УДК 004.932

Сапожнікова В. Є., здобувачка вищої освіти 4 курсу,
Ніколюк П. К., д-р фіз.-мат. наук,
професор кафедри інформаційних технологій,
Донецький національний університет імені Василя Стуса

РОЗРОБКА НЕЙРОМЕРЕЖЕВОЇ СИСТЕМИ ДЕТЕКЦІЇ ВОГНЮ НА ОСНОВІ МОДЕЛІ MOBILENETV2

Анотація. У роботі реалізовано систему класифікації зображень для виявлення вогню з використанням глибокої згорткової нейронної мережі MobileNetV2. Створено власний генератор даних BalancedDataGenerator, що забезпечує балансування класів і поділ на тренувальну та валідаційну вибірки. Навчання виконано в два етапи: базове тренування верхніх шарів і подальше тонке донавчання останніх шарів базової моделі. Отримана модель досягла високої точності класифікації на валідаційних даних.

Ключові слова: MobileNetV2, комп'ютерний зір, класифікація зображень, вогонь, TensorFlow.

Вступ. Сучасні системи відеоспостереження дедалі частіше використовуються для забезпечення пожежної безпеки. Однак традиційні датчики диму або температури не завжди ефективні у відкритих або великих приміщеннях. Тому зростає актуальність застосування методів штучного інтелекту для виявлення вогню безпосередньо на зображеннях чи відео [1].

Метою роботи є розробка нейромережевої системи детекції вогню, яка поєднує високу точність розпізнавання та низькі обчислювальні витрати, що дає можливість її використання на мобільних або вбудованих пристроях у режимі реального часу.

Основний текст. Для вирішення задачі обрано архітектуру MobileNetV2, відому ефективним поєднанням точності та швидкодії [2]. У системі реалізовано власний генератор BalancedDataGenerator, який забезпечує автоматичний поділ вибірки та балансування класів. Основна логіка його роботи наведена нижче.

```python
class BalancedDataGenerator(Sequence):
    def __getitem__(self, idx):
        batch_images = self.images[idx*self.batch_size:(idx+1)*self.batch_size]
        batch_labels = self.labels[idx*self.batch_size:(idx+1)*self.batch_size]
        X = []
        for img_path in batch_images:
            img = load_img(img_path, target_size=(self.img_size, self.img_size))
            img = img_to_array(img)/255.0
            X.append(img)
        X = np.array(X)
        y = np.array(batch_labels)
        return X, y
```

Рисунок 1 – Фрагмент коду генератора даних BalancedDataGenerator

Генератор реалізовано на основі класу Sequence бібліотеки Keras, що дає змогу передавати дані батчами без завантаження всього набору в пам'ять. Це забезпечує гнучкість під час роботи з великими зображеннями та рівномірне подання прикладів обох класів.

```
base_model = MobileNetV2(weights="imagenet", include_top=False, input_shape=(IMG_SIZE, IMG_SIZE, 3))
x = GlobalAveragePooling2D()(base_model.output)
x = Dense(64, activation="relu")(x)
output = Dense(1, activation="sigmoid")(x)
model = Model(inputs=base_model.input, outputs=output)
```

Рисунок 2 – Фрагмент коду побудови моделі на базі MobileNetV2

Після побудови моделі базову частину мережі *MobileNetV2* було заморожено (base_model.trainable = False), що дало змогу спочатку навчати лише класифікаційний блок на верхніх шарах. Такий підхід запобігає перенавчанню й допомагає зберегти узагальнювальні властивості попередньо натренованої моделі.

Після початкового етапу навчання виконано етап Fine-Tuning (FT), під час якого частину верхніх шарів базової моделі було розморожено (base_model.trainable = True для останніх 30 шарів). Це дало змогу адаптувати вже наявні ваги MobileNetV2 до специфічних ознак зображень пожежі та диму в нашому наборі даних [3].

