну від прямих нейронних мереж, RNN можуть використовувати свій внутрішній стан (пам'ять) для оброблення послідовностей вхідних даних, що робить їх надзвичайно корисними для часових рядів або послідовних даних. Вони широко використовуються серед задач прогнозування.

Рекурентні нейронні мережі мають два види:

1. Довга короткочасна пам'ять (Long shortterm memory (LSTM)). Добре зв'язують попередню інформацію з поточною. Стан комірки пам'яті контролюється фільтрами, які дають змогу пропускати інформацію по нейронній мережі за певних умов: 0 – не пропускати інформацію, 1 – пропустити все.

2. Керовані рекурентні нейрони (Gated Recurrent Units (GRU)).

Пакет MATLAB пропонує декілька нейронних мереж для розв'язання чотирьох типів задач:

1. Pattern Recognition app – для класифікації образів.

2. Fitting App – для апроксимації.
 3. Clustering app – для кластеризації.
 4. Time Series app – для прогнозування.
- Наведемо їхню характеристику.

1. *Pattern Recognition app*. Двошарова нейронна мережа прямого зв'язку із прихованим та вихідним шарами (рис. 6) дуже добре вирішує завдання класифікації ударних БПЛА. При цьому функція активації у прихованому шарі має бути сигмоподібна (сигмоїда).

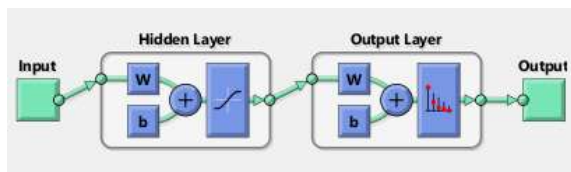


Рисунок 6 – Архітектура двошарової нейронної мережі прямого зв'язку

2. *Fitting App*. Двошарова нейронна мережа прямого зв'язку із сигмоподібним прихованим шаром та лінійною функцією активації вихідного шару (*fitnet*) розв'язує задачі багатовимірної апроксимації (рис. 7).

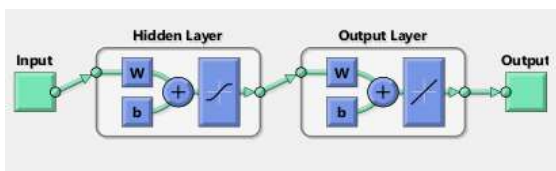


Рисунок 7 – Архітектура двошарової нейронної мережі прямого зв'язку

3. *Clustering app* (кластеризація). За допомогою самоорганізованої карти (*selforgmap, SOM*) можна класифікувати набір даних векторів з будь-якою розмірністю, на стільки класів, скільки нейронів містить шар (рис. 8). Нейрони впорядковані у двовимірній топології, що дозволяє шару формувати представлення розподілу та здійснювати двовимірну апроксимацію топології набору даних. Навчання нейронної мережі здійснюється за алгоритмом SOM (*trainbu, learnsomb*). Використовується для розв'язання задач кластеризації, дані групуються за подібністю.

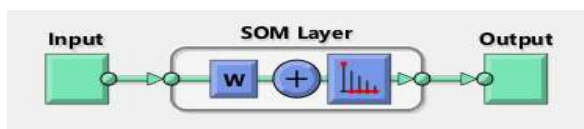


Рисунок 8 – Архітектура нейронної мережі із самоорганізованою картою

4. *Time Series app* (прогнозування часових рядів). Це різновид динамічного фільтрування, у якому попередні значення одного чи кількох часових рядів використовуються для прогнозування наступних (майбутніх) значень. Динамічні нейронні мережі, які включають лінії затримки з розгалуженнями, застосовуються для нелінійної фільтрації та прогнозування, а також для динамічного моделювання (для ідентифікації системи).

Динамічні моделі використовують для здійснення аналізу, моделювання, моніторингу та керування різними системами, зокрема робототехнікою та аерокосмічними системами. За допомогою цього інструменту можна розв'язувати три типи задач із нелінійними часовими рядами (рис. 9), а саме:

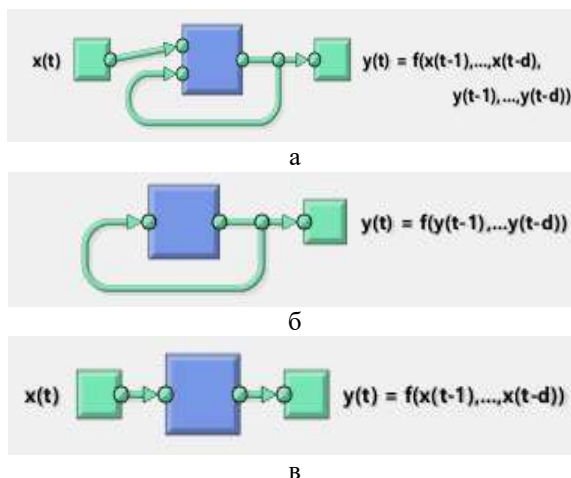


Рисунок 9 – Архітектура нейронних мереж:
 а) задача нелінійної авторегресії із зовнішнім (екзогенним) входом (NARX);
 б) задача нелінійної авторегресії (NAR);
 в) задача нелінійного перетворення.

Архітектура НМ визначається залежно від типу задачі. У нашому випадку виберемо додаток *Pattern Recognition app*, за допомогою якого створимо двошарову НМ для класифікації БПЛА. Для цього задаємо кількість нейронів, кількість входів та виходів та тип функцій активації для нейронів. Задача класифікації ударних БПЛА зводиться до групування вхідних даних відповідно до їхньої подібності (за тактико-технічними характеристиками ударних БПЛА).

На рис.10 показано розроблену архітектуру нейронної мережі класифікації ударних БПЛА. Така архітектура нейронної мережі класифікації ударних БПЛА обрана за результатами проведення тестових розрахунків з урахуванням різної кількості нейронів у прихованих шарах, різних функцій активації, а також точності та збіжності отриманих результатів. Нейронна мережа для класифікації ударних БПЛА має два шари нейронів: прихований та вихідний. Отже, маємо нейронну мережу з двома входами та трьома виходами. Кількість нейронів у прихованому шарі – 18.

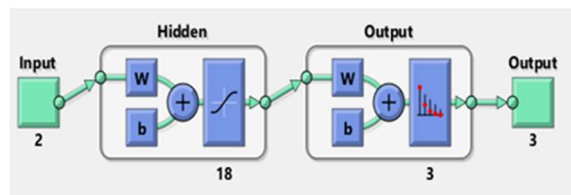


Рисунок 10 – Архітектура нейронної мережі класифікації ударних безпілотних літальних апаратів

Зазначимо, що багатошарові мережі володіють значно більшими можливостями, ніж одношарові. Проте багатошарові мережі можуть привести до збільшення обчислювальної потужності порівняно з одношаровими лише в тому разі, якщо функція активації між шарами буде нелінійною. Функції активації для прихованого і вихідного шарів можна вибрати залежно від характеристик вихідного шару. Це можуть бути результати, які описуються рівномірним, логарифмічним, експоненціальним розподілом тощо. Пакет MATLAB пропонує різні функції активації. Перелік найбільш поширених з них, вид функцій та їх математичні вирази наведено у табл. 1. У розробленій НМ використовуються сигмоподібні функції активації. У таблиці 1 такими функціями є квадратичний Z-сплайн і S-сплайн.

