УДК 005.8:331.5
DOI https://doi.org/10.32782/2663-5941/2025.2.2/28

Федорович В. І.
Західноукраїнський національний університет

МЕТОДИ ПІДВИЩЕННЯ ТОЧНОСТІ ВИЯВЛЕННЯ ВІЗУАЛЬНИХ АНОМАЛІЙ У ПРОГРАМНОМУ ЗАБЕЗПЕЧЕННІ НА ОСНОВІ ШТУЧНОГО ІНТЕЛЕКТУ

Виявлення аномалій є важливою характеристикою штучного інтелекту в автоматизованому тестуванні. Штучний інтелект, зокрема методи машинного навчання та нейронних мереж, достатньо широко використовуються для автоматизації процесу тестування та покращення ефективності виявлення помилок. Наприклад, інтелектуальні алгоритми можуть застосовуватись для визначення оптимальних шляхів проведення тестування програми, а також для прогнозування можливих проблеми в роботі програми, що дозволяє зменшити кількість помилок та скоротити час, необхідний для тестування програмного забезпечення. У цьому контексті дослідження методів штучного інтелекту для виявлення візуальних аномалій у програмному забезпеченні може допомогти у пошуку можливих шляхів покращення якості тестування. Інструменти штучного інтелекту використовують алгоритми машинного навчання для виявлення аномалій та патернів, які можуть бути непоміченими вручну або стандартними методами тестування. Раннє виявлення проблеми забезпечує більш ефективну обробку. Виявлення аномалій може служити індикатором можливих виробничих проблем або ситуацій, які можуть виникнути в життєвому циклі програмного забезпечення. Тому важливим є методи підвищення точності виявлення візуальних аномалій. У цій статті досліджуються методи побудови та навчання нейронних мереж з використанням глибокого навчання та технологій комп'ютерного зору для розпізнавання зображень. Розглядаються теоретичні основи глибокого навчання та комп'ютерного зору, проводиться аналіз існуючих підходів та архітектур нейронних мереж, застосовуваних у цій галузі. Описується методологія підготовки даних, вибору архітектури мережі та навчання моделі. Наведено результати практичної реалізації, включаючи аналіз точності та ефективності розробленої моделі. Проводиться порівняльний аналіз двох підходів до вирішення задачі розпізнавання зображень: детектування з використанням моделі YOLOv8 та семантичної сегментації із застосуванням архітектури U-Net. Наприкінці обговорюються висновки та перспективи подальших досліджень у галузі розпізнавання зображень.

Ключові слова: *алгоритми машинного навчання, візуальні елементи, виявлення аномалій, штучний інтелект, нейромережі, збої рендерингу.*

Постановка проблеми. Використання алгоритмів штучного інтелекту (ШІ), таких як машинне навчання та обробка природної мови, відкриває потенціал для підвищення точності та ефективності перевірки інформації у програмному забезпечені. Один із підходів до проблеми перевірки тверджень заснований на використанні нейронних мереж, зокрема рекурентних нейронних мереж (Recurrent neural networks (далі – RNNs)). RNNs є потужним інструментом для аналізу послідовностей даних і дозволяє виявляти помилкові. Дослідження [1], показало, що RNNs ефективні у виявленні візуальних аномалій.

Нейронні мережі можуть бути використані для автоматичного виявлення візуальних аномалій. Згорткові нейронні мережі (CNNs). CNNs − це один із видів нейронних мереж, які успішно використовуються для обробки тексту. Вони зазвичай використовуються для аналізу коротких текстів, таких як твіти. Їх особливістю є здатність виявлення патернів та взаємозв'язків між словами в тексті, що робить їх ефективними у виявленні хибної інформації у коротких текстах. Рекурентні нейронні мережі (RNN). RNNs − це інший вид нейронних мереж, призначених для обробки послідовних даних, як-от текст. Вони краще справляються з довшими текстами, такими як статті новини, завдяки здатності «розуміти» контекст і залежності між словами та пропозиціями.

RNNs дозволяє автоматично аналізувати текстові дані та їх ідентифікувати. Інший підхід полягає у використанні методів, що базуються на знаннях. Ці методи використовують великі бази знань, такі як Всесвітня павутина, для підтримки процесу перевірки [2]. Бази знань містять інформацію, яка може бути використана для зіставлення та перевірки фактів та тверджень. Візуальне тестування може допомогти виявити такі

проблеми, як несправність у відображенні графіки елементів, несумісність з різними дозволами екрану, некоректне відображення на різних пристроях, і т. д. Враховуючи швидкі темпи розвитку технологій, важливо постійно вдосконалювати візуальні методи та засоби тестування для забезпечення якості програмного забезпечення та задоволеності користувачів, що робить розпізнавання візуальних дефектів дуже важливим аспектом.