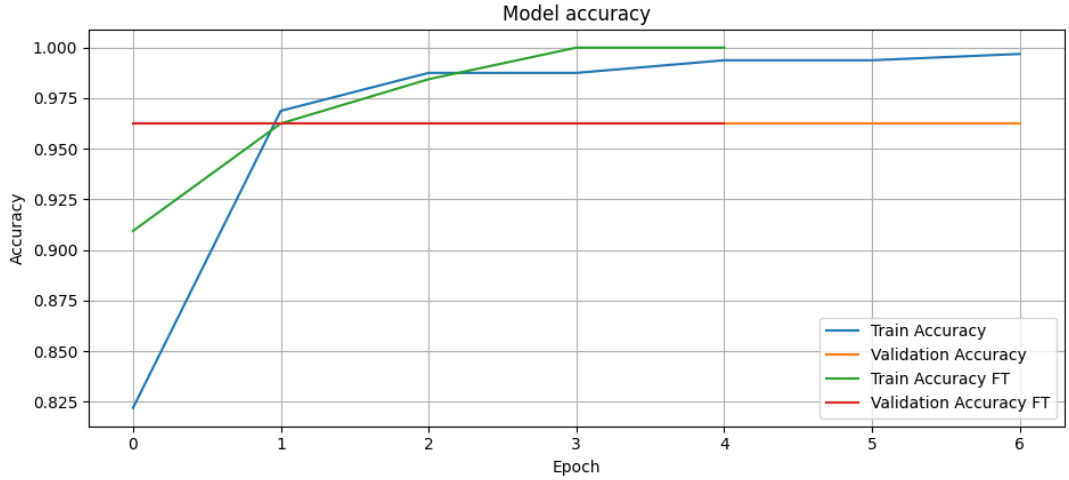

Рисунок 3 – Графік точності навчання та Fine-Tuning

Видно, що після застосування FT точність моделі на валідаційній вибірці зросла, а крива помилки стабілізувалася після 2–3 епох, що свідчить про досягнення стійкої збіжності. На фінальному етапі середня точність на валідації становила приблизно 0.98, що є показником якісної генералізації моделі за відсутності перенавчання.

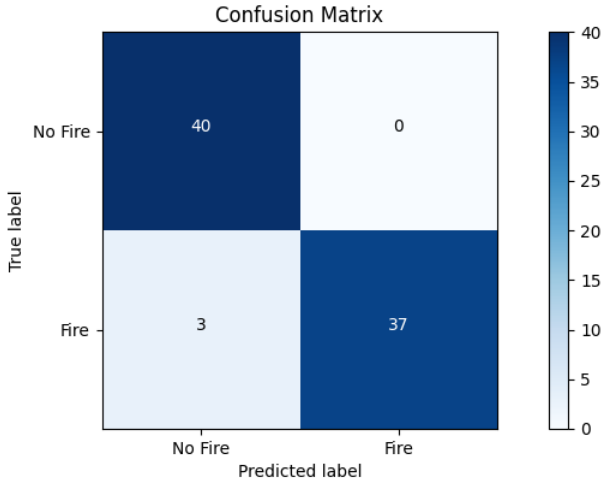

Рисунок 4 – Матриця плутанини класифікатора виявлення пожежі

Для кількісного підтвердження ефективності навчання було побудовано матрицю плутанини (рис. 4), що відображає результати класифікації на валідаційному наборі. Модель правильно розпізнала 40 зображень класу No Fire та 37 – класу Fire, припустившись лише трьох хибнонегативних спрацьовувань.

```
Classification Report:

              precision    recall  f1-score   support

     No Fire       0.93      1.00      0.96        40
        Fire       1.00      0.93      0.96        40

    accuracy                           0.96        80
   macro avg       0.97      0.96      0.96        80
weighted avg       0.97      0.96      0.96        80
```

Рисунок 5 – Звітність точності моделі

Отримані метрики підтверджують збалансовану роботу класифікатора. Значення F1-міри 0.96 для обох класів демонструє стабільність результатів та відсутність перенавчання.

Після навчання найкраща версія мережі була автоматично збережена у файл fire_detection_model.h5 і готова до використання в системах розпізнавання пожежі. Для перевірки її роботи створено окремий скрипт, який зчитує зображення з каталогу, визначає очікуваний клас за назвою файла та порівнює його з прогнозом моделі.

```
Всього зображень: 199
Вгадано правильно: 196 (98.49%)
Вогонь: 155/155 (100.00%)
Без вогню: 41/44 (93.18%)
```

Рисунок 6 – Результати тестування моделі

На незалежному наборі з 199 зображень, що не використовувався під час навчання, модель показала загальну точність 98.49 %, підтверджуючи її високу здатність до генералізації (рис. 6). Це свідчить про високу здатність моделі узагальнювати нові зображення й надійно відрізняти кадри з полум’ям від звичайних сцен без вогню.

Висновки. Розроблена модель на основі *MobileNetV2* із застосуванням підходу Transfer Learning + Fine-Tuning забезпечує високу ефективність виявлення пожежі на зображеннях у режимі реального часу.

Заморожування базових шарів на першому етапі дало змогу зберегти універсальні ознаки з ImageNet, а донавчання останніх шарів під час FT адаптувало модель до специфічних особливостей вибірки.

Результатом реалізації є навчена модель fire_detection_model.h5, що забезпечує автоматичне розпізнавання пожежі на статичних зображеннях із точністю понад 98 %.

Список використаних джерел

1. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications / A. Howard et al. URL: https://arxiv.org/abs/1704.04861 (дата звернення: 25.10.2025).

2. Chollet F. Deep Learning with Python, , Second Edition. Manning Publications, 2018. URL: https://books.google.com/books?id=XHpKEAAAQBAJ (дата звернення: 25.10.2025).

3. TensorFlow Documentation: MobileNetV2 and Transfer Learning Guides [Online]. URL: https://www.tensorflow.org/tutorials/images/transfer_learning (дата звернення: 25.10.2025).