3. *Навчання нейронної мережі.* Для навчання НМ після формування бази даних (підготовки даних) набір даних поділяють на три набори: 1) тренувальний; 2) перевірочний; 3) тестовий. Під час розв'язання задачі класифікації ударних БПЛА було визначено, що тренувальний набір становить 70%

даних, набір для перевірки – 15% і тестовий набір – 15% (рис. 11).

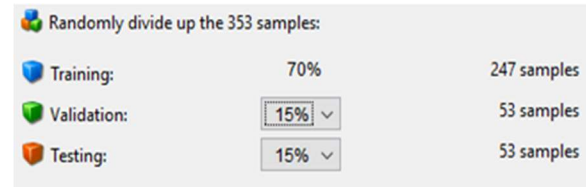


Рисунок 11 – Розподіл даних на тренувальний, перевірочний і тестовий набори

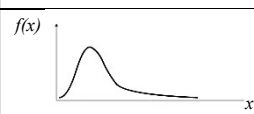
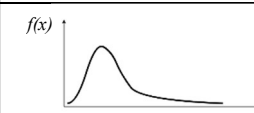
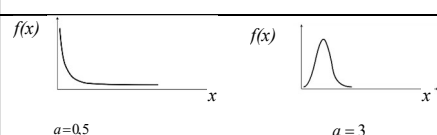
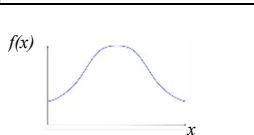
Слід зазначити, що відсоткову кількість даних можна обирати залежно від задачі та типів даних. Застосований метод поділу вхідних даних на три набори полягає у довільному виборі даних із набору підготовлених вхідних даних.

4. *Вибір алгоритму навчання нейронної мережі.* Процедура, яку використовують для навчання НМ, називається алгоритмом навчання (*learning algorithm*). Ця процедура вибудовує синаптичні ваги нейронної мережі, щоб забезпечити необхідну структуру взаємозв'язків між нейронами.

Таблиця 1

Вид та математичний вираз функцій активації

1.	Рівномірний розподіл	
	$f(x) = \begin{cases} \frac{1}{b-a}, & a \leq x \leq b \\ 0, & x < a, x > b \end{cases}$	
2.	Трикутний розподіл	
	$f(x) = \begin{cases} \frac{2(x-a)}{(m-a)(b-a)}, & a \leq x \leq m \\ \frac{2(b-x)}{(b-m)(b-a)}, & m \leq x \leq b \end{cases}$ $\mu = \frac{a+m+b}{3}$ $\sigma = \frac{\sqrt{a(a-m)+b(b-a)+m(m-b)}}{3\sqrt{2}}$	
3.	Квадратичний Z-сплайн	
	$f(x, a, b) = \begin{cases} 1, & x \leq a \\ 1 - \frac{2(x-a)^2}{(b-a)^2}, & a < x \leq \frac{a+b}{2} \\ \frac{2(x-a)^2}{(b-a)^2}, & \frac{a+b}{2} < x < b \\ 0, & x \geq b \end{cases}$	
4.	Квадратичний S-сплайн	
	$f(x, a, b) = \begin{cases} 0, & x \leq a \\ \frac{2(x-a)^2}{(b-a)^2}, & a < x \leq \frac{a+b}{2} \\ 1 - \frac{2(x-a)^2}{(b-a)^2}, & \frac{a+b}{2} < x \leq b \end{cases}$	
5.	Експоненціальний розподіл	
	$f(x) = \begin{cases} \lambda e^{-\lambda x}, & 0 \leq x < \infty \\ 0, & x < 0 \end{cases}$	
6.	Розподіл Гауса	
	$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, -\infty \leq x \leq +\infty$	

7.	Логнормальний розподіл	
	$f(x) = \begin{cases} \frac{1}{xb\sqrt{2\pi}} e^{-\frac{(\ln x - \mu)^2}{2b^2}}, & x \geq 0 \\ 0, & x \leq 0 \end{cases}$	$\mu = e^{\frac{a+b^2}{2}}$ $\sigma = \sqrt{e^{\frac{a+b^2}{2}} (e^{b^2} - 1)}$
8.	Розподіл Ерланга	
	$f(x) = \begin{cases} \lambda e^{-\lambda x}, & 0 \leq x \leq \infty \\ 0, & x < 0 \end{cases}$	$\mu = \kappa \frac{1}{\lambda}$ $\sigma = \sqrt{\kappa} \frac{1}{\lambda}$
9.	Розподіл Вейбула	
	$f(x) = \begin{cases} ab^{-a} x^{a-1} e^{-\frac{x^a}{b^a}}, & x \geq 0 \\ 0, & x \leq 0 \end{cases}$	$\mu = \frac{b}{a} \Gamma\left(\frac{1}{a}\right)$ $\sigma = \sqrt{\frac{1}{a} \left(2\Gamma\left(\frac{2}{a}\right) - \frac{1}{a} \left(\Gamma\left(\frac{1}{a}\right) \right)^2 \right)}$
10.	Колоколоподібна функція	
	$f(x, a, b, c) = \frac{1}{1 + \left \frac{x-c}{a} \right ^{2b}}$	a – коефіцієнт концентрації функції приналежності; b – коефіцієнт крутизни функції належності c – координата максимуму функції належності

Вибір алгоритму навчання нейронної мережі залежить від: обсягу даних, складності завдання, доступності обчислювальних ресурсів, а також особливостей даних та мережі. Наведемо деякі алгоритми навчання:

Метод зворотного поширення похибки. Це один із найпоширеніших алгоритмів навчання для нейронних мереж. Він використовує градієнтний спуск для змінення ваг мережі з метою мінімізації функції втрат.

Метод Adam. Цей алгоритм є комбінацією методу градієнтного спуску та ідей адаптивного вибору швидкості навчання. Добре працює для різних завдань та може забезпечити швидше навчання порівняно зі стандартним методом градієнтного спуску.

Метод RMSprop. Цей алгоритм також використовує градієнтний спуск, проте він адаптивно регулює швидкість навчання для кожного параметра в нейронній мережі окремо.

Метод моменту. Використовується для прискорення навчання, враховуючи попередні кроки градієнта. Це допомагає уникнути стагнації або в'язкості в градієнтному спуску.

Метод оптимізації Нестерова (Nesterov Accelerated Gradient). Цей метод також працює на основі ідеї моменту, але він використовує коригований градієнт для оновлення параметрів.

При виборі алгоритму важливо провести експерименти з різними методами та встановити, який працює найкраще для вашого конкретного завдання.

Пакет MATLAB пропонує різні функції для навчання та адаптації нейронних мереж: `adapt` – адаптація параметрів мережі; `adaptwb` – встановлення режиму адаптації; `train` – навчання мережі; `trainb` – групове навчання; `trainbfg` – алгоритм навчання Бройтона, Флетчера; `trainbr` – алгоритм навчання Бейєса; `trainc` – навчання із

циклічним представленням вхідних сигналів; `traingd` – метод градієнтного спуску; `trainr` – навчання у режимі випадкового представлення вхідних сигналів; `trainrp` – пороговий алгоритм зворотного поширення похибки; `trains` – адаптивне навчання із послідовним представленням вхідних сигналів; `trainscg` – алгоритм спряжених градієнтів (Scaled Conjugate Gradient, SCG).