Аналіз останніх досліджень і публікацій. В останні роки дослідження в галузі глибокого навчання та комп'ютерного зору значно просунулися завдяки роботам провідних вчених та розробників. У 2012 році команда під керівництвом Алекса [3] представила модель AlexNet, яка досягла вражаючих результатів на змаганні ImageNet Large Scale Visual Recognition Challenge (ILSVRC) і започаткувала широку застосовність згорткових нейронних мереж (CNN) для розпізнавання зображень. Згодом з'явилися більш складні та ефективні архітектури, як VGG [4], ResNet [5], Inception [6] та EfficientNet [7], кожна з яких вносила свій внесок у підвищення точності та продуктивності моделей.

Гібридні методики розпізнавання аномалій, дозволяють поєднувати переваги різних підходів. При цьому різні техніки можуть застосовуватись як послідовно, так і паралельно для досягнення усереднених результатів. Прикладами гібридних систем розпізнавання аномалій можуть бути такі дослідження:

- Поєднання кластеризації та алгоритму найближчого сусіда у роботі [8].
- Паралельне використання суміщених алгоритмів Байєсових мереж та вирішальних дерев, а також алгоритму найближчого сусіда з класифікацією на основі правил у роботі [9].
- Поєднання методу опорних векторів та нейронної мережі глибинного навчання у роботі [10].

На основі аналізу літературних джерел, можна зробити такі висновки:

– для детектування мережевих аномалій використовуються як алгоритми машинного навчання (Decision Trees, Support Vector Machines, Nearest Neighbor Search, Random Forest, Logistic Regression, KNN, Adaptive Boosting), так і технології глибокого навчання (Convolutional Neural Network, Recurrent Neural Networks, Long Short-term Memory) [11–15];

– відкритими наборами даних, які найчастіше використовуються для навчання та тестування систем виявлення мережевих аномалій є CTU [16], NSL-KDD [17], UNSW-NB15B [18], CSE-CIC IDS-2018, bot-IoT [19].

Джерелами даних для створення датасетів для мережевих IDS-систем є пакети (парсинг пакетів, аналіз корисного навантаження), logфайли журналу з урахуванням вікон сеансу [20]. У потокових IDS (рис. 1) замість перегляду всіх пакетів, що проходять через мережеве з'єднання, збирається агрегована інформацію про пов'язані пакети мережного трафіку у вигляді потоку, до аналізу якого застосовуються методи вилучення візуальних аномалій з нейронних мереж глибокого навчання.

Кожен потік містить агреговану інформацію про кількість переданих пакетів та байтів, IP-адреси та порти джерела та призначення, тимчасові мітки, прапори TCP, мережні маски та інші параметри. У ході експерименту [22] було зібрано навчальні вибірки з набору даних CTU для 5 сценаріїв ботнет атак на програмне забезпечення. На першому етапі для отримання ознак аномальних та нормальних станів мережі використана згорткова нейронна мережа, реалізована в вигляді Python-додатки для інтерактивних обчислень (файл формату ipynb), містять вихідний код, вхідні дані, результати обчислень у числовому та

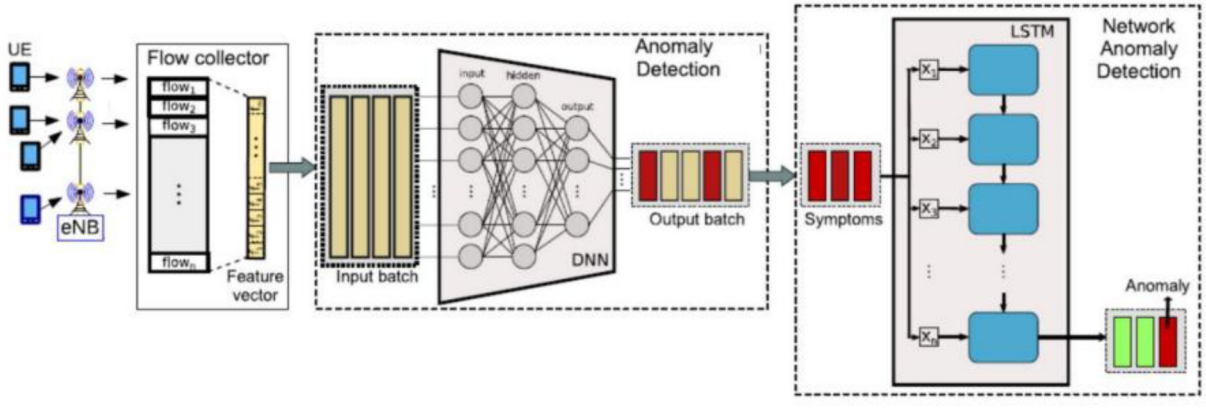

Рис. 1. Потокова IDS-система, адаптована для стільникових мереж [21]

