

РОЗПІЗНАВАННЯ ВІЙСЬКОВОЇ ТЕХНІКИ ЗА СУПУТНИКОВИМИ ЗНІМКАМИ ЗА ДОПОМОГОЮ ШІ

В. О. Витрикуш, Н. В. Гузинець

Національний університет “Львівська політехніка”,

кафедра електронних обчислювальних машин

E-mail: vitalii.vytrykush.mkiks.2022@lpnu.ua, nataliia.v.huzynets@lpnu.ua

© Витрикуш В. О., Гузинець Н. В., 2025

Дослідження систем навчання нейронних мереж, що використовуються для розпізнавання зображень військової техніки за допомогою відкритих, але обмежених супутникових зображень військової техніки. Реалізація виконується з погляду обробки додавання зображень у вигляді внутрішнього циклу, який заморожує всі числові параметри навчання нейронної мережі з одночасним вибором дещо більшої частини навчального набору, включаючи інкрементну частину штучних зображень, доданих до системи. Під час постановки цієї проблеми можна виділити дві широко використовувані мережі розпізнавання ResNet50 і Xception. Аналізуючи ці дві мережі, можна сказати, що в результаті дослідження найкращим методом навчання мережі є метод, який складається зі стратегії розширення даних. Це означає, що під час навчання ми можемо розширити базу образів, завдяки різним представленням цих образів.

Ключові слова: ResNet50, Xception, розширення даних, розпізнавання зображень.

Вступ

На сьогодні існує кілька можливих стратегій, які використовуються для класифікації зображень. Одним із таких підходів є згорткові [2] нейронні мережі, які дуже корисні для класифікації одного зображення. Цей підхід можна використовувати в повсякденному житті, для розпізнавання облич, тварин, порід тварин тощо [3]. Таке широке використання забезпечується саме доступністю та величезною кількістю матеріалу для навчання нейронної мережі.

Але розглядаючи цю технологію з погляду військових, необхідно розуміти, що багато інформації, яка може бути використана для створення універсальної мережі розпізнавання військової техніки, може бути надана тільки в разі аналізу великої кількості різної інформації. Але треба врахувати, що часто це не є просто, тому що багато військової інформації приховано. Це ставить завдання, яке потребує вирішення проблеми створення навчальної бази, яка може забезпечити універсальність знань.

У 2017 році деякі дослідження використовували обмежений набір даних. Основною ідеєю дослідження було використання випадкового пошуку для знаходження оптимального значення гіперпараметрів, що дало змогу отримати високу точність 95 % і зменшити обчислювальну складність до 65 % під час навчання згорткової [2] мережі. На основі цих результатів можна зробити висновок, що порівняння згорткового [2] мережевого навчання з нуля з комплементом перевершує архітектуру навчання з перенесенням більш ніж на 35 %.

Крім того, об'єднавши ці алгоритми з методом збільшення зображення та вимірявши рівень точності, можна зробити висновок, що архітектури спеціального дизайну можна оптимізувати та отримати результати з високою точністю, використовуючи менше обчислювальних ресурсів. Тому згорткові [2] нейронні мережі найкраще підходять для вирішення задачі розпізнавання зображень.

Огляд літературних джерел

Сьогодні існує дуже багато досліджень, які пов'язані з навчанням глибоких мереж та комп'ютерним зором. Усі вони охоплюються надзвичайно широким спектром методів та алгоритмів, які містять як класичні підходи, так і нові. У цьому розділі розглядаються наукові праці та ресурси, які стосуються розпізнавання військової техніки за допомогою комп'ютерного зору та глибоких мереж.

В основі наукових статей лежать два основні інструменти для розробки моделей глибокого навчання, такі як Keras [1] та TensorFlow [10]. Ці два інструменти є пов'язані, оскільки Keras є високорівневою бібліотекою для машинного навчання, яка працює поверх TensorFlow та надає простий та зрозумілий API для побудови нейронних мереж. TensorFlow є інструментом, який дає змогу створювати продуктивні моделі машинного навчання. У сукупності цих двох моделей ми маємо змогу створювати, тренувати та оптимізовувати складні глибокі нейронні мережі.

Також розглядається бібліотека Scikit-learn [6], яка надає набір алгоритмів для реалізації класичного машинного навчання, включаючи кластеризацію, регресію, класифікацію та оптимізацію. Це один з найпоширеніших інструментів, який використовується для попередньої обробки даних та аналізу ефективності моделей.

Однією з найвідоміших моделей для класифікації зображень, яка у 2012 році стала проривом завдяки використанню глибоких згорткових шарів та методу нормалізації локальних відповідей (LRN). Вона значно покращила точність класифікації на ImageNet. Її назва AlexNet, вона описана у праці [12].

В одній із робіт [7], яка присвячена розробці ResNet, – це глибока залишкова нейронна мережа, яка використовує резидуальні зв'язки та змогла подолати проблему зникнення градієнта, що дозволило створити глибші та стабільніші моделі розпізнавання.

Автор [8] розглядає архітектуру Xception, яка є розширенням мережі Inception, та використовує глибокі згорткові мережі із роздільними згортками. Так, цей підхід покращує обчислювальну ефективність та продуктивність моделей.

Автори [2] у своїй праці розглядають один з ключових методів, який використовується сьогодні у задачах комп'ютерного зору YOLO (You look only once). У своїй праці автори пропонують покращення моделі YOLOv3, які передбачають використання багаторівневих ознак та вдосконалену архітектуру. Переваги цієї моделі є у високій швидкості та точності виявлення об'єктів на зображеннях та відео.