Навчання мережі здійснюється:

за алгоритмом зворотного поширення Левенберга–Марквардта (`trainlm`). У випадках, коли не вистачає пам'яті, використовується масштабоване зворотне поширення спряженого градієнта (`trainscg`);

з учителем за алгоритмом зворотного поширення похибки (*Error back-propagation algorithm*). Цей алгоритм ґрунтується на корекції похибок (*error correction learning rule*). Його можна розглядати як узагальнення алгоритму адаптивної фільтрації – алгоритму мінімізації середньоквадратичної похибки (LMS);

методом зворотного поширення похибки передбачає два проходи всіма шарами мережі: прямий і зворотний.

Під час прямого проходу (*forward pass*) вхідний вектор подають на сенсорні вузли мережі, далі – поширюють мережею від шару до шару. У результаті генерується набір вихідних сигналів, який і є фактичною реакцією мережі на цей вхідний вектор. Крім того, всі синаптичні ваги мережі фіксовані. Під час зворотного проходу (*back-ward pass*) усі синаптичні ваги налаштовують відповідно до правила корекції похибок, а саме: фактичний вихід мережі віднімають від бажаного (цільового) відгуку, у результаті чого формується сигнал похибки (*error signal*). Цей сигнал згодом поширюється мережею в напрямку, зворотному напрямку синаптичних зв'язків. Звідси назва – алгоритм зворотного поширення похибки (або

алгоритм зворотного поширення – *back-propagation algorithm*). Синаптичні ваги налаштовуються з метою максимального наближення вихідного сигналу мережі до бажаного.

Процес навчання, реалізований за цим алгоритмом, називається навчанням на основі зворотного поширення (*back-propagation learning*). Алгоритм навчання методом зворотного поширення похибки складається з таких етапів:

1. Ініціалізація матриць ваг випадковим чином (у циклах).

2. Представлення нейронній мережі образу (на вхід подається значення з навчальної вибірки – вектор X) і береться відповідний вихід (вектор D).

3. Прямий прохід: обчислення в циклах виходів усіх шарів та отримання вихідних значень нейронної мережі (вектор Y).

$$y_i^k = f\left(\sum_{j=0}^{N_{k-1}} w_{ij}^k y_j^{k-1}\right)$$

$$y_j^0 = x_j, \quad y_0^{k-1} = 1, \quad y_0^k = 1, \quad x_0 = 1, \quad (1)$$

де y_i^k – вихід i -нейрона k -шару, f – функція активації, w_{ij}^k – синаптичний зв'язок між i -нейроном шару $k-1$, та i -нейроном шару k , x_j – вхідне значення.

4. Зворотний прохід: зміна вагів у циклах за формулами:

$$w_{ij}^k(t+1) = w_{ij}^k(t) + \eta \delta_i^k y_j^{k-1}, \quad (2)$$

$$\delta_i^k = (d_i - y_i) y_i (1 - y_i) - \text{для останнього (вихідного)}$$

шару, $\delta_i^k = y_i (1 - y_i) \sum_{l=k+1}^{N_{k+1}} \delta_l^{k+1} w_{il}^{k+1}$ – для проміжних

шарів, l – номер поточної ітерації циклу навчання (номер епохи), η – коефіцієнт навчання задається від

0 до 1, y_i^k – вихід i -нейрона k -шару, w_{ij}^k – синаптичний зв'язок між j -нейроном шару $k-1$, та i -нейроном шару k , d_i – бажане вихідне значення на

i -нейроні, y_i – реальне значення на i -нейроні вихідного шару.

5. Перевірка умови продовження навчання (обчислення значення похибки та/або перевірка заданої кількості ітерацій). Якщо навчання завершено, то переходимо до 2-го етапу, в іншому разі закінчуємо навчання. Ефективне значення похибки обчислюється таким чином:

$$\varepsilon = \frac{1}{Q} \sum_{q=1}^Q \sum_{i=1}^N (d_i - y_i)^2, \quad (3)$$

де Q – загальна кількість прикладів, N – кількість нейронів у вихідному шарі.

5. Оцінювання результатів навчання нейронної мережі. Після завершення навчання необхідно оцінити ефективність НМ на тестовому наборі даних. Це допоможе зрозуміти, наскільки добре НМ справляється із завданням. Якщо НМ не демонструє задовільних результатів, необхідно налаштувати такі параметри, як швидкість навчання, кількість шарів та нейронів, функції активації тощо.

Навчання нейронних мереж – ітеративний процес, який передбачає постійний її аналіз та вдосконалення. Для оцінювання результатів навчання мережі використовуються матриці похибок, гістограма похибок. Крім того, здійснюється перевірка, чи правильно обчислюється градієнт поблизу випадкової початкової точки. Матриця похибок показує відсоток правильних і неправильних класифікацій.

Правильною класифікацією є зелені квадрати на діагоналі матриць. Неправильні класифікації утворюють червоні квадрати. Якщо нейронна мережа навчилася правильно класифікувати, відсотки в червоних квадратах мають бути дуже малими, що вказує на незначну кількість неправильних класифікацій. Якщо це не так, то доцільним буде подальше навчання або навчання мережі з більшою кількістю нейронів у прихованому шарі нейронів. На рис. 12 наведено матрицю похибок.



Рисунок 12 – Матриці похибок під час навчання, перевірки і тестування нейронної мережі

На рис. 12 видно, що нейронну мережу було навчено з точністю 99,2%, перевірено і протестовано з точністю – 98,1%, а загальна точність нейронної мережі становить 98,9%. Під час розв’язання задачі оптимізації виконують обчислення перших похідних цільових і нелінійних функцій обмежень, а за допомогою функції `checkGradients` здійснюється перевірка запрограмованих похідних. Крім того, здійснюється перевірка, чи правильно обчислюється градієнт поблизу випадкової

початкової точки.

На рис. 13 показано результати оцінювання градієнта цільової функції та якобіанів. На рис. 14 наведено гістограму похибок між цільовими значеннями та прогнозованими значеннями після навчання нейронної мережі. Значення похибки 0,03868 вказує на найкращу продуктивність мережі. Оскільки похибки близькі до нуля, то нейронна мережа моделює добре. На рис. 15 чотири графіки наводять дані навчання, перевірки і тестування нейронної мережі.