графічне представлення. На другому етапі розпізнавання мережевих аномалій реалізовано з використанням алгоритмів машинного навчання.

Сучасні веб-системи для виявлення аномалій стали важливим інструментом у багатьох сферах, де необхідно обробляти великі обсяги даних в реальному часі. Веб-системи дозволяють користувачам швидко інтегрувати алгоритми машинного навчання, такі як Isolation Forest, LOF та DBSCAN, у свої робочі процеси без необхідності глибоких знань у програмуванні. У останні роки активно розвиваються методи, що використовують глибинне навчання для виявлення аномалій. Сучасні дослідження орієнтовані на вдосконалення традиційних алгоритмів, таких як Isolation Forest або Local Outlier Factor (LOF), оскільки вони мають обмеження у роботі з великими, складними та багатовимірними наборами даних. Крім того багато нових робіт орієнтуються на використання нейронних мереж, зокрема автокодерів (autoencoders), для виявлення аномалій, що дозволяє значно покращити точність виявлення.

Постановка завдання. Метою статті є аналіз методів підвищення точності розпізнавання візуальних дефектів у ПЗ за допомогою ШІ

Виклад основного матеріалу.

З огляду літератури можна виділити основні типи моделей:

- Convolutional Neural Networks (CNN) для обробки зображень.
- Autoencoders та Variational Autoencoders (VAE) для виявлення аномалій.
- Generative Adversarial Networks (GANs) для синтезу еталонних зображень та оцінки аномалій.
- Vision Transformers (ViT) для точного аналізу структурованих візуальних даних.

У дослідженні використовувалася нейронна мережа (CNN), заснована на архітектурі, запропонованій у роботі LeNet-5, та подальших модифікаціях. CNN є потужним інструментом для аналізу зображень (рис. 2), дозволяючи ефективно отримувати та узагальнювати ознаки із вхідних даних. Архітектура CNN включає кілька згорткових шарів з функціями активації ReLU, шари об'єднання (pooling), повнозв'язкові шари та шари класифікації softmax.

Як матеріали для навчання та тестування моделі використовували набір даних MNIST, що складається з 60 000 тренувальних зображень рукописних цифр та 10 000 тестових зображень. Зображення представлені чорно-білими розмірами 28 × 28 пікселів.

1. Завантаження даних: Набір даних MNIST був завантажений за допомогою бібліотеки TensorFlow.
2. Передобробка даних: Зображення були нормалізовані (наведені до діапазону [0,1]) та змінені на розмір 28 × 28 пікселів для відповідності вхідним даним моделі.
3. Створення моделі: Була побудована згорткова нейронна мережа з використанням бібліотеки TensorFlow та її високорівневого API Keras. Модель включає згорткові шари, шари об'єднання, повнозв'язкові шари та шар класифікації softmax.
4. Компіляція моделі: Модель була скомпільована з використанням оптимізатора Adam та функції втрат категоріальної крос-ентропії.
5. Навчання моделі: Модель була навчена на тренувальних даних протягом 5 епох з розміром пакета 64.
6. Оцінка моделі: Продуктивність моделі була оцінена на тестових даних шляхом обчислення точності класифікації.

Цей порядок дій забезпечує побудову, навчання та оцінку моделі для завдання розпізнавання візуальних аномалій на наборі даних MNIST. Приклад коду на Python, який створює та навчає нейронну мережу для розпізнавання візуальних аномалій із набору даних MNIST, використовуючи бібліотеку TensorFlow:

```python
import tensorflow as tf
від tensorflow.keras import layers, models
```