У праці [4] безпосередньо досліджено можливості методів глибокого навчання для розпізнавання саме військової техніки у соціальних мережах. Автор використовував сучасні підходи для виявлення об'єктів, чим саме демонструє можливості штучного інтелекту у військовій аналітиці.

Використання гіперпараметрів та їх оптимізація для підвищення ефективності моделей машинного навчання може значно покращити продуктивність алгоритмів розпізнавання, як описали автори та інші у своїй праці [11]. Саме оптимізація алгоритмів є однією з ключових пік час тренування нейронних мереж. Саме бібліотека TensorFlow [10] надає можливість використання широкого набору інструментів та налаштування параметрів навчання, які дають змогу досягти кращої узгодженості в моделях машинного навчання.

Проаналізувавши джерела, можна зробити висновок про те, що на сьогодні є надзвичайно велика кількість методів та підходів глибокого навчання та комп'ютерного зору, що забезпечують

можливість досягти високої точності та ефективності. Саме сучасні архітектури нейронних мереж, такі як ResNet чи Xception, а також об'єктне розпізнавання YOLO, дають змогу створювати високопродуктивні засоби та рішення для різних сфер застосувань.

Постановка задачі

Метою роботи є розробка програмного забезпечення, яка дасть змогу розпізнавати типи військової техніки за аерофотознімками. Йдеться про створення власного набору даних, його розширення та адаптація під реальні обставини. А також навчання моделей штучного інтелекту для вміння розпізнавати військову техніку в режимі реального часу. Тобто йдеться про аналіз та збір доступних матеріалів в мережі “Інтернет”, а також використання 3d-моделей для штучного створення набору даних та подальше використання його під час навчання моделей штучного інтелекту.

Дані

Для розробки та тестування системи було зібрано навчальні дані для трьох класів [5] з урахуванням конфлікту на сході України: основні бойові танки Т-72, БМП та інші зображення різноманітної цивільної техніки, вуличні сцени та повсякденні зображення із зони конфлікту. У той час як перші два класи, Т-72 і БМП, представляють гусеничні машини, які широко використовують обидві сторони конфлікту, третій клас, по суті, є негативним класом, включеним у навчальні дані, щоб дозволити мережам дізнатися про інші типи опублікованих зображень із зони конфлікту [5]. Зображення були зібрані з трьох джерел: фотографії, опубліковані в соціальному сервісі обміну фотографіями Flickr, кадри з відео з YouTube [5] у зоні конфлікту та фото, зібрані з інтернету [5]. Джерела даних і обсяги наведені у табл. 1.

Таблиця 1

Результати набору даних з різних джерел

Name	Flickr	YouTube	інтернет	Всього
Т-72	773	513	171	1457
БМП	182	884	62	1088
Інші	819	559	108	1477
Всього	1774	1907	341	4022

Як показано у табл. 1 використання різних джерел дозволило приблизно збалансувати кількість даних для кожного класу. Наприклад, хоча БМП-зображення були порівняно рідкісними на Flickr, цей недолік можна було б усунути, зібравши більше даних із відео YouTube. Відео дискретизувалося зі швидкістю один кадр на секунду. Збір даних із вищевказаних джерел потребував ручної попередньої обробки. Дані Flickr, наприклад, довелося відфільтрувати для фотографічного вмісту, оскільки користувачі також завантажують на сервіс намальовані від руки ілюстрації та фотографії масштабних моделей. Відео на YouTube, опубліковані в зоні бойових дій, також підбиралися для кадрів Т-72, БМП або зображень навколишнього ландшафту. Вилучення кадрів із відео на YouTube майже подвоїло обсяг даних і забезпечило зображення транспортних засобів за різного освітлення та погодних умов, а також під різними кутами та з різних відстаней. Однак потрібно зазначити, що вилучення кількох кадрів з одного відео може заповнити набір даних зображеннями, які мають багато інших функцій, окрім транспортних засобів, зображених на них. Навчання нейронної мережі на цих даних викликано ризиком перетренування, тобто особливостями навчання та нерелевантними завданнями розпізнавання, які необхідно враховувати під час навчання.

Штучне збільшення кількості зображень

Оскільки військова промисловість має проблеми з великими наборами даних для навчання, то з міркувань безпеки був розроблений новий підхід для розширення бази знань. А саме: штучне збільшення кількості зображень у базі знань. Спочатку всі зображення змішуються, потім діляться на частини, одна з яких використовується для навчання нейронної мережі.

Далі генеруються штучні зображення, які використовуються виключно для навчання нейронної мережі, за схемою, наведеною на рис. 1. Штучно створені зображення вважаються виключно доповненням до загального набору даних, включеного в навчання.

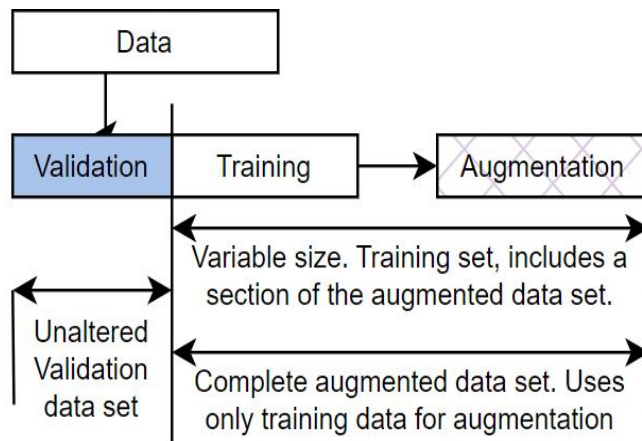


Рис. 1. Підготовка зображень до навчання

Збільшення кількості зображень у навчальних наборах даних відбувається завдяки застосуванню випадкових перетворень до вхідних зображень. Відбуваються різні трансформації, наприклад, збільшення або зменшення розміру зображення, дзеркальне відображення зображення в різних напрямках, застосування різних фільтрів, а також застосування інших видів трансформації. Сьогодні метод розширення кількості зображень активно використовується для завдань класифікації транспортних засобів, розпізнавання рукописного тексту, класифікації тварин тощо. Це дає змогу значно підвищити точність розпізнавання реальних зображень.