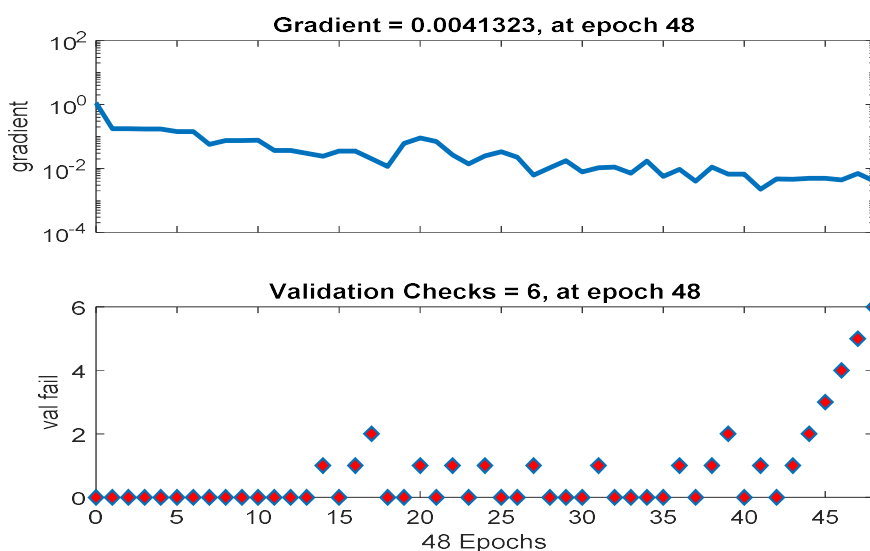


Рисунок 13 – Перевірка дійсності градієнтів або якобіанів

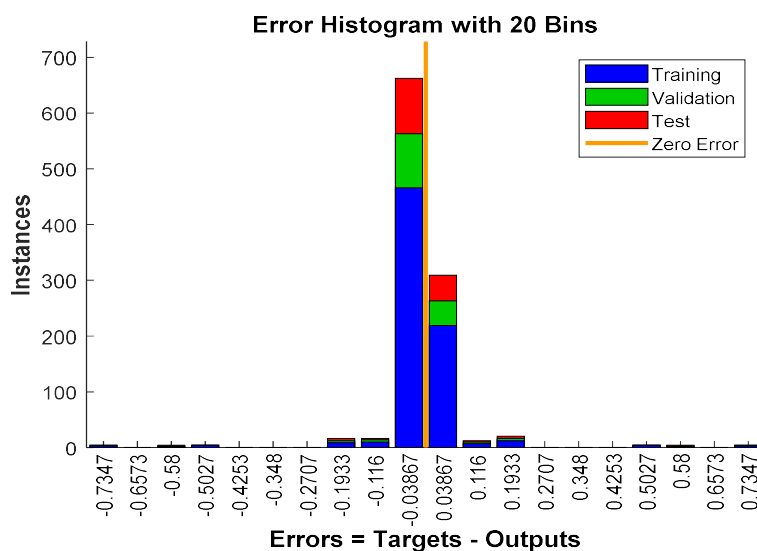


Рисунок 14 – Гістограма похибок

Пунктирна лінія на кожній діаграмі відображає оптимальний результат. Суцільна лінія – лінія лінійної регресії, яка найкраще підходить між результатами та цільовими показниками. Значення R є показником зв’язку між результатами та цілями. Якщо $R=1$, це означає, що є точний лінійний зв’язок

між результатами та цільовими показниками. Якщо R близьке до нуля, тоді немає лінійної залежності між виходами та цільовими показниками. Проаналізувавши графіки на рис. 15, з’ясуємо, що вони мають значення, ближчі до $R=1$.

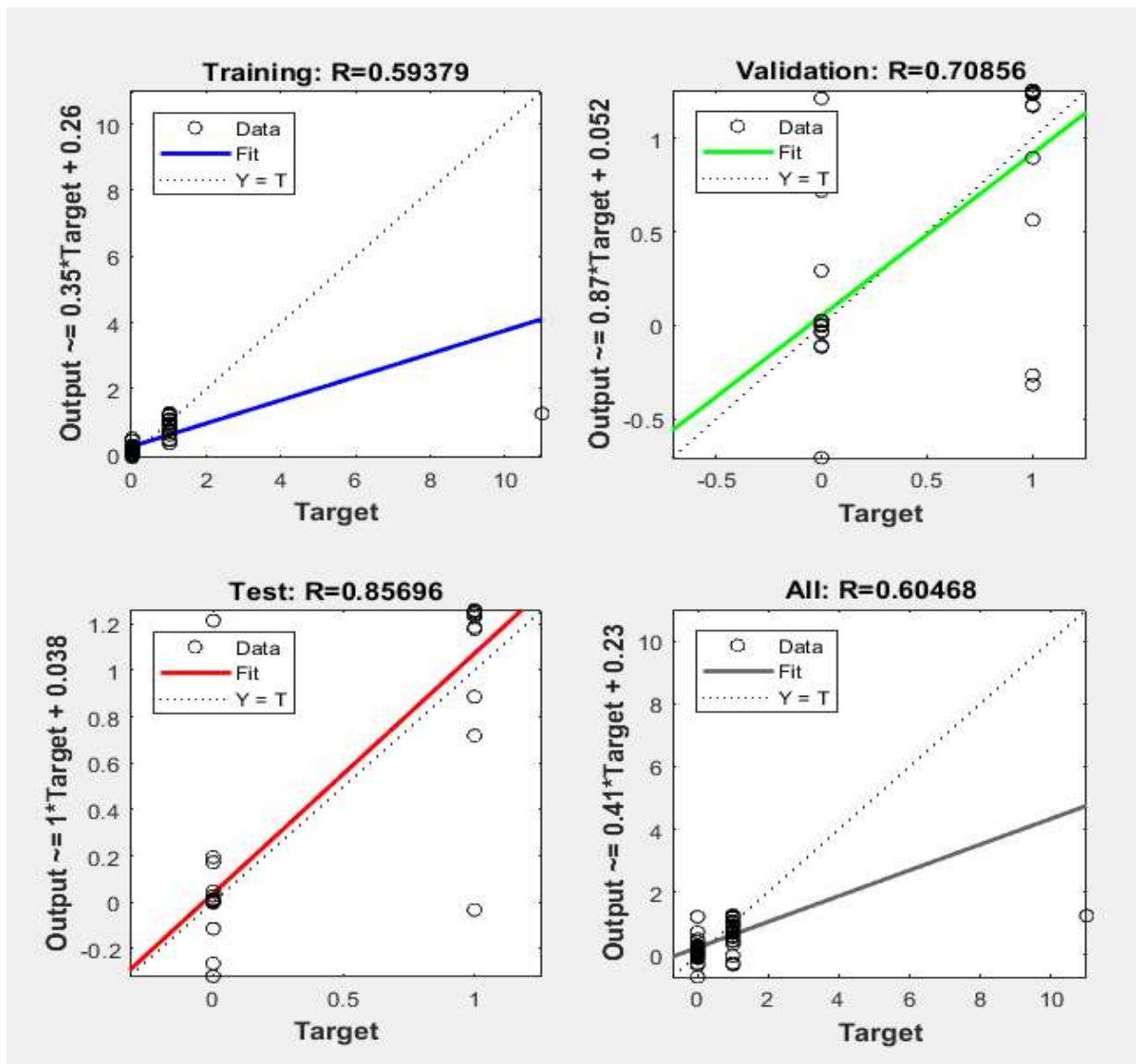


Рисунок 15 – Дані навчання, перевірки та тестування нейронної мережі

Графік зміни значення цільової функції за епохами – циклами навчання наведено на рис. 16.

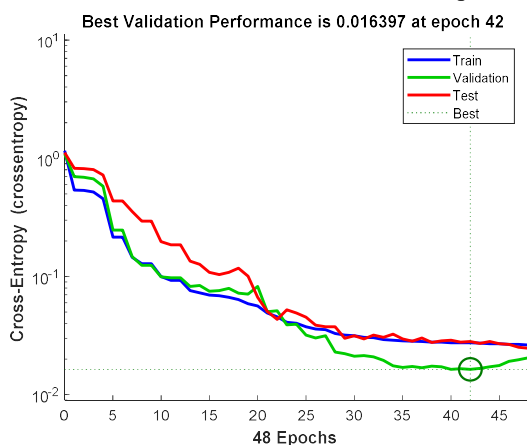


Рисунок 16 – Графік зміни значення цільової функції у процесі навчання

Серед параметрів навчання задано: goal – максимально припустиме значення цільової функції;

epochs – максимально припустима кількість циклів навчання мережі;

show – крок виведення на екран інформації про навчання мережі, задається у циклах навчання.

