

ЗАСТОСУВАННЯ ЗГОРТКОВОЇ НЕЙРОННОЇ МЕРЕЖІ З ДЕКІЛЬКОМА ВИХОДАМИ ДЛЯ РОЗПІЗНАВАННЯ ОБІГОВИХ МОНЕТ

Проблема розпізнавання обігових монет може постати як перед досвідченими колекціонерами, так і перед звичайними туристами або колекціонерами-початківцями. Найлегшим рішенням для тих, хто не має достатньо знань для самостійного розпізнавання монет, може стати розпізнавання за допомогою програмного засобу, побудованого на основі нейронної мережі. Оскільки більшість характеристик монет можна визначити за зображеннями аверсу та реверсу а сам процес створення фотографій для розпізнавання є швидким та не потребує додаткових вимірювальних приладів, найзручнішим підходом для вирішення даної проблеми є розпізнавання монет за зображеннями.

У роботі розглянуто проблему розпізнавання обігових монет за допомогою згорткових нейронних мереж, які містять шари, призначені спеціально для роботи із зображеннями. Описано традиційний підхід до вирішення задачі розпізнавання зображень, який передбачає використання звичайної згорткової нейронної мережі з одним виходом, коли кожному зображенню поставлено у відповідність клас, який однозначно його визначає. Аналіз даного підходу та отриманих результатів показує що для вирішення поставленої задачі для зібраного набору даних, коли виникає досить велика кількість класів, така архітектура не є оптимальною. Для підвищення точності розпізнавання запропоновано архітектуру згорткової нейронної мережі з декількома виходами, яка передбачає розгалуження структури нейронної мережі на декілька гілок на певному етапі. При використанні такої нейронної мережі зображенням у відповідність наводиться перелік із декількох незалежних характеристик замість одного складеного класу. Таким чином задача розпізнавання монети розбивається на декілька підзадач, кожній із яких відповідає окрема гілка нейронної мережі. Також в роботі був проведений порівняльний аналіз результатів отриманих при застосуванні традиційної нейронної мережі та мережі із декількома виходами були визначені причини відмінностей у результатах та переваги і недоліки кожного із розглянутих підходів.

Ключові слова: згорткова нейронна мережа, нейронна мережа з декількома виходами, розпізнавання зображень, машинне навчання, штучний інтелект.

Вступ. Нумізмати́ка, як наука що вивчає історію монет та їх виготовлення, і, власне, колекціонування монет виникле ще за часів Древнього Риму і за весь час свого існування не втрачає своєї популярності. Навіть у наш час, коли стрімко розвивається безготівковий розрахунок, випуск металевих монет як для обігу, так і виключно для колекціонування не припиняється. Ситуації, в яких виникає необхідність розпізнати певну монету можуть виникнути і в колекціонерів, і в зацікавлених людей, і в звичайних туристів. Звичайно, що досвідчені колекціонери впораються власними силами, проте для початківців та інших осіб така задача може стати суттєвою проблемою. У такому випадку найлегшим рішенням є розпізнавання монети за допомогою програмних засобів, що нададуть користувачеві базову інформацію про монету. Серед усіх характеристик монети більшість, такі як номінал та грошова одиниця, країна та рік випуску, позначення монетних дворів та правителів можуть бути визначені за зображеннями аверсу та реверсу монети, на відміну від, наприклад, діаметру та ваги монети, визначення яких потребує додаткових інструментів. Оскільки фотографій аверсу та реверсу достатньо для визначення базових характеристик монет і створення таких фотографій не потребує багато зусиль та приладів, для користувачів найзручнішим підходом

є розпізнавання монет саме за цими зображеннями. На сьогодні найкращим підходом до розпізнавання зображень вважається використання згорткової нейронної мережі. У даній роботі представлено результати застосування нейронних мереж для розпізнавання монет на прикладі датасету зображень, що містить фотографії сучасних обігових монет деяких країн Європи.

Постановка задачі. Задача даного дослідження полягає у побудові згорткової нейронної мережі для розпізнавання обігових монет країн Європи. Задля пошуку найкращого рішення було розглянуто традиційний підхід до розпізнавання зображень та внесено зміни до моделі нейронної мережі з метою підвищення точності розпізнавання зображень з урахуванням особливостей вхідних даних.

Для якісного навчання будь-якої згорткової нейронної мережі необхідно забезпечити якомога більший тренувальний датасет розмічених зображень, тобто зображень, яким поставлені у відповідність певні класи. Для переважної більшості задач розпізнавання зображень неважко знайти готові датасети на спеціалізованих сайтах у мережі Інтернет, проте для даного дослідження, за відсутності готового датасету, виникає необхідність власноруч розмітити зображення, що знаходяться у вільному доступі у мережі Інтернет та власні фотографії.