Отже, процес доповнення даних забезпечує генерацію додаткової копії зображення, яка модифікується шляхом застосування різноманітних перетворень до зображення. Для виконання цього завдання можна використовувати бібліотеку глибокого навчання Keras [1], яка дає змогу легко виконувати різні геометричні операції, застосовані до зображення. Завдяки цьому можна ефективно поповнювати навчальну базу знань. Під час перетворення даних до зображення можна застосувати такі перетворення:

- Зсув зображення до 20 градусів.
- Поворот зображення до 10 градусів по будь-якій осі, для відображення військової техніки з різних ракурсів.

- Зміщення зображення по висоті на 5- 7 %, а також по вертикалі до 20 %.

Збільште зображення до 120 %, щоб обрізати та сфокусуватися на об'єкті.

Архітектура мережі та оптимізація

Архітектури мереж

Під час підходу до проблеми розпізнавання військової техніки за зображеннями існує дві стратегії: стратегія навчання мережі з нуля за допомогою додавання бази знань і стратегія навчання нейронної мережі без використання доповнених зображень. Якщо не використовувати додаткові зображення, можна використовувати вже навчені нейронні мережі на інших об'єктах. Однак стратегія навчання мережі з нуля потребує використання згорткової мережі, такої як мережа Хатту-

нена, яка розпізнає транспортні засоби, для введення згорткових шарів. Така структура називається ConvNet, яка складається з двох згорткових блоків, які розділені ще на два шари, кожен з яких містить 32 і 64 карти функцій, після чого виконується операція максимального об'єднання результатів. Якщо після операції злиття карти функцій вирівнюються перед повністю зв'язаним шаром з 256 вузлів, після чого активується Softmax, який дає змогу розділити результат групування на 3 класи. Друга стратегія навчання продовжує використовувати залишкову мережу ResNet50. Це декілька, які використовуються в задачах класифікації, виявлення та локалізації.

Основна відмінність між цими двома мережами полягає в тому, що залишкові мережі пропускають пропуск з'єднання, щоб додати початковий вихідний рівень до вихідного рівня, показуючи, що така архітектура полегшує оптимізацію глибоких нейронних мереж, забезпечуючи значне збільшення глибини мережі. Робота над глибокими залишковими мережами продовжується і дала ще більш глибокі архітектури, які показують головний елемент залишкової мережі, залишковий блок, який після активації попереднього рівня є вхідним. Потім цей вхід передається на рівень згортки, після чого відбувається пакетна нормалізація (BN) і активація випрямленої лінійної одиниці (ReLU). Кількість карт функцій, отриманих у кожному згортковому шарі, додається разом із глибиною мережі. Нарешті, активація виводу додається до виходу перед іншою активацією ReLU, вихід якої діє як вхід для наступної блокової одиниці. ResNets накопичує кілька залишків блоків перед виконанням усередненого пулу, який виводить 2048-вимірний вектор, який потім передається на повністю підключений рівень. Keras [1] надає 50-шарову залишкову сітку (ResNet50) із попередньо встановленими вагами на ImageNet [7]. Крім того, ImageNet містить багато різних типів транспортних засобів [9], зокрема основні бойові танки та бронетранспортери, які, як можна припустити, демонструють багато функцій, пов'язаних із поточним завданням розпізнавання в мережі. Це дає змогу використовувати ResNet50 як екстрактор функцій, беручи вихідні дані середнього рівня об'єднання та надсилаючи ці функції до повністю зв'язаного блоку, щоб отримати трикласовий розподіл ймовірностей Softmax.

Математична установка за res-net

Давайте розглянемо архітектуру DNN, зокрема швидкість навчання та інші гіперпараметри, які можуть досягати класу ознак F . Отже, для всіх $f \in F$ існують параметри W , які ми можемо отримати після навчання мережі для цього набору даних. Якщо f^* позначає функцію, яку ми дійсно хотіли б знайти (результат найкращої можливої оптимізації), але якщо вона не в F , тоді ми намагаємося знайти f_1 , що є найближчим відповідником в F . Якщо ми розробимо надійнішу архітектуру G , ми маємо отримати найкращий результат g_1 , який кращий за f_1 . Але якщо $F \not\subseteq G$, немає гарантії, що наведене вище припущення буде виконуватися. g_1 може бути гіршим, ніж f_1 , і це проблема деградації. Отже, суть полягає в тому, що якщо глибші функціональні класи нейронної мережі містять простіші та тонші мережеві, функціональні класи, то ми можемо гарантувати, що глибша мережа збільшить якість виявлення ознак вихідної зовнішньої мережі.