На графіку вказано ітерацію, на якій ефективність перевірки досягла мінімуму. На рис. 14 видно, що на 42-ій ітерації досягнуто локального мінімуму цільової функції. Отже, згорткову нейронну мережу може бути застосовано для класифікації ударних БПЛА з різними значеннями параметрів тактичного радіуса дії та маси бойової частини.

Алгоритм створення двошарової нейронної мережі для класифікації ударних БПЛА наведено на рис. 17.

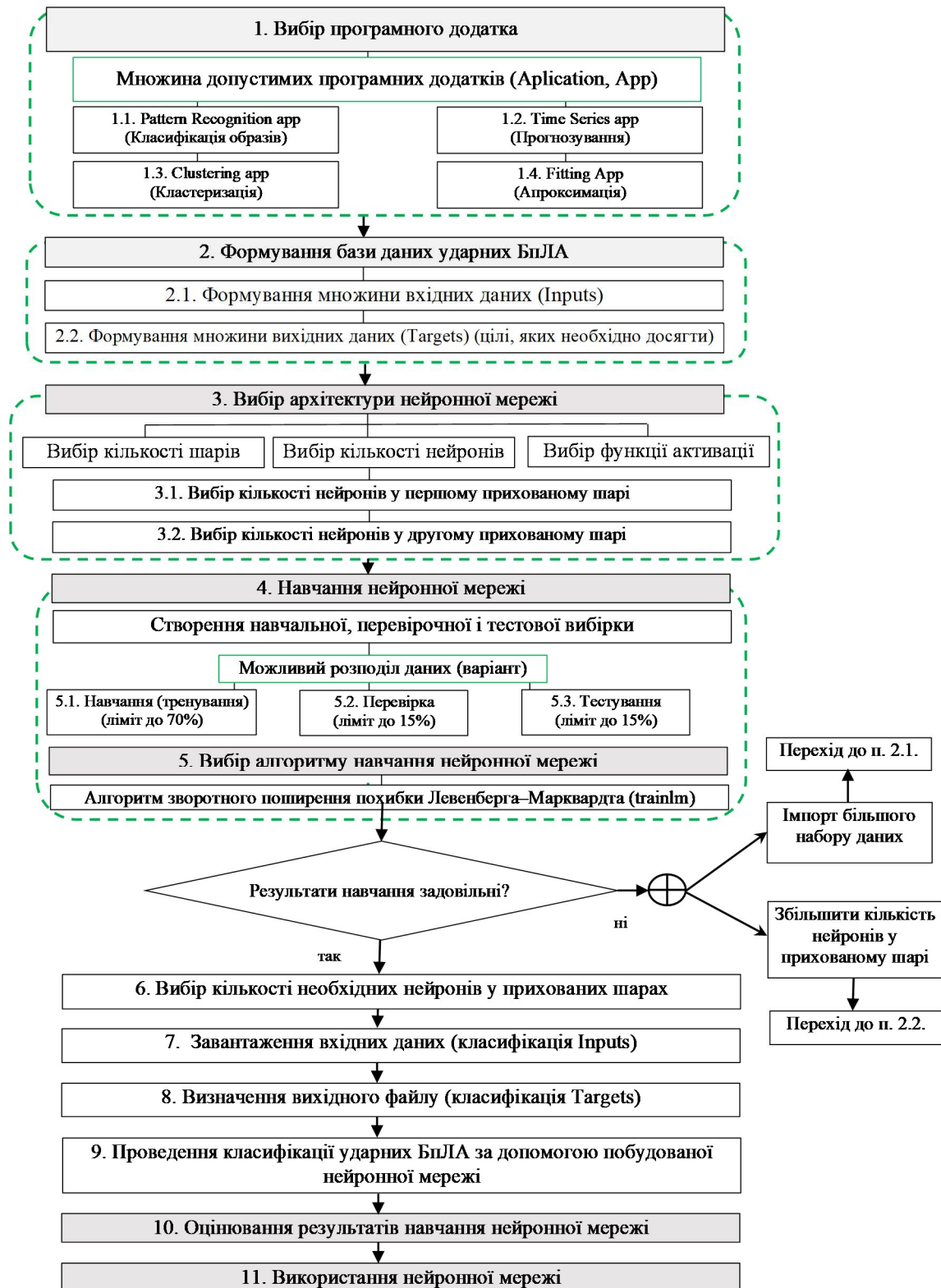


Рисунок 17 – Алгоритм створення нейронної мережі для класифікації ударних безпілотних літальних апаратів

Висновки й перспективи подальших досліджень

У роботі показано, що завдання класифікації ударних безпілотних літальних апаратів може бути вирішено за допомогою методів та алгоритмів

штучного інтелекту, а саме нейронних мереж. Створення та використання нейронних мереж для класифікації ударних безпілотних літальних апаратів передбачає:

формування бази даних безпілотних літальних апаратів;

вибір архітектури нейронної мережі;
 вибір алгоритму навчання нейронної мережі;
 навчання нейронної мережі;
 оцінювання результатів навчання нейронної мережі;
 використання нейронної мережі класифікації ударних БпЛА.

На етапі формування бази даних безпілотних літальних апаратів здійснюється збір даних про ударні зразки таких літальних апаратів, що застосовуються Збройними Силами України або є у перспективній розробці. На основі аналізу тактико-технічних характеристик ударних безпілотних літальних апаратів виокремлено ті характеристики, що дають змогу здійснити класифікацію саме ударних безпілотних літальних апаратів: маса бойової частини і тактичний радіус дії. Тоді у створюваній нейронній мережі має бути два входи. На цьому етапі формуємо множину вхідних даних (Inputs). Для визначення класів, до яких мають належати ударні безпілотні літальні апарати за їхніми основними тактико-технічними характеристиками (маса бойової частини, тактичний радіус), проводиться експертне опитування, за результатами якого, формується множина вихідних даних (Targets). Множини вхідних (Inputs) та вихідних (Targets) даних є навчальним набором даних для нейронної мережі класифікації ударних безпілотних літальних апаратів.

На другому етапі використано додаток MATLAB Pattern Recognition app для побудови архітектури нейронних мереж щодо класифікації ударних безпілотних літальних апаратів. Водночас визначено кількість нейронів, а також входів та виходів і тип функцій активації для нейронів. Крім того, визначено кількість шарів нейронів: прихований та вихідний.

Нейронні мережі для класифікації ударних безпілотних літальних апаратів має 2 входи та 3 виходи, а прихований шар має 18 нейронів.

На третьому етапі сформовано три набори даних: тренувальний; перевірочний; тестовий. Водночас визначено, що тренувальний набір становить 70% даних, набір для перевірки – 15% і тестовий набір – 15%. Відсоткову кількість даних можна вибрати залежно від задачі та типів даних.