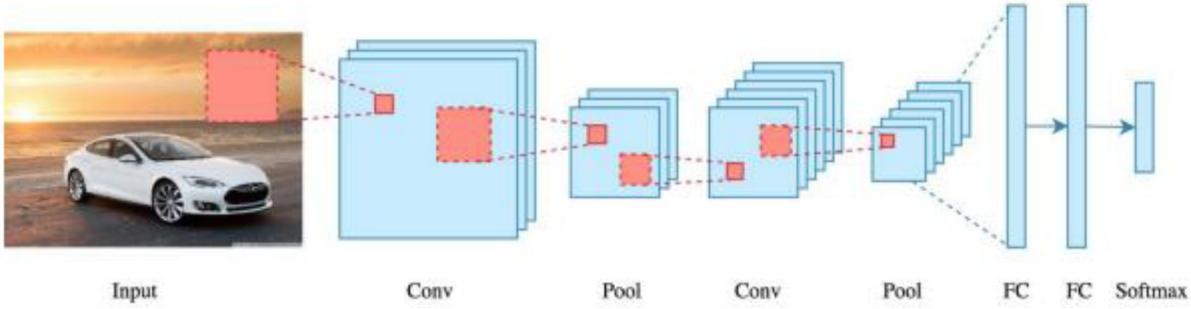

Рис. 2. Архітектура CNN

```
from tensorflow.keras.datasets import mnist
# Завантаження даних
(train_images, train_labels), (test_images,
test_labels) = mnist.load_data()
# Передобробка даних
train_images = train_images.reshape((60000,
28, 28, 1))
train_images = train_images.
astype('float32') / 255
test_images = test_images.reshape((10000,
28, 28, 1))
test_images = test_images.
astype('float32') / 255
# Перетворення тегів у категоріальний формат
train_labels = tf.keras.utils.to_
categorical(train_labels)
test_labels = tf.keras.utils.to_
categorical(test_labels)
# Створення моделі нейронної мережі
model = models.Sequential([
layers.Conv2D(32, (3, 3), activation='relu',
input_shape=(28, 28, 1)),
layers.MaxPooling2D((2, 2)),
layers.Conv2D(64, (3, 3),
activation='relu'),
layers.MaxPooling2D((2, 2)),
layers.Conv2D(64, (3, 3),
activation='relu'),
layers.Flatten(),
layers.Dense(64, activation='relu'),
layers.Dense(10, activation='softmax')
])
# Компіляція моделі
model.compile(optimizer='adam',
loss = 'categorical_crossentropy',
metrics=['accuracy'])
# Навчання моделі
model.fit(train_images, train_labels,
epochs=5, batch_size=64, validation_
split=0.2)
# Оцінка моделі на тестових даних
test_loss, test_acc = model.evaluate(test_
images, test_labels)
print('Test accuracy:', test_acc)
```

Одним із обмежень даного дослідження є використання тільки одного набору даних MNIST. Хоча MNIST є стандартним набором даних для навчання моделей розпізнавання аномалій, його відносна простота та невеликий розмір можуть не повністю відбивати складності реальних завдань. Для більш повного розуміння продуктивності моделі та її застосовності до реальних сценаріїв необхідно провести експерименти більш різноманітних і складних наборах даних. Також варто зазначити, що, можливо, існують більш сучасні архітектури нейронних мереж, які можуть демонструвати ще вищі результати під час вирішення цієї задачі.

У цій роботі було успішно описано нейронну мережу (CNN) для розпізнавання візуальних аномалій із набору даних MNIST. Досягнута точність моделі на тестовому наборі даних склала 99.1 %, що свідчить про високу продуктивність і надійність запропонованого підходу. Представимо основні методи виявлення аномалій.

1. *Статистичні випробування*. Як правило, їх застосовують для окремих ознак і виділяють екстремальні значення (Extreme-Value Analysis). Для цього використовують, наприклад, *Z*-value [23]:

$$Z_i = \frac{x_i - \mu}{\delta}.$$

2. *Модельні тести*. Ідея дуже проста: будуємо модель, яка описує дані; точки, які сильно відхиляються від моделі (на яких модель сильно помиляється), і є аномалії. При виборі моделі можна врахувати природу завдання, функціонал якості тощо. Наприклад, у досліджуваному завданні можна прогнозувати значення тимчасових рядів з допомогою LSTM-нейронної мережі [24]. Якщо реальні значення сильно відрізняються від передбачуваних, то це свідчить про аномальну поведінку.

3. *Ітераційні методи*. Можна послідовно видаляти групи «особливо підозрілих об'єктів». Наприклад, у n-мірному ознаковому просторі можна видаляти опуклу оболонку точок-об'єктів. Як правило, методи цієї групи досить трудомісткі.

4. *Методи машинного навчання*. Завдання виявлення аномалій розглядають також як окреме завдання навчання без вчителя (unsupervised learning).