Основною мотивацією оригінальної роботи ResNet було вирішення проблеми деградації в глибокій мережі. Додавання додаткових рівнів до достатньо глибокої нейронної мережі спочатку призведе до насиченості в точності, а потім точність погіршується. Автор навів таку картину тестової помилки з набором даних Cifar-10 з використанням ванільної мережі, показаної на рис. 2 та рис. 3. Ідея залишкового блоку повністю базується на інтуїції, яка була пояснена раніше. Простіша функція (менша мережа) має бути підмножиною складної функції (глибша мережа), щоб можна було вирішити проблему деградації. Розглянемо вхід x і позначимо потрібне відображення входу на вихід $g(x)$. Замість того, щоб мати справу з цією функцією, ми будемо працювати з більш простою функцією $f(x) = g(x) - x$. Потім вихідне відображення перетворюється на $f(x) + x$. У статті є припущення [4], що легше оптимізувати залишкову $f(x)$, ніж сам вихідний g . Крім того, оптимізація залишків забезпечує уникнення проблеми у разі відображення ідентичності $f(y) \rightarrow y$ у глибоких мережах. Розглянемо залишкову блок-схему на рис. 4.

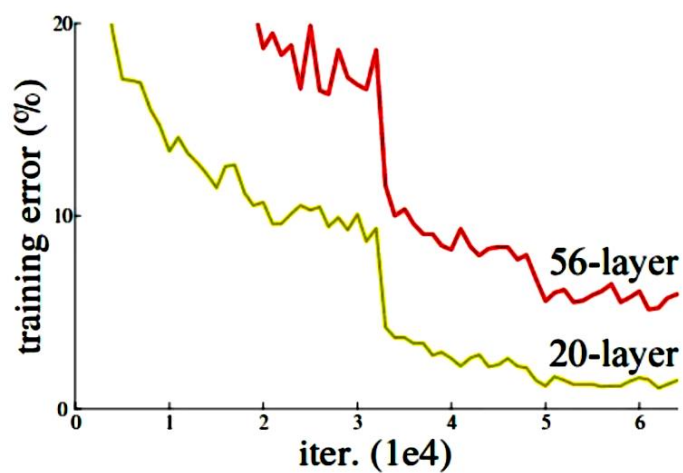


Рис. 2. Помилка класифікації з даними Cifar-10 зростає зі збільшенням числа шарів для тестових даних

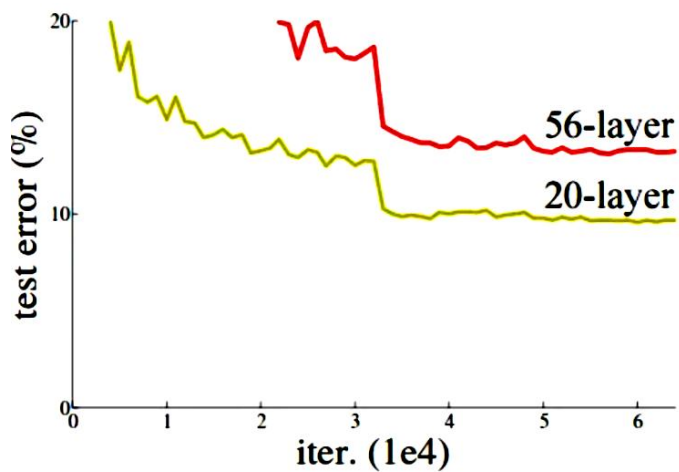


Рис. 3. Помилка класифікації з даними Cifar-10 зростає зі зменшенням числа шарів для тестових даних

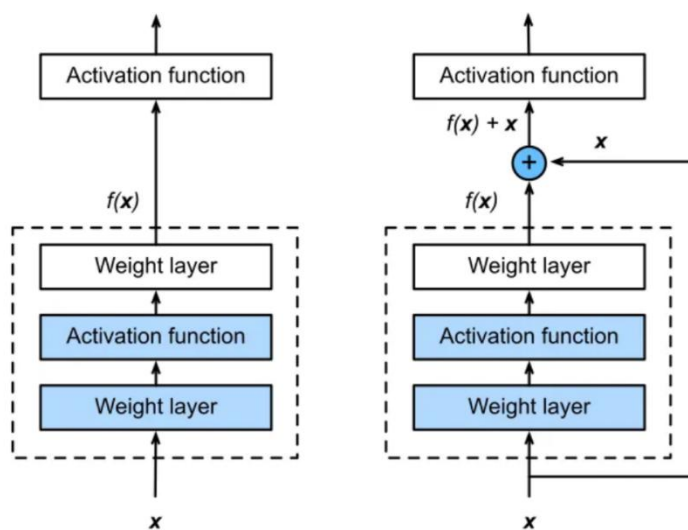


Рис. 4. Залишковий блок і пропуск з'єднання для відображення ідентичності

Формулювання залишкового навчання гарантує, що коли тотожні відображення є оптимальними (тобто $g(x) = x$), оптимізація призведе до нульових ваг функції залишків. ResNet складається з багатьох залишкових блоків, де залишкове навчання адаптується до кожного з кількох (зазвичай 2 або 3 шари) складених шарів. Будівельний блок показаний на рис. 2, а кінцевий результат можна побачити як $y = f(x, W) + x$. Тут W - вага, отримана під час навчання. Операція $f + x$ виконується швидким ("пропускаючи" 2/3 шарів) з'єднанням і поелементним додаванням. Це найпростіший блок, де при пропуску підключення не задіяні додаткові параметри. Поелементне додавання можливе лише за однакових розмірів f і x . Якщо це не так, тоді ми множимо вхідний сигнал x на матрицю проекції W_s так, щоб розміри f і x збігалися. У цьому випадку результат зміниться з попереднього рівняння на $y = f(x, W) + W_s * x$. Елементи в матриці проекцій також можна навчати.

Ми створимо ResNet із 50 шарами, дотримуючись методу, прийнятого в оригінальній статті Хе та ін [4]. Архітектура, прийнята для ResNet-50, відрізняється від 34-рівневої архітектури. Підключення швидкого доступу пропускає 3 блоки замість 2, і схематична діаграма нижче допоможе нам прояснити деякі моменти.