Вибір алгоритму навчання нейронної мережі

Список бібліографічних посилань

1. Глибовець М. М., Олеський О. В. Системи штучного інтелекту. Київ : КМ Академія, 2002. 366 с. 2. Погудіна О. К. та ін. Методологія формування інтелектуальної складової агентної системи рою безпілотних літальних апаратів. Моногр. Харків : НАУ ім. М. С. Жуковського «ХАІ», 2021. 219 с. 3. Аврунін О. Г., Владов С. І., Петченко М. В., Семенець В. В., Татарінов В. В., Тельнова Г. В., Філатов В. О., Шмельов Ю. М., Шушляпіна Н. О. Інтелектуальні системи автоматизації. Моногр. Кременчук :

зроблено з урахуванням обсягу даних, складності завдання, доступності обчислювальних ресурсів, а також особливостей даних та мережі. Для навчання нейронної мережі обрано алгоритм зворотного поширення Левенберга–Марквардта (trainlm).

Оцінювання результатів навчання здійснено на тестовому наборі даних. Для оцінювання результатів навчання мереж використано матриці похибок та гістограму похибок. Крім того, обчислюється градієнт поблизу випадкової початкової точки.

Нейронну мережу було навчено з точністю 99,2%, перевірено і протестовано з точністю – 98,1%, а загальна точність нейронної мережі становить 98,9%. Гістограма похибок між цільовими значеннями та прогнозованими значеннями після навчання нейронної мережі показала, що значення похибки становить 0,03868, що вказує на найкращу продуктивність мережі. Оскільки похибки близькі до нуля, то нейронна мережа моделює добре. Чотири графіки навчання, перевірки та тестування нейронної мережі демонструють, що значення показника зв'язку між результатами та цільями R максимально наближено до R=1. За таких умов, на 42-ій ітерації досягнуто локального мінімуму цільової функції. Отже, згортована нейронна мережа може бути застосована для класифікації ударних безпілотних літальних апаратів з різними значеннями параметрів тактичного радіуса та маси бойової частини. Подібні алгоритми можуть бути застосовані для класифікації будь-яких типів безпілотних бойових систем.

Наступним етапом досліджень може бути розроблення та інтеграція програмно-апаратної системи на основі розпаралелених та частково розпаралелених засобів обчислювальної техніки, що дасть змогу значно пришвидшити обчислювальні операції, досягти виконання процесів навчання та тренування нейронної мережі в режимі реального часу та без втрати точності. Напрямом подальшого дослідження може бути створення нейронних мереж для класифікації об'єктів групових цілей, а також для розв'язання задачі розподілу неоднорідного рою ударних безпілотних літальних апаратів по об'єктах нестационарної неоднорідної групової цілі. Матеріали, які наведено у статті, можуть бути корисними науковим та науково-педагогічним працівникам у галузі математичного моделювання складних систем воєнного призначення, а також здобувачам вищої освіти (курсантам, студентам, слухачам та ад'юнктам), які досліджують зазначені питання.

Вид-во «НОВАБУК», 2021. 322 с. 4. **Методи** та системи штучного інтелекту : навч. посіб. для студентів напряму підготовки 6.050101 «Комп'ютерні науки». Уклад. А. С. Савченко, О. О. Синельников. Київ : НАУ, 2017. 190 с. 5. **Порохова О. Є.** Сутність і проблематика штучного інтелекту в управлінні проектами. Одеса : Одеський національний університет імені І. І. Мечникова, 2020. URL: <https://dspace.onua.edu.ua/search?spc.page=1&query=Сутність%20і%20проблематика%20штучного>

%20інтелекту (дата звернення: 09.06.2024).

6. Нікітіна Л. О., Нікітін С. О. Моделі та методи штучного інтелекту у комп'ютерних іграх. Харків: «Друкарня Мадрид», 2018. 102 с.

7. Методи та системи штучного інтелекту: навч. пос. / Уклад. Д. В. Лубко, С. В. Шаров. Мелітополь : ФОП Однорог Т. В., 2019. 264 с.

8. Шевченко А. І., Агарков А. В., Азаренко Д. С., Герасімов І. Г., Дорохіна Г. В., Іванова С. Б., Ніценко А. В., Шелепов В. Ю. Проблеми штучного інтелекту. Аналіз та синтез комунікаційної інформації : моногр. ІПШ МОН України і НАН України. Донецьк : ІПШ «Наука і освіта», 2014. 212 с.

9. Методи та системи штучного інтелекту: навч. пос. для студентів спеціальності 122 «Комп'ютерні науки». Уклад. І. М. Удовик, Г. М. Коротенко, Л. М. Коротенко, В. О. Трусов, А. Т. Харь. Дніпро : Держ. ВНЗ «Національний гірничий університет», 2017. 105 с.

10. Руденко О. Г., Бодяньський Є. В. Штучні нейронні мережі : навч. посіб. Харків : ТОВ «Компанія СМІТ», 2006. 404 с.

11. Субботін С. О. Нейронні мережі: теорія та практика : навч. посіб. Житомир : Вид. О. О. Євенок, 2020. 184 с.

12. Стратегія Повітряних Сил 2035. Вінниця : КПС ЗС України, 2020. 40 с.

13. Zhikai Y. The development and application of UAV intelligent machine learning system based on artificial intelligence. *Advances in Education, Humanities and Social Science Research*. 2023. № 6. P. 140-146. DOI: 10.56028/aehtsr.6.1.140.2023.

14. Mrad I., Samara L., Abdellatif A., Al-Abbasi A., Hamila R., A. Erbad. Federated Learning for UAV Swarms Under Class Imbalance and Power Consumption Constraints. *2021 IEEE Global Communications Conference (GLOBECOM)*. Madrid, Spain, 2021. P. 01–06. DOI: 10.1109/GLOBECOM46510.2021.9685143.

15. Харченко О. В., Артюшин Л. М., Кононов О. А. Перспективи реалізації спільного застосування безпілотних літальних апаратів. *Збірник наукових праць Державного науково-дослідного інституту авіації*. 2022. Вип. 18 (25). С. 7–13. DOI: <https://doi.org/10.54858/dndia.2022-18-1>.

16. Артюшин Л. М., Кононов О. А., Герасименко В. В., Наусенко Б. Ю. Метод вибору варіанта реалізації групового застосування безпілотних літальних апаратів. *Сучасні інформаційні технології у сфері безпеки та оборони*. 2022. № 1 (43). С. 48–59. DOI: <https://doi.org/10.33099/2311-7249/2022-44-2-10-20>.

17. Шовкошитний І. І., Василенко О. А. Проблемні питання ройового застосування ударних безпілотних літальних апаратів. *Сучасні інформаційні технології у сфері безпеки та оборони*. № 3(48). С. 27–34. DOI:10.33099/2311-7249/2023-48-3-27-34.

18. Компанієць О. М., Ключніков І. М., Дмитрієв А. Г. Комплексний аналіз впливу факторів на ефективність управління роями безпілотних літальних апаратів. *Системи озброєння і військова техніка*. 2023. № 3 (75). С. 66–70.

19. Korolyov V., Ogurtsov M., Khodzinsky A. Statement of the problem of complete set of UAV group on the basis of models of granular calculations and fuzzy logic. *Cybernetics and Computer Technologies*. 2021. № 2. P. 25–38.

20. Івашенко А. О. Інформаційне та програмне

забезпечення системи ідентифікації безпілотних літальних апаратів. Кваліф. робота бакалавра. Суми : Сумський держ. ун-т, 2022. 47 с.