Аналіз останніх досліджень. При виникненні задачі розпізнавання зображень зазвичай використовується традиційний підхід – використання звичайної згорткової нейронної мережі, яка була вперше описана Яном ЛеКуном у 1995 році [1]. Згорткова нейронна мережа приймає на вхід певне зображення та повертає вектор чисел, що відображають впевненість нейронної мережі у відповідності зображення кожному із заданих класів. При такому підході кожне зображення у тренувальному датасеті визначається одним і лише одним класом, який повинен повною мірою охарактеризувати дане зображення і визначити відмінності від інших. Для розпізнавання обігових монет кожному зображенню монети буде доречним поставити у відповідність характеристику, що містить номінал, грошову одиницю та країну монети. Таким чином будуть отримані класи виду «1 копійка, Україна» для кожного із зображень. Для країн Європи можна отримати декілька сотень таких класів, що при невеликій кількості зображень у датасеті призведе до досить малого обсягу зображень у кожному окремому класі.

Традиційна згорткова нейронна мережа, приклад якої подано на рис. 1, складається із послідовного чергування шарів згортки (convolution) із функцією активації ReLU, які являють собою набір фільтрів, що після проходження по зображенню формують мапи ознак для кожного фільтру, і агрегування (pooling), які виконують підвибірку і слугують для зменшення розділення мап ознак [2]. На виході такої мережі розміщено один або декілька звичайних повнозв'язних нейронних шарів для класифікації зображення.

Для більшості задач розпізнавання зображень звичним підходом є використання готової архітектури нейронної мережі, здебільшого обирають архітектури переможців щорічного змагання ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [3], під час якого оцінюються алгоритми виявлення об'єктів та розпізнавання зображень. Найчастіше використовують мережі VGG16 [4] або ResNet50 [5], що були навчені на наборі зображень ImageNet, і застосовують метод передавального навчання, описаний Стево Бозіновським [6], для переносу навчання з великого датасету ImageNet на менший. У даній роботі, з огляду на специфіку вхідних даних і обмеженість ресурсів для навчання нейронних мереж, використання такого підходу не надає суттєво кращого результату і виникає необхідність у проектуванні власної нейронної мережі.

Основна частина. Для даного дослідження було зібрано датасет, що містить більше ніж 40000 зображень монет, до нього увійшли сучасні обігові монети України, Польщі, Великобританії та країн Євросоюзу (Австрія, Андорра, Бельгія, Ватикан, Греція, Ірландія, Іспанія, Італія, Кіпр, Латвія, Литва, Люксембург, Мальта, Монако, Нідерланди, Німеччина, Португалія, Сан-Марино, Словаччина, Словенія, Фінляндія, Франція, Естонія), усього монети належать до 26 країн. Усі зображення були приведені до спільного вигляду і мають ширину 300 пікселів і висоту 150 пікселів, оскільки кожне з них містить два квадратних (150*150

пікселів) зображення, на яких подано фотографії монети із двох сторін на білому або світло-сірому фоні. Датасет було розділено на дві частини, 80% від загальної кількості становлять тренувальні дані, на яких мережа навчатиметься, а 20% – тестові, яких мережа «не бачила», та на яких перевірялася її робота. Під час навчання нейронної мережі тренувальні дані було знову розділено на 80% виключно для тренування та 20% для валідації під час кожної епохи навчання, що дозволило швидко визначити перенавчання.