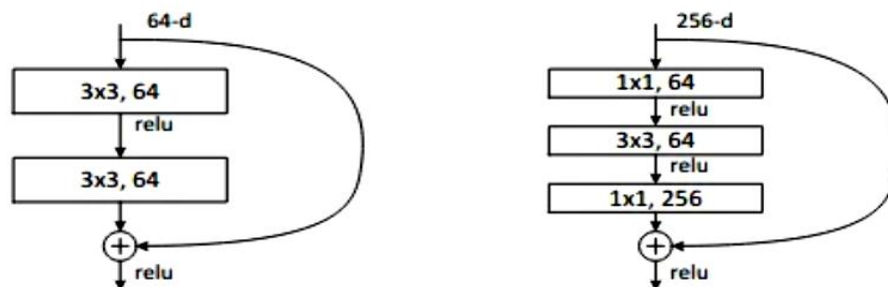


Рис. 5. Пропуск шарів у ResNet: 2 шари в ResNet-34 (зліва) та 3 шари, включаючи згортку 1×1 , у ResNet-50 (справа)

У ResNet-50 складені шари в залишковому блоці завжди матимуть шари згортки 1×1 , 3×3 та 1×1 . Згортка 1×1 спочатку зменшує розмірність, потім обчислює характеристики на вузькому шарі 3×3 , а потім знову збільшує розмірність на наступному шарі 1×1 . Використання фільтра 1×1 для зменшення та збільшення розмірності карт функцій до та після шару вузьких місць, як описано в моделі GoogLeNet Szegedy et al. в оригінальному документі. Оскільки в залишковому блоці немає шару об'єднання, розмірність зменшується на згортку 1×1 із кроком 2. Пам'ятаючи про це, можна створити ResNet-50 за допомогою TensorFlow 2.0.

Оптимізація мережі

Як показано, експерименти склалися з двох різних сценаріїв: навчання невеликої мережі ConvNet з нуля, використання переваг онлайн-розширення даних і використання попередньо навченого ResNet50 для визначення функцій із необроблених даних перед тонким налаштуванням повністю підключеного блоку. Випадковий пошук використовувався для пошуку оптимальних гіперпараметрів для мережі в усіх експериментах, дотримуючись підходу, запропонованого Bergstra et al [11], як реалізовано в бібліотеці sci-kit-learn [6]. Випадковий пошук було виконано на Keras [1] за допомогою серверної частини Theano, щоб подолати проблеми з керуванням пам'яттю за допомогою TensorFlow. Доповнення даних не використовувалося під час випадкового пошуку. Для експерименту з потоком очищення було оцінено 19 мільйонів параметрів моделі, які можна навчити, 50 випадкових налаштувань гіперпараметрів і 50 епох навчання з 3-кратною перехресною перевіркою для кожного набору. Для менш дорогого експерименту з 1 мільйоном параметрів моделі, які можна тренувати, було оцінено 50 випадкових налаштувань гіперпараметрів, навчання в 100 епохах з 10-кратною перехресною перевіркою для кожного параметра. Табл. 2 показує випад-

ково знайдені розділи гіперпараметрів, які були збиті, щоб містити загалом 576 можливих налаштувань, тоді як крайній правий стовпець показує найкраще значення, знайдене під час випадкового пошуку.

Таблиця 2

Випадковий пошук діапазонів гіперпараметрів

Architect	Hyperparam	Range	Best
Conv-Net	Batch size	{32,64,128,256}	256
	Learning rate	{0.1,0.01,0.0001}	0.0001
	L2	{0,001, 0,0001}	0.001
	Dropout rate	{0.25,0.5,0.75}	0.5
	Node in FC	{128,256,512}	256
ResNet	Batch size	{32,64,128,256}	32
	Learning rate	{0.1,0.01,0.0001}	0.01
	L2	{0,001, 0,0001}	0.001
	Dropout rate	{0.25,0.5,0.75}	0.5
	Node in FC	{128,256,512}	512

Експерименти

У цьому розділі описано навчання та оцінку двох мережевих структур, наведених вище. Обидві мережі були реалізовані за допомогою Keras [1], який працює на основі TensorFlow [10]. Вагові параметри для всієї мережі були оптимізовані за допомогою стохастичного градієнтного спуску (SGD) із розмірами міні-пакетів, перерахованими в табл. 2, імпульсом Нестерова 0,9 та розпадом 10^{-5} для мінімізації втрат для категоріальної крос-ентропії між істинними та прогнозованими мітками. Продуктивність обох мереж була оцінена за допомогою категоричної точності. В обох експериментах використовувався 10-кратний розподіл перехресної перевірки. Дані, наведені в розділі III-A, представлені в наборі для навчання (90 %) і перевірки (10 %). Для експерименту дані були заповнені онлайн за допомогою конфігурації, описаної в розділі III-B. Мережа була навчена протягом 1000 епох. Для кожної епохи було створено 2048 зразків для навчання та 256 для перевірки. Гіперпараметри були обрані на основі випадкового пошуку, показаного в табл. 2. ConvNet оцінено в експерименті (1), досягає середньої точності перевірки 0,7914 (SD = 0,0287) протягом 10 разів. На рис. 6 показано криву навчання для найкращої згортки (точність: 0,8398).