21. Оганезов А. Л. Применение нейронных сетей в задачах распознавания образов. Автореферат дисс. ... канд. физ.-мат. наук: 05.13.11. Тбилиси, 2006. 20 с.

22. Перепелицин С. О. Система захисту від загроз удару БпЛА із використанням блоків нейромережевого аналізу. *Науковий журнал*. 2020. № 1 (45). С. 19–27. DOI: 10.18372/2310-5461.45.14579.

23. Дзелендзяк У. Ю., Пазинюк М. Ю. Система виявлення літальних апаратів на основі аналізу звукових сигналів. *Computer systems and networks*. 2023. Vol. 5. № 1. С. 29–35. DOI: <https://doi.org/10.23939/csn2023.01.029>.

24. Ясенко Л. С., Соболев В. В., Солодчук М. О., Алексеев С. В. Особливості створення бази даних для перевірки нейронних мереж обробки зображень, що були отримані з безпілотних авіаційних комплексів. II наук.-техн. конф. Держ. наук.-досл. ін-ту випробувань і сертифікації ОБТ. 28.09.2023. С. 351–352.

25. Приставка П., Чолишкіна О., Козачук О., Яременко Д. Нейромережева автоматизація наповнення набору даних аерофотозйомки. *Інформаційні технології та суспільство*. 2022. № 2(4). С. 88–99. URL: <https://doi.org/10.32689/maup.it.2022.2.12>.

26. Guoxiang Li, Xuejun Wang, Yun Li, Zhitian Li. Adaptive Clustering Object Detection Method for UAV Images Under Long-tailed Distributions. *Information Technology and Control*. 2023. Vol. 52. № 4. P. 1025–1044. DOI: 10.5755/j01.itc.52.4.33460.

27. Голенко М. Ю., Іванов Д. І., Єфіменко А. А., Воротніков В. В. Аналіз методів розпізнавання об'єктів та компресії зображень під час аерофотозйомки з безпілотних літальних апаратів. *Технічна інженерія*. 2023. № 1(91). С. 146–155. DOI: [https://doi.org/10.26642/ten-2023-1\(91\)-146-155](https://doi.org/10.26642/ten-2023-1(91)-146-155).

28. Москаленко Ю. В. Методи розпізнавання за діагностичним сигналом на основі гібридних нейронних мереж. дис ... докт. філософії: 122. Київ. 2020. 175 с.

29. Demuth H., Beale M. Neural Network toolbox for use with MATLAB. URL: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/http://cda.psych.uiuc.edu/matlab_pdf/nnet.pdf (дата звернення: 09.06.2024).

30. Навчання машин та штучний інтелект. Метод. вказівки до виконання лабораторних робіт з дисципліни «Технології програмування» для студентів напряму 163 «Біомедична інженерія» для всіх форм навчання. Уклад. М. В. Верескун. Маріуполь : ПДТУ, 2019. 61 с.

31. Довідник з MATLAB. Електронний навч. пос. з курсового і дипломного проектування. Київ: НТУУ «КПІ», 2013. 132 с.

32. Дослідження комп'ютерних систем штучного інтелекту. Методичні вказівки до лабораторних робіт для студентів 5-го курсу спеціальності КСМ. URL: <chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.uzhnu.edu.ua/en/infocentre/get/10973> (дата звернення: 10.06.2024).

33. Методи та технології напівкерованого навчання: курс лекцій. Київ: КПІ ім. Ігоря Сікорського, 2022. URL: <chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://ela.kpi.ua/server/api/core/bitstreams/214edd4c-0556-4c8b-b56b-1fd246c9def5/content> (дата звернення: 10.06.2024).

ALGORITHM FOR CREATING THE NEURAL NETWORK FOR CLASSIFICATION OF STRIKING UNMANNED AERIAL VEHICLES

Husak Yurii (Doctor of Military Sciences, Professor)¹
Vasylenko Olha²

¹ National Defence University of Ukraine, Kyiv, Ukraine

² The Central Research Institute of the Armed Forces of Ukraine, Kyiv, Ukraine

Formulation of the problem in general. Creating neural networks for a specific task is a complicated process, as many parameters need to be taken into account. Various software tools are available for this purpose. Analysis of software tools designed for modeling neural networks shows that they require experience in developing program codes in various programming languages, and also make it possible to solve a narrow range of tasks, i.e. they are narrowly focused. Therefore, it is worth considering the MatLab system modeling software package, which contains many built-in libraries for creating neural networks for solving problems of various classes, namely, the Neural Network Toolbox, a package of application programs designed to model neural network architectures.

Research methods. In conducting the study, the authors applied the basic principles of artificial intelligence theory. By combining methods and systems of artificial intelligence, they were able to systematically approach the development of an algorithm for creating a neural network for classifying striking UAVs.

Analysis of recent researches and publications. Given the prospects of artificial intelligence technologies, the development of neural networks for UAV classification in order to form heterogeneous swarms of striking UAVs in the context of the Russian-Ukrainian war is an important task. Many papers consider the use of different types of neural networks for object recognition and classification, present theoretical materials on the design of neural networks and methods for finding solutions in such systems for different classes of tasks. At the same time, these works hardly pay attention to how to choose the type and architecture of a neural network, as well as the algorithm for creating them, for example, using well-known application packages.

Presenting the main material. The paper shows that artificial intelligence methods, namely neural networks, can be used to solve the problem of classifying striking UAVs. An algorithm for creating a neural network for classifying striking UAVs according to their tactical and technical characteristics is proposed. Approaches to designing the architecture, training methods, data preparation for training, training and testing of a neural network for classifying striking UAVs are described. A structural and functional diagram of the neural network is developed. Based on the analysis of different types and architectures of NNs, a convolutional neural network CNN was selected. In the MATLAB package, the Pattern Recognition app was selected to classify striking UAVs as a tool for building the architecture of the neural network. To train the NN, the database of striking UAVs was divided into training, validation, and test sets in the ratio of 70% : 15% : 15%, respectively. The network is trained using the Levenberg-Marquardt back-propagation algorithm. The evaluation results indicate that the overall accuracy of the neural network is 98.9% and the error value is 0.03868, which indicates the high performance of the network.

Elements of scientific novelty. The scientific novelty lies in the proposed algorithm for creating a neural network for classifying striking unmanned aerial vehicles, which summarizes the typical stages and processes of classifying a swarm of striking unmanned aerial vehicles based on artificial intelligence methods.

Practical significance of the article. The practical significance of the research results is that they can be used to create a neural network depending on the complexity and type of task, as well as to classify striking UAVs depending on their tactical and technical characteristics.

Keywords: algorithm for creating the neural network, striking unmanned aerial vehicles, swarm, neural network, UAV classification, russian-ukrainian war.