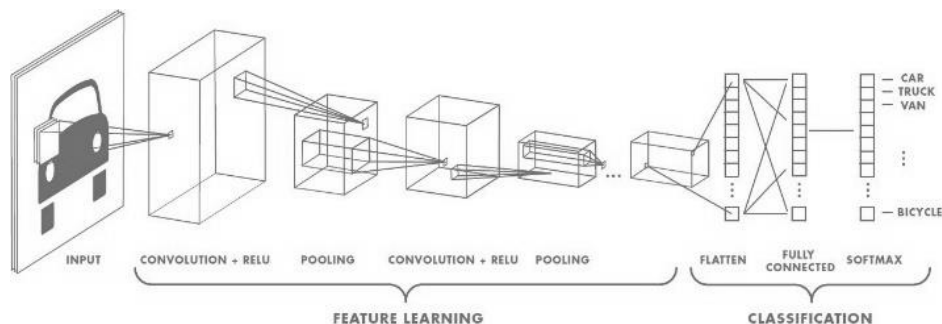


Рисунок 1 – Згорткова нейронна мережа

Через специфічні особливості вхідних даних, кількість тренувальних зображень для монет кожного класу суттєво залежить від часу, протягом якого вона випускалася та тиражів випуску. Як наслідок, найбільшим недоліком зібраного датасету є нерівномірність розподілення зображень за класами, наприклад, найменше усього зображень зібрано для монет невеликої Андорри, а найбільше – для монет Німеччини, які випускаються одразу на 5 монетних дворах дуже великими тиражами. Загалом, зображення були розподілені на 212 класів виду «номінал, грошова одиниця, країна». У датасеті наявні 8 грошових одиниць: копійка, гривня, грош, злотий, пенні, фунт, євроцент та євро і 8 номіналів: 0.500, 1, 2, 5, 10, 20, 25 та 50. Також необхідно зауважити, що до датасету було включено як звичайні обігові монети, так і пам'ятні монети, що були випущені в обіг меншими тиражами, це такі монети, як, наприклад, пам'ятні українські гривні та пам'ятні монети номіналом 2 євро у країнах Євросоюзу. Останні досить сильно ускладнюють розпізнавання, адже усі монети євро мають спільний дизайн реверсу, і, хоча звичайні монети мають індивідуальні дизайни аверсу для монет кожної країни і інколи зазначення назви країни, на деяких пам'ятних монетах вказівки на країну походження є досить непомітними і для людини.

Для роботи із нейронною мережею було застосовано середовище Google Colab, яке надає потужності для створення, навчання, тестування та використання нейронних мереж. Блокноти Google Colab надають можливість працювати з відкритою бібліотекою для машинного навчання TensorFlow [7] та Keras [8] – надбудовою над нею. TensorFlow підтримує декілька популярних мов програмування, проте найбільш зручним та оптимальним вибором є саме мова Python, яка зазвичай використовується при розробці нейронних мереж.

Враховуючи специфіку вхідних даних, які передбачають прямокутне (300*150 пікселів) зображення замість традиційно квадратного та вузько стандартизоване подання зображення (дві сторони монети на білому або світло-сірому фоні) і обмеженість ресурсів, що надає Google Colab для навчання нейронних мереж, було вирішено застосувати власну нейронну мережу, що задається кодом представленим на рис. 2.

```

model = keras.models.Sequential([
    keras.layers.experimental.preprocessing.Rescaling(1./255, input_shape=(150, 300, 3)),
    keras.layers.Conv2D(64, 3, activation='relu'),
    keras.layers.MaxPooling2D(),
    keras.layers.Conv2D(128, 3, activation='relu'),
    keras.layers.MaxPooling2D(),
    keras.layers.Conv2D(256, 3, activation='relu'),
    keras.layers.MaxPooling2D(),
    keras.layers.Flatten(),
    keras.layers.Dense(512, activation='relu'),
    keras.layers.Dense(256, activation='relu'),
    keras.layers.Dense(212)
])

```

Рисунок 2 – Задання нейронної мережі

Дана нейронна мережа є досить нескладною і містить три пари шарів згортки, що містять 64, 128 та 256 фільтрів розміру 3*3 і чергуються з шарами максимізаційного агрегування (max pooling), що містять фільтри розміром 2*2, які віднаходять і передають на наступний шар максимальне значення. На виході нейронної мережі розміщено 2 повнозв'язних шари із функцією активації ReLU та кількістю нейронів 512 і 256 і один повнозв'язний шар із 212 нейронів для класифікації. Граф отриманої нейронної мережі подано на рис. 3.

Результати роботи мережі, отримані після навчання протягом 10 епох подані на рис. 4. Для наочності ці дані було представлено також у вигляді графіків точності розпізнавання та функції втрат на тренувальних та валідаційних даних (рис. 5).