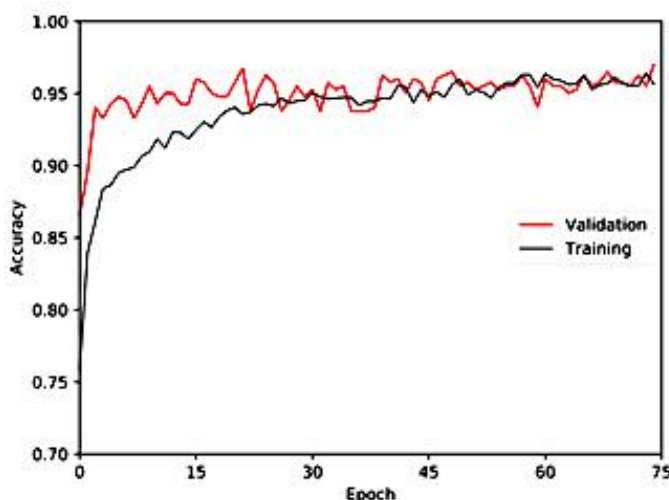


Рис. 6. Крива навчання для найкращого згортання експерименту: навчання згорткової [2] нейронної мережі (ConvNet)

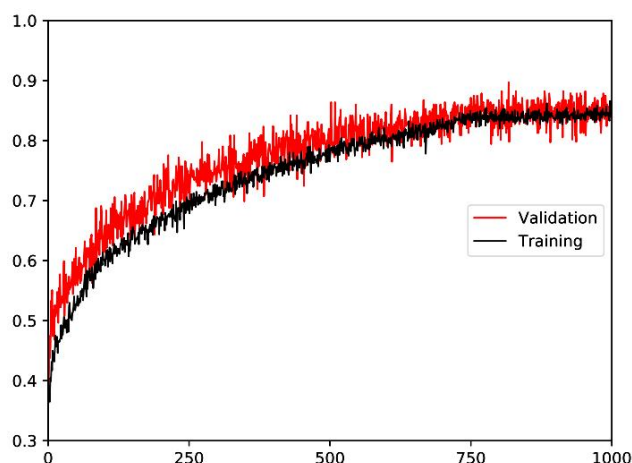


Рис. 7. Крива навчання для найкращого згортання експерименту:
передача навчання за допомогою залишкової нейронної мережі (ResNet50)

Для експерименту весь набір даних, описаний у Розділі III-A, було передано в ResNet50, попередньо навчений на ImageNet [7] для виділення ознак, запис результату кінцевого середнього шару об'єднання для створення 2048-вимірному вектора ознак для кожного зображення. Потім ці функції були розділені на набори для навчання та перевірки і подані в повну нейронну мережу з 512 вузлами, навченими протягом 75 епох. Використані параметри відповідали параметрам, знайденим у випадковому пошуку, як зазначено в табл. 2. Середня точність становила 0,9518 (SD = 0,0094). На рис. 3 показана крива навчання для найкращої збіжності моделі в експерименті 2 (точність: 0,9701). На рис. 6 і 7 показано, що хоча обидві мережі можуть навчитися розрізняти різні класи, ConvNet досягає плато після 750 епох. Незважаючи на збільшення обсягу даних, необроблені дані не видаються достатньо різноманітними, щоб покращити продуктивність ConvNet. Навчання передачі за допомогою ResNet показує, що модель, попередньо навчена на ImageNet, засвоїла багато функцій, необхідних для розрізнення різних типів транспортних засобів. Це призводить до високої точності лише через кілька епох, яку можна покращити шляхом подальшого навчання. Щоб оцінити здатність мереж узагальнювати дані, які не входять до набору, описаного в Розділі III-A, обидві мережі також були оцінені на жорсткому тестовому наборі, що містив 30 зображень зони бойових дій, зібраних з різних джерел. Ці зображення були призначені для представлення можливих джерел вмісту, що поширюється в соціальних мережах, починаючи від високоякісних професійних фотографій для преси до скріншотів із новинних відео, а також від створених користувачами відео до скріншотів публікацій у соціальних мережах і вмісту. Рівні 0,5 і 0,25 включено, щоб відобразити здатність інструменту позначати зображення, для яких імовірність Softmax перевищує заданий поріг. Це дає змогу, наприклад, враховувати неправильну класифікацію різних типів транспортних засобів і зображень, які містять кілька типів транспортних засобів. ResNet значно перевершує ConvNet на всіх рівнях точності.

Таблиця 3

Продуктивність на тестовому наборі

Architecture	Accuracy	Accuracy at 0.5	Accuracy at 0.25
ConvNet			
Best	0.533	0.433	0.667
Mean	0.440	0.353	0.663
ResNet			
Best	0.900	0.900	0.933
Mean	0.877	0.853	0.940

Ця стаття представляє систему, розроблену для розпізнавання військових машин на зображеннях у соціальних мережах за допомогою глибокого навчання. У документі, повністю створеному з використанням бібліотек глибокого навчання з відкритим кодом, архітектур і попередньо навчених моделей, досліджено два різні рішення для системи навчання розпізнавання транспортних засобів з обмеженими навчальними даними. У статті експериментували з іншими архітектурами нейронних мереж: згортковими [2] нейронними мережами та залишковими нейронними мережами. Результати показали, що тоді як згорткові мережі досягають прийнятного тесту на точність, залишкові нейронні мережі можуть краще узагальнювати за допомогою навчання перенесення, тобто використовуючи функції, взяті з іншого розподілу для опису зображень у поточному наборі даних. Навчання згорткової [2] нейронної мережі з нуля за допомогою доповнення даних для синтезу додаткових даних не могло зрівнятися з рівнем продуктивності, досягнутим за допомогою навчання перенесення. На додаток до впровадження додаткових інструментів у систему, майбутня робота може містити оцінку її продуктивності з використанням геологованих даних соціальних мереж, пов'язаних із війною в Україні, щоб оцінити, наскільки добре спостереження, зроблені за допомогою інструменту, корелювали з подіями під час їхнього розвитку. Крім того, сайт соціальних медіа часто є мультимодальним, тобто контент поданий за допомогою кількох способів вираження, таких як фотографії та текстовий опис, іншою можливістю також є обробка підписів до зображень. Методи обробки природної мови, такі як розпізнавання іменованих об'єктів, можуть допомогти отримати назви місць та інші важливі метадані для контекстуалізації зображення.