References

1. Hlybovets, M. M., Oletsky, O. V., (2002). Artificial Intelligence Systems. Kyiv : KM Academy, 366.
2. Pogudina, O. K. et al. (2021). Methodology for the formation of the intellectual component of the agent system of a swarm of unmanned aerial vehicles. Monograph. Kharkiv : NAU «KhAI», 219.
3. Avrunin, O. G., Vladov, S. I., Petchenko, M. V., Semenets, V. V., Tatarinov, V. V., Telnova, G. V., Filatov, V. O., Shmelev, Y. M., Shushlyapina, N. O., (2021). Intelligent automation systems. Monograph. Kremenchuk: «Novabuk» Publishing House, 322.
4. **Methods and systems of artificial intelligence.** (2017). Study guide for students majoring in 6.050101 «Computer Science». Compiled by A. Savchenko, O. Sinelnikov. Kyiv : NAU, 190.
5. Porokhova O. Ye., (2019.) The essence and problems of artificial intelligence. Odesa, 30.
6. Nikitina, L. O., Nikitin, S. O., (2018). Models and methods of artificial intelligence in computer games. Kharkiv: «Printing house Madrid», 102.
7. **Methods and systems of artificial intelligence.** Educational and scientific research. (2019). In. D. V. Lubko, S. V. Sharov. Melitopol : FOP Odnorog T. V., 264.
8. Shevchenko, A. I., Agarkov, A. V., Azarenko, D. S., Gerasimov, I. G., Dorokhina, G. V., Ivanova, S. B., Nitsenko, A. V., Shelepov, V. Yu., (2014). Analysis and synthesis of communication information. Monograph of the IPCI of the Ministry of Education and Science of Ukraine and the National Academy of Sciences of Ukraine. Donetsk. IPSI «Science and Education», 212.
9. **Methods and systems of artificial intelligence.** (2017). Study guide for students of specialty 122 «Computer Science». Compiled by I. Udovyyk, H. Korotenko, L. Korotenko, V. Trusov, A. Khar. Dnipro : State Higher Educational Institution «National Mining University», 105.
10. Rudenko, O. G., Bodiansky, E. V., (2006). Artificial neural networks. Study guide. Kharkiv: SMIT Company LLC, 404.
11. Subbotin, S. O., (2020). Neural networks: theory and practice. Study guide. Zhytomyr : O. O. Yevenok Publishing House, 184.
12. **Air Force Strategy 2035.** (2020). Vinnytsia. CAF of the Armed Forces of Ukraine, 40.
13. Zhikai, Ya., (2023). The development and application of UAV intelligent machine learning system based on artificial intelligence. DOI: 10.56028/aehtsr., 140.
14. Mrad, I., Samara, L., Abdellatif, Alaa Awad. Federated Learning for UAV Swarms Under Class Imbalance and Power Consumption Constraints DOI:

10.1109/GLOBECOM46510.2021.9685143.

- 15. Kharchenko, O. B., Artyushin, L. M., Kononov, O. A.,** (2022). Prospects for the realization of joint use of unmanned aerial vehicles. DOI: <https://doi.org/10.54858/dndia.2022-18-1>.
- 16. Artyushin, L. M., Kononov, O. A., Gerasimenko, V. V., Nausenko, B. Y.,** (2022). Method of choosing the option for the implementation of group use of unmanned aerial vehicles. *Modern information technologies in the field of security and defense*. 1 (43), 48–59.
- 17. Shovkoshytnyi, I. I., Vasylenko, O. A.,** (2023). Problematic issues of swarming use of striking unmanned aerial vehicles, *Modern information technologies in the field of security and defense*. 48, № 3, 27–34. DOI:10.33099/2311-7249/2023-48-3-27-34.
- 18. Kompaniets, O. M., Klyushnikov, I. M., Dmitriev, A. G.,** (2023). Comprehensive analysis of the influence of factors on the effectiveness of controlling swarms of unmanned aerial vehicles. *Weapons systems and military equipment*. 3 (75), 66–70.
- 19. Korolyov, V., Ogurtsov, M., Khodzinsky, A.,** (2021). Statement of the problem of complete set of UAV group on the basis of models of granular calculations and fuzzy logic. *Cybernetics and Computer Technologies*. 25–38.
- 20. Ivashchenko, A. O.,** (2022). Information and software for the identification system of unmanned aerial vehicles. Bachelor's thesis. Sumy. Sumy State University, 47.
- 21. Oganezov, A. L.,** (2006). Application of neural networks in tasks of pattern recognition. Abstract of dissertation. Candidate of Phys.-Math. sciences: 05.13.11. Tbilisi, 20.
- 22. Perepelitsyn, S. O.,** (2020). Protection system against UAV attack threats using neural network analysis units. *Science-intensive technologies*. 1 (45). DOI: 10.18372/2310-5461.45.14579.
- 23. Dzelendziak, U. Y., Pazyniuk, M. Y.,** (2023). Aircraft detection system based on the analysis of sound signals. *Computer systems and networks*. 5, 1. DOI: <https://doi.org/10.23939/csn2023.01.029>.
- 24. Yasenko, L. S., Sobolev, V. V., Solodchuk, M. O., Alekseev, S. V.,** (2023). Features of creating a database for testing neural networks for image processing obtained from unmanned aerial vehicles. II Scientific and Technical Conference of the State Research Institute for Testing and Certification of WMH. 28.09.2023. 351–352.
- 25. Prystavka, P., Cholyskhina, O., Kozachuk, O., Yaremenko, D.,** (2022). Neural network automation of filling the aerial photography data set. *Information technology and society*. 2 (4), 88–99. DOI: <https://doi.org/10.32689/maup.it.2022.2.12>.
- 26. Guoxiang, Li, Xuejun, Wang, Yun, Li, Zhitian, Li.,** 2023. Adaptive Clustering Object Detection Method for UAV Images Under Long-tailed Distributions. *Information Technology and Control*, 52. 4. 1025–1044. DOI 10.5755/j01.itc.52.4.33460.
- 27. Golenko, M. Y., Ivanov, D. I., Efimenko, A. A., Vorotnikov, V. V.,** (2023). Analysis of methods of object recognition and image compression during aerial photography from unmanned aerial vehicles. State University «Zhytomyr Polytechnic», 1 (91). 146–155. DOI: [https://doi.org/10.26642/ten-2023-1\(91\)-146-155](https://doi.org/10.26642/ten-2023-1(91)-146-155).
- 28. Moskalenko Y. V.,** (2020). Recognition methods by diagnostic signal based on hybrid neural networks. PhD thesis: 122. Kyiv, 175.
- 29. Demuth, H., Beale, M.** Neural Network toolbox for use with MATLAB [online]. Available at: chrome-extension://efaidnbmnnnibpcajpcgclefindmkaj/http://cda.psych.uiuc.edu/matlab_pdf/nnet.pdf [Accessed: 09 June 2024].
- 30. Machine learning and artificial intelligence.** (2019). Methodical instructions for performing laboratory work in the discipline «Programming Technologies» for students of the direction 163 «Biomedical Engineering» for all forms of education. Compiled by. M. V. Vereskun. Mariupol, 61. 31.
- Guide to MATLAB** (2013). Electronic tutorial for course and diploma design. Kyiv: NTUU «KPI», 2013.
- 32. Research of computer systems of artificial intelligence.** Methodical instructions for laboratory work for 5th year students majoring in CSM. [online]. Available at: <chrome-extension://efaidnbmnnnibpcajpcgclefindmkaj/https://www.uzhnu.edu.ua/en/infocentre/get/10973> [Accessed: 09 June 2024].
- 33. Methods and technologies of semi-guided learning: A course of lectures.** K.: Igor Sikorsky Kyiv Polytechnic Institute. (2022). [online]. Available at: <chrome-extension://efaidnbmnnnibpcajpcgclefindmkaj/https://ela.kpi.ua/server/api/core/bitstreams/214edd4c-0556-4c8b-b56b-1fd246c9def5/content> [Accessed: 09 June 2024].