5. *Ансамблі алгоритмів*. Як і в багатьох інших областях машинного навчання, при пошуку аномалій часто використовують кілька алгоритмів зазвичай різної природи.

Методи, які вирішують проблему виявлення аномалій у багатовимірних і багатовимірних даних, підсумовані в таблиці 1. Кожен метод має переваги та недоліки, коли потрібно виправляти різні проблеми залежно від характеру даних [25].

Сучасні моделі глибокого навчання, такі як YOLOv8 [26] та U-Net [27], зарекомендували себе як потужні інструменти у завданнях класифікації, детектування та сегментації зображень.

YOLOV8 (You Only Look Once) є однією з передових архітектур для детектування об'єктів,

Таблиця 1

Порівняння продуктивності алгоритмів виявлення аномалій [25]

Алгоритм	Переваги	Недоліки
PCA	широко використовується завдяки простоті та ефективності	з великою розмірністю оцінка зазвичай складна; наявність аномалії може вплинути на продуктивність PCA
DOBIN	дозволяє виявити аномалію за допомогою меншої кількості компонентів	Чутливий
ROBEM	для виявлення аномалії використовується критичне значення; таким чином, це призводить до успішної продуктивності щодо виявлення аномалій	найповільніший алгоритм
DAE-KNN	знижує обчислювальні витрати та покращує ефективність виявлення порівняно з одним детектором аномалій	побудова DAE займає багато часу, якщо набір даних великий

забезпечує високу продуктивність та точність. U-Net, у свою чергу, є класичною архітектурою для семантичної сегментації, яка демонструє чудові результати під час аналізу складних візуальних даних (рис. 3).

Для нашого проекту було обрано енкодер Res-Net-18, це версія енкодера з 18 шарами яка добре підходить для отримання ознак із зображення. Архітектура мережі U-Net з енкодером Res-Net-18 представлена на рис. 4.

Як початкові ваги для енкодера отримані на наборі даних ImageNet. Це дозволить прискорити навчання та підвищити продуктивність моделі.

Як функція активації для вихідного шару «softmax2d», яка застосовується до двовимірних даних та активно використовується у завданнях сегментації. Ця функція перетворює вихідні значення нейронної мережі у ймовірності для кожного класу, що дозволяє інтерпретувати результати як ймовірність приналежності кожного пікселя до певного класу, У контексті завдання сегментації це дозволить моделі передбачати, до якого класу належить кожен піксель зображення.

Для двовимірного масиву Z розміром $H \times W \times C$ (де H – висота, W – ширина, C – кількість класів), функція втрат «softmax2d» визначається як:

$$softmax(Z_{i,j,k}) = \frac{e^{z_{i,j,k}}}{\sum_{c=1}^{c} e^{z_{i,j,c}}},$$

де $e^{z_{i,j,k}}$ – логіт (не нормалізоване передбачення) для пікселя (i, j) та класу k. Сума у знаменнику