Передача навчання

На сьогодні одними з найпопулярніших згорткових мереж для розпізнавання зображень є ResNet50 і Xception [8], хоча є й багато інших. Для цих мереж на ImageNet [3] є ваги, які використовуються, як об'єкти для цих мереж з 1000 варіантів класифікації (1000 категорій об'єктів, таких як клавіатура, миша, олівець і багато тварин, переважно цивільні зображення). У разі використання невеликих наборів даних застосовується підхід перенесення навчання, який дає змогу використати вже навчені та адаптувати навчені глибокі нейронні мережі для вирішення конкретного цільового завдання, після чого можна здійснювати подальше навчання моделі на наявних даних. Такий метод є випробуваним та ефективним у різних сферах. Наприклад, розпізнавання шкідливого програмного забезпечення або аналізу крові та підрахунку лейкоцитів у медичних діагностичних системах на базі ResNet50 та Inception [9].

Дві нейронні мережі, які розглядаються, в своїй основі мають суттєво різну архітектуру. ResNet50 – глибока нейронна мережа, яка містить 50 шарів, вона є компактнішою, порівняно з її попередницею ResNet [10]. Її ключова ідея полягає у концепції залишкового навчання, коли кожен шар працює з залишковою різницею між вхідними та вихідними даними, у такий спосіб покращуючи процес оптимізації. Одразу порівнюючи Xception [8], яка містить 71 шар і є модифікованою версією Inception, де в основі лежать згортки, розділені по глибині [3], а не стандартні модулі. В основі структури цієї мережі лежить 36 шарів, які організовані у 14 модулів із залишковими зв'язками. Глибока згортка, за якою йде точкова згортка, лежать в основі Xception, що значно підвищує ефективність обробки зображень.

Зазвичай додавання нових шарів до вже навченої моделі не дасть жодних переваг, крім того, це може порушити вихідну архітектуру моделі. Тому підхід з перенесенням знань дає змогу уникнути цього, і його ідея полягає у тому, щоб замінити верхній рівень архітектури нейронної мережі на новий, який є адаптованим для конкретного завдання. У такому випадку модель складатиметься із двох частин: незмінного базового шару та нових навчальних шарів.

У цьому дослідженні для збереження оптимізованої базової архітектури вихідних моделей, було додано лише три нових щільних шари у вигляді пірамідальної конфігурації (1024-1024-512 нейронів) із коефіцієнтом випадання 0,2 перед шаром класифікації Softmax [1], який має стільки вузлів, скільки класів у задачі.

У результаті додаткове навчання гібридної моделі здійснюється з урахуванням не лише нових доданих шарів, а й частиною початкової мережі. Як стартові ваги використовуються попередньо навчені ваги, а подальше навчання виконується для останніх 5, 10, 15 або 20 рівнів мережі. У такому випадку використовуються не лише переваги попереднього навчання, а й відбувається пошук найоптимальнішого рівня для адаптації до конкретного військового застосування.

Псевдовипадкове розміщення

На сьогодні технологічні пристрої є менш варіативні на відміну від біологічних видів. Завдяки прогресу та розвитку у сферах 3D-дизайну та методів візуалізації стало можливим створення штучних зображень, що забезпечують практично необмежений обсяг навчальних даних для алгоритмів обробки зображень.

Через те, що зображення військової техніки є досить обмеженими у зв'язку із збереження секретності у військових установах, це породжує проблему малої кількості зображень, які можна використати для навчання моделей. Саме тому для вирішення цієї проблеми можна використовувати алгоритми машинного навчання для штучного збільшення кількості зображень.

Дослідження показали, що випадкове розміщення зображень (2D або 3D) на невідповідному фоні, який не має жодної кореляції з основним об'єктом, негативно впливає на якість навчання нейронних мереж для розпізнавання [6]. Тому важливо забезпечити гармонійне поєднання об'єкта та навколишнього середовища.

Раніше вже здійснювались експерименти із встановленням до 3D-моделей кораблів реальних фонових зображень, таких як верфі, узбережжя та аерофотознімки. Це дало змогу значно покращити якість та кількість навчальних даних. Проте методи штучного доповнення даних потребують обережності під час накладання зображень, вибору правильних ракурсів для 3D-візуалізацій, а також додавання текстур та реалістичних оболонок до моделей.

Застосування штучного розширення набору даних військової техніки дає змогу використати переваги під час використання 3D-моделей, які можуть бути створені та модифіковані у САПР. Наприклад, додавання камуфляжів різного виду, покриття, умови освітлення [4].

Ще одним етапом після модифікації 3D-моделей є додавання фону із зображенням різної місцевості, отриманих із Google Maps [7], щоб симулювати розміщення військової техніки в реальному середовищі бойових дій. Така система підтримує різні конфігурації камери, масштабування, варіанти освітлення, а також можливість додавання атмосферних ефектів (туман, дощ, вогонь, сніг, пісок). Крім того, як фони можуть використовуватися зображення вулиць з усього світу, прив'язані до GPS-координат.