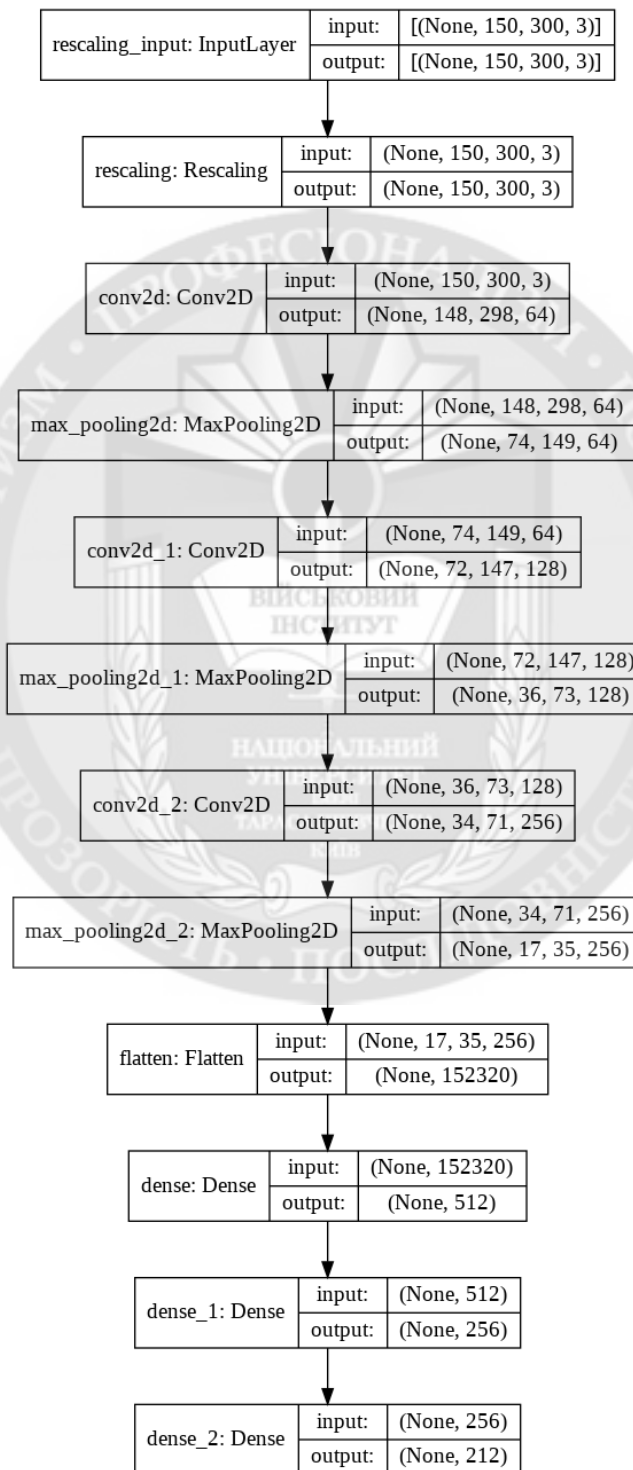


Рисунок 3 – Граф нейронної мережі

```

Epoch 1/10
810/810 [=====] - 194s 236ms/step - loss: 3.9809 - accuracy: 0.1064 - val_loss: 3.1180 - val_accuracy: 0.2072
Epoch 2/10
810/810 [=====] - 189s 233ms/step - loss: 1.2578 - accuracy: 0.6597 - val_loss: 0.4496 - val_accuracy: 0.8691
Epoch 3/10
810/810 [=====] - 189s 233ms/step - loss: 0.2589 - accuracy: 0.9241 - val_loss: 0.3044 - val_accuracy: 0.9103
Epoch 4/10
810/810 [=====] - 188s 232ms/step - loss: 0.1232 - accuracy: 0.9628 - val_loss: 0.2728 - val_accuracy: 0.9239
Epoch 5/10
810/810 [=====] - 188s 232ms/step - loss: 0.0834 - accuracy: 0.9751 - val_loss: 0.2574 - val_accuracy: 0.9349
Epoch 6/10
810/810 [=====] - 188s 231ms/step - loss: 0.0684 - accuracy: 0.9792 - val_loss: 0.3021 - val_accuracy: 0.9324
Epoch 7/10
810/810 [=====] - 187s 231ms/step - loss: 0.0559 - accuracy: 0.9837 - val_loss: 0.2058 - val_accuracy: 0.9460
Epoch 8/10
810/810 [=====] - 187s 231ms/step - loss: 0.0461 - accuracy: 0.9860 - val_loss: 0.2041 - val_accuracy: 0.9529
Epoch 9/10
810/810 [=====] - 187s 230ms/step - loss: 0.0418 - accuracy: 0.9879 - val_loss: 0.2009 - val_accuracy: 0.9531
Epoch 10/10
810/810 [=====] - 187s 230ms/step - loss: 0.0259 - accuracy: 0.9915 - val_loss: 0.3338 - val_accuracy: 0.9372