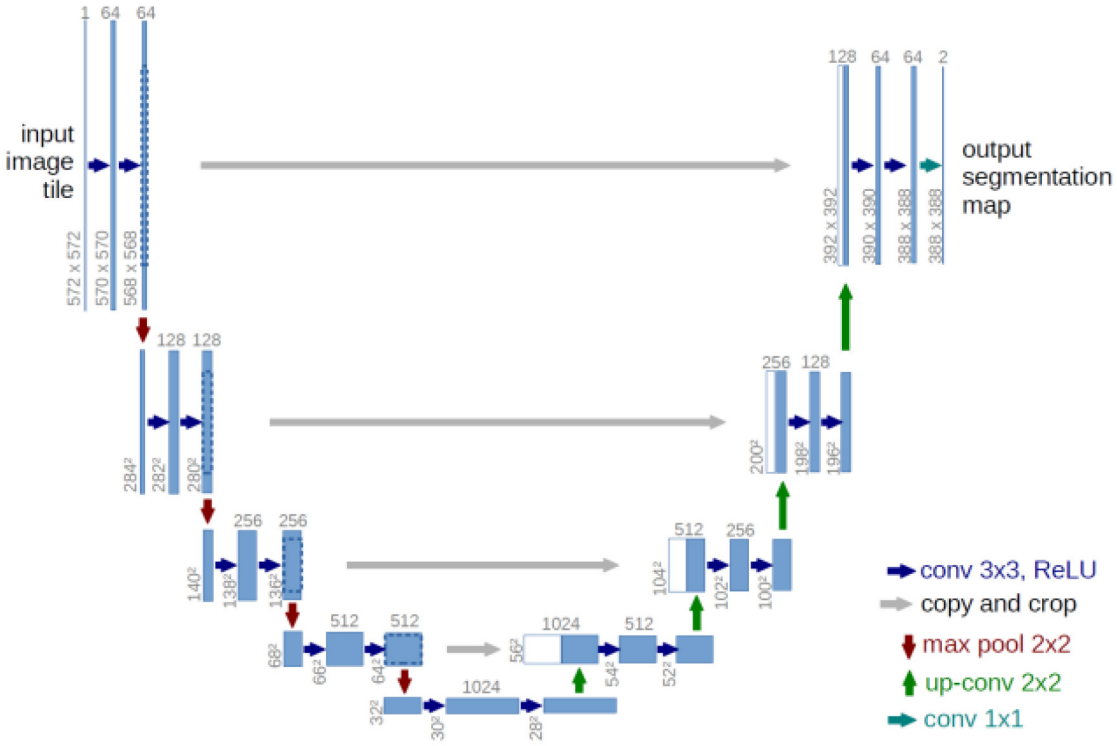

Рис. 3. Архітектура U-Net [27]

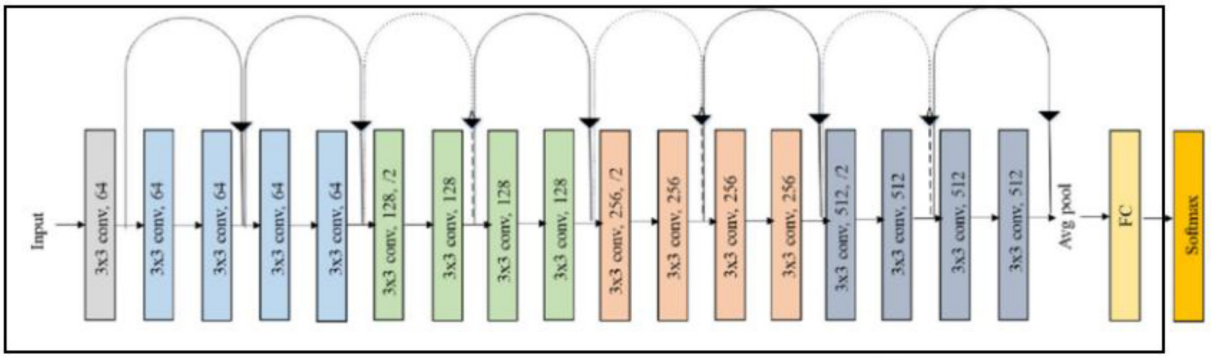

Рис. 4. Архитектура Res-Net-18

береться за всім класам c для кожного пікселя (i, j).

Для відстеження ефективності навчання та роботи мережі U-Net використовуються такі метрики:

1. F1-score.
2. IoU.

Дані метрики активно застосовують при задачах сегментації. Розглянемо кожну метрику докладніше. F1-score – це гармонійне середнє між точністю (precision) та повнотою (recall). Ця метрика особливо корисна у завданнях, де важливо враховувати як хибнопозитивні, так і хибнонегативні результати, наприклад, у задачах класифікації з дисбалансу класів. Ця метрика визначається як:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}.$$

$$Precision = \frac{TR}{TR + FP} \quad Recall = \frac{TR}{TR + FN},$$

де TP – кількість істинно позитивних прогнозів (True Positives), FP – кількість хибнопозитивних (False Positives), FN – кількість хибнонегативних передбачень (False negatives).

$F1$-score приймає значення від 0 до 1, де означає 1 ідеальна відповідність (максимальна точність та повнота), а 0 – відсутність відповідності. IoU – коефіцієнт перетину та об'єднання, який показує, як перетин між передбачуваною областю та істинною областю відноситься до їхнього об'єднання. Обчислюється за такою формулою:

$$IoU = \frac{|A \cap B|}{|A \cup B|},$$

де A – область, передбачена моделлю, B – істинна область

Метрика IoU лежить у діапазоні [0, 1] і чим більше її значення, тим сильніше збігаються передбачена і справжня маска.

Проведення дослідження на двох нейронних мережах відмінних по архітектурі та підході до навчання дозволило зробити висновок, що YOLOv8 має ряд переваг: зручність використання, відсутність необхідності в детальному налаштуванні шарів, низький поріг входження. Що стосується U-Net, вона має великий потенціал: детальне налаштування як архітектури так і параметрів. Не можна не відзначити при цьому, що поріг входження набагато вищий, потрібно більш детальна робота із даними.