Такий набір дає змогу синтезувати великий набір штучних зображень, які надалі можуть бути використані у різних схемах навчання для оцінки впливу різних доповнень до зображень на нейронних мережах.

Результати досліджень

У результаті аналізу використаних джерел можна зробити аналіз точності та швидкості сучасних методів глибокого навчання.

- Використовуючи ResNet-50, можна досягти точності у 76,3 % у рейтингу топ – 1 та 92,2 % у рейтингу топ-5 на ImageNet.
- Xception показує покращену продуктивність порівняно з Inception з точністю у 79,0 % на ImageNet.
- YOLOv3 має точність 59,7 % під час тестування на COCO базі даних, що показує про швидке та точне розпізнавання об'єктів.

- Використовуючи Bayesian optimization для оптимізації гіперпараметрів, можна скоротити час пошуку параметрів на 40 %.
- Автори у своїх працях, застосовуючи моделі глибокого навчання для розпізнавання військової техніки, досягали точності у 85 %.

Висновки

Проаналізовано дві різні контрольовані схеми машинного навчання для класифікації військових зображень: навчання змінних моделей передачі вихідних даних і навчання самопроєктуючих згорткових нейронних мереж. З погляду трансферного навчання ResNet виявилася хорошою нейронною мережею, яка швидко адаптувалася до нових варіантів класифікації, отриманих за результатами донавчання, із середнім значенням точності 88 %. З іншого боку, та сама схема передачі даних, застосована до нейронної мережі Xception, не була успішною в адаптації до нових випадків класифікації, здебільшого через дві причини: недостатньо великий/змінний набір даних і необхідність донавчання більшої частини початкової моделі, тобто більшої кількості останніх шарів, які можна модифікувати. Крім того, виявилося, що використовувана стратегія донавчання з використанням нових множин вхідних даних не впливає на точність роботи досліджених алгоритмів машинного навчання. Проте можна використати новий підхід отримання додаткових наборів множин даних для навчання шляхом генерації зображень об'єктів з 3D-моделей військових транспортних засобів, що може покращити якість і варіативність даних, щоб відповідати попередньо навченим мережам, таким як Xception та багатьом іншим. Другий напрям цього дослідження базується на навчанні згорткових мереж зі змінною архітектурою. Результати показали, що змінні архітектури більше підходять для різних випадків класифікації із середнім значенням 86,4 %. Це значно менші архітектури (≈ 40 разів), навчені з нуля, які виявилися приблизно такими надійними, як і великі попередньо навчені архітектури з цієї статті, з різницею ≤ 2 %.

Результати показали, що, хоча обидві мережі досягають прийнятної точності перевірки, залишкові нейронні мережі можуть узагальнювати набагато краще, використовуючи навчання перенесення, тобто використовуючи як основу моделі, отримані в результаті іншого навчання, і дотреновування їх під свої потреби. Водночас навчання згорткової нейронної мережі з нуля за допомогою розширення набору даних для синтезу додаткових множин даних не може зрівнятися з рівнем продуктивності, досягнутим за допомогою навчання з перенесенням.

Список літератури

1. Keras API reference. URL: <https://keras.io/api/>.
2. Redmon J., Farhadi A. YOLOv3: An incremental improvement. URL: <http://arxiv.org/abs/1804.02767>.
3. MathType. URL: <http://www.mathtype.com>.
4. Hiipala T. Recognizing military vehicles in social media images using deep learning. In: *IEEE International Conference on Intelligence and Security Informatics*. 2017. Doi: <https://doi.org/10.1109/ISI.2017.8004875>.
5. Digital Geography Lab. (n.d). URL: <https://github.com/DigitalGeographyLab/MilVehicles/>.
6. Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning*. 2019. Doi: <https://doi.org/10.48550/arXiv.1201.0490>.
7. He K., Zhang X., Ren S., J. Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. Doi: <https://doi.org/10.48550/arXiv.1201.0490>.
8. Chollet F. Xception: deep learning with depth-wise separable convolutions. In: *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition*, 2017. Doi: <https://doi.org/10.1109/CVPR.2017.195>.
9. Deng J., Dong W., Socher R., Li L.-J., Li K., Fei-Fei L. ImageNet: A large-scale hierarchical image database. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*. 2009. Doi: <https://doi.org/10.1109/CVPR.2009.5206848>.
10. Create production-grade machine learning models with TensorFlow. URL: <https://www.tensorflow.org/>.

11. Algorithm for hyper-parameter optimization. URL: https://papers.nips.cc/paper_files/paper/2011/hash/86e8f7ab32cfd12577bc2619bc635690-Abstract.html.
12. Krizhevsky, A., Sutskever, I., Hinton, G. E. ImageNet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*. 2012. Pp. 1097–110. Doi: <https://doi.org/10.48550/arXiv.1201.0490>.

RECOGNIZING MILITARY EQUIPMENT FROM SATELLITE IMAGES USING AI

V. O. Vytrykush, N. V. Huzynets

Lviv Polytechnic National University,
Department of Electronic Computing Machines

© Vytrykush V. O., Huzynets N. V., 2025

Neural networks training systems used for military equipment recognition on the images are considered in the article. The implementation, considered in the article, uses pretrained part of the model with freezing most of the trained parameters and fined tuning of some part of the model with using extra data set including artificially obtained images. Two widely used recognition networks was considered: ResNet50 and Xception. After these two networks analysis, we can say that the best approach for training the network is using data expansion strategy. This means that during training we can expand the image base by providing different representations of these images.

Keywords: ResNet50, Xception, data augmentation, image recognition.