Висновки.

1. Ефективність архітектури CNN: Використання кількох згорткових шарів дозволяє ефективно витягувати та узагальнювати ознаки зображень, що призводить до високої точності розпізнавання.

2. Значення передобробки даних: Нормалізація зображень та збільшення даних (аугментація) відіграє важливу роль у покращенні якості вхідних даних та сприяє кращому навчанню моделі.

3. Оптимізація гіперпараметрів: Правильний вибір гіперпараметрів, таких як розмір кроку навчання, кількість шарів та фільтрів, використання методів регуляризації, дозволяє досягти кращих результатів.

4. Як модель для детекції була обрана YOLOV8, для семантичної сегментації – U-Net. Для кожної з моделей були розглянуті: архітектура, параметри, процес навчання, тип даних та формат анотацій

Хоча досягнуті результати є значними, існує кілька напрямів для подальших досліджень та покращень.

1. Застосування до складніших наборів даних: У майбутньому доцільно протестувати та адаптувати модель для роботи з більш складними та різноманітними наборами даних, такими як CIFAR-10, CIFAR-100 та ImageNet, щоб оцінити її узагальнюючу здатність.

2. Поліпшення архітектури: Дослідження більш сучасних та складних архітектур CNN, таких як ResNet, DenseNet та EfficientNet, може призвести до подальшого підвищення точності та ефективності.

3. Автоматизація підбору гіперпараметрів: Використання методів автоматизованого підбору гіперпараметрів (AutoML) та байєсівської оптимізації може прискорити процес налаштування моделі та привести до покращення її продуктивності.

4. Застосування методів ансамблювання: Об'єднання кількох моделей в ансамбль може підвищити точність та стійкість рішення, особливо в умовах різноманітних і зашумлених даних.

5. Оптимізація обчислювальних витрат: Дослідження методів зменшення обчислювальних витрат та прискорення навчання, таких як розподілене навчання та використання спеціалізованих апаратних рішень (наприклад, TPU), може зробити глибоке навчання більш доступним та ефективним. Таким чином, це дослідження підтверджує ефективність використання глибокого навчання та технологій комп'ютерного зору для розпізнавання зображень.

Список літератури:

1. Smith, J., Johnson, A. Leveraging Artificial Intelligence and Machine Learning for Network Anomaly Detection: A Comprehensive Review. Journal of Cybersecurity Advances, 2023, 5(2), 123–145.

2. An enhanced AI-based network intrusion detection system using generative adversarial networks. Park, C., Lee, J., Kim, Y., et al. IEEE Internet of Things Journal, 2020, 10(3), 2330–2345.

3. Alexe, B., Deselares, T., Ferrari, V. Measuring the objectness of image windows. In PAMI, 2012.

4. Simonyan, K., Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv 2014, arXiv:1409.1556.

5. He, K., Zhang, X., Ren, S., Sun, J. Deep residual learning for image recognition. In: CVPR, 2016.

6. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In CVPR, pages 1–9, 2015.

7. Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., and Le, Q. V. MnasNet: Platform-aware neural architecture search for mobile. CVPR, 2019.

8. Lin, W-C. Ke, S-W. Tsai, C-F. "An intrusion detection system based on combining cluster centers and nearest neighbors", KnowledgeBased Systems, 2015, vol. 78, pp. 13–21.

9. Pinto, S. J., Siano, P., Parente, M. Review of cybersecurity analysis in smart distribution systems and future directions for using unsupervised learning methods for cyber detection. Energies, 2023, 16(4), 1651.

10. Tange K., De Donno M., Fafoutis X. Dragoni N. A Systematic Survey of Industrial Internet of Things Security: Requirements and Fog Computing Opportunities. EEE Communications Surveys & Tutorials, 2020, V. 22(4), 2489–2520.

11. Kilincer I. F., Ertam F., Sengur A. Machine learning methods for cyber security intrusion detection: Datasets and comparative study. Computer Networks, 2021, V. 188, P. 107840.

12. Kumar R. SP2F: A secured privacy-preserving framework for smart agricultural Unmanned Aerial Vehicles. Computer Networks, 2021, V. 187, P. 107819.

13. Van Eck, N.; Waltman, L. Manual for VOS Viewer Version 1.6.10, CWTS, Universiteit Leiden, Leiden, Holland, 2019.

14. Pang, G., Shen, C., Cao, L., Hengel, A.V.D. Deep learning for anomaly detection: A review. ACM computing surveys (CSUR), 2021, 54(2), 1–38.

15. CNN features with bi-directional LSTM for real-time anomaly detection in surveillance networks. Ullah, W., Ullah, A., Haq, I. U. et. al. Multimedia Tools and Applications, 2021, 80(11), 16979–16995.

16. Komar, M., Fedorovych, V., Poidych, V., Taborovskyi, A. Intelligent System For Visual Testing Of Software Products. The First International Workshop of Young Scientists on Artificial Intelligence for Sustainable Development; May 10–11, 2024, Ternopil, Ukraine, 2024.

17. NSL-KDD99 Dataset. 2009. Available online: https://www.unb.ca/cic/datasets/nsl.html (accessed on 06 March 2025).

18. Komar, M., Kochan, V., Dubchak, L., Sachenko, A., Golovko, V., Bezobrazov, S., Romanets, I. High performance adaptive system for cyber attacks detection. Proceedings of the 2017 IEEE 9th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2017, 2017, 2, pp. 853–858.

19. Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J. Jansen, Kam-Fai Wong, and Meeyoung Cha. Detecting rumors from microblogs with recurrent neural networks. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16). AAAI Press, 2016, 3818–3824.

20. A detection method for anomaly flow in software-defined networks. Peng, H., Sun, Z., Zhao, X., et. al. IEEE Access, 2018, 6, 27809–27817.

21. Wang, C., Wang, B., Liu, H., Qu, H. Anomaly detection for industrial control system based on autoencoder neural network. Wireless Communications and Mobile Computing, 2020, 1, 8897926.

22. Verbal and Nonverbal Clues for Real-life Deception. Pérez-Rosas, V., Abouelenien, M., Mihalcea, R. Detection, 2015, 2336–2346. https://doi.org/10.18653/v1/D15-1281

23. Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends, and new perspectives. Himeur, Y., Ghanem, K., Alsalemi, A., at. al. Applied Energy, 2021, 287, 116601.

24. Deep learning-enabled anomaly detection for IoT systems. Abusitta, A., de Carvalho, G. H. S., Wahab, O. A., at. al. Internet of Things, 21, 100656.

25. Kim, J., Lee, H., Park, J. AI-based anomaly detection over encrypted traffic: A systematic review. Scientific Reports, 2023, 13(1), 12345.

26. "YOLOv8 Models Documentation", available at: https://github.com/ultralytics/ultralytics/blob/main/docs/en/models/yo lov8.md (Accessed: 02.03.2025).

27. "U-Net: Convolutional Networks for Biomedical Image Segmentation", available at: https://lmb.informatik.unifreiburg.de/people/ronneber/u-net/ (Accessed: 02.03.2025).

Fedorovych V. I. METHODS FOR IMPROVING THE ACCURACY OF DETECTING VISUAL ANOMALIES IN SOFTWARE BASED ON ARTIFICIAL INTELLIGENCE

Anomaly detection is an important feature of artificial intelligence in automated testing. Artificial intelligence, in particular machine learning and neural network methods, are widely used to automate the testing process and improve the efficiency of error detection. For example, intelligent algorithms can be used to determine the optimal ways to conduct program testing, as well as to predict possible problems in the operation of the program, which allows to reduce the number of errors and reduce the time required for software testing. In this context, research into artificial intelligence methods for detecting visual anomalies in software can help in finding possible ways to improve the quality of testing. Artificial intelligence tools use machine learning algorithms to detect anomalies and patterns that may be unnoticed by manual or standard testing methods. Detect problems early and provide more efficient processing. Anomaly detection can serve as an indicator of possible production problems or situations that may arise in the software life cycle. Therefore, methods to improve the accuracy of visual anomaly detection are important. This article explores methods for building and training neural networks using deep learning and computer vision technologies for image recognition. The theoretical foundations of deep learning and computer vision are considered, and existing approaches and architectures of neural networks used in this field are analyzed. The methodology for data preparation, network architecture selection, and model training is described. The results of practical implementation are presented, including an analysis of the accuracy and efficiency of the developed model. A comparative analysis of two approaches to solving the image recognition problem is conducted: detection using the YOLOv8 model and semantic segmentation using the U-Net architecture. Finally, conclusions and prospects for further research in the field of image recognition are discussed.

Key words: *machine learning algorithms, visual elements, anomaly detection, artificial intelligence, neural networks, rendering glitches.*