```

Рисунок 4 – Результати навчання нейронної мережі

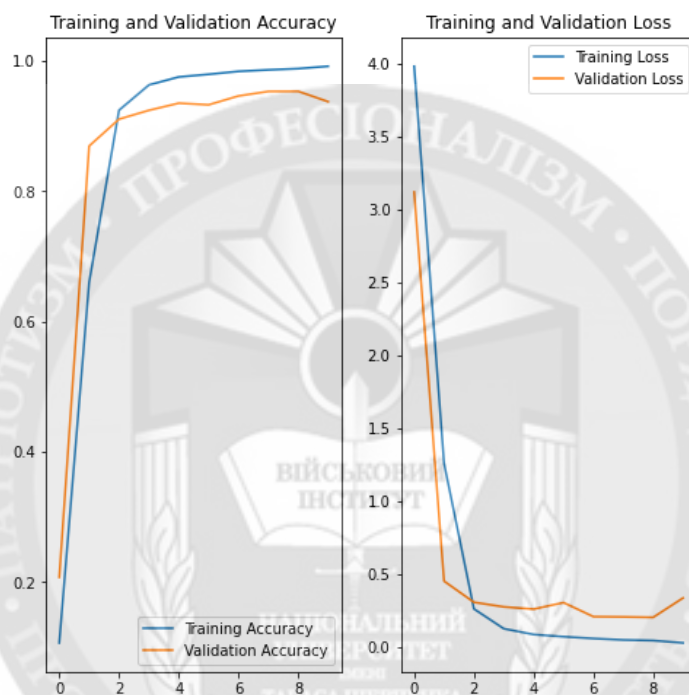


Рисунок 5 – Графік точності розпізнавання та функції втрат

За характером графіків валідаційних даних можна побачити, що збільшення кількості епох призведе лише до перенавчання, коли мережа починає «завчати» тренувальні дані, збільшуючи при цьому точність їх розпізнавання, при майже незмінній точності на валідаційних даних. Точність розпізнавання на тестових даних становила 0,9365 (рис. 6).

```

254/254 [=====] - 18s 69ms/step - loss: 0.3274 - accuracy: 0.9365
[0.32737427949905396, 0.9365196824073792]

```

Рисунок 6 – Результати тестування нейронної мережі

Протягом роботи над нейронною мережею було проведено декілька експериментів з її ускладненням шляхом додавання додаткових шарів та нейронів, проте таке ускладнення себе не виправдало, адже точність розпізнавання тестових даних не зазнала суттєвого збільшення, залишаючись у межах 93-94%, при цьому значно збільшився обсяг необхідних обчислювальних ресурсів та часу на навчання. Для збільшення точності розпізнавання можна внести зміни до датасету зображень, розширивши їх за допомогою аугментації зображень, або застосувати іншу структуру нейронної мережі. З огляду на специфіку даних, одним із

можливих варіантів для підвищення точності є використання згорткової нейронної мережі із декількома виходами.

Огляд нейронної мережі з декількома виходами. При аналізі причин досить низької точності використання класичної згорткової нейронної мережі можна помітити одну із особливостей даних – поточне визначення характеристик обігових монет («номінал, грошова одиниця, країна») надає досить вузьку класифікацію зображень і велику кількість класів (а саме 212), в межах кожного з яких є досить мало зображень, а мала кількість зображень при багатьох класах призводить до зменшення точності навчання.

При подальшому розгляді можна помітити, що кожен клас є складеним і його можна розділити на три окремих характеристики: номінал, грошову одиницю та країну. Кожна із таких характеристик окремо не є достатньою для однозначного визначення певної монети, проте не важко побачити, що при розпізнаванні монет, наприклад, виключно за номіналом, можна отримати значно меншу кількість класів для такої ж кількості тренувальних зображень, що призведе до більшої кількості зображень в межах кожного окремого класу, а отже і до кращих результатів розпізнавання за даною характеристикою. Це стосується і розпізнавання монет виключно за грошовою одиницею чи країною. При цьому, в той час, як розширення датасету за рахунок додавання, наприклад, зображень сучасних обігових монет ще однієї країни, до комбінованої характеристики необхідно додати близько десяти класів, до переліку класів країн можна додати лише один клас, а до переліку класів грошових одиниць та номіналів – від нуля до двох класів.

У випадку, коли невеликий за розмірами датасет містить велику кількість складених класів, можна звернутися до менш поширеної архітектури згорткової нейронної мережі із декількома виходами [9, 10], що передбачає отримання на вхід зображення і повернення декількох векторів відповідностей, по одному для кожної із декількох класифікацій. Нейронна мережа з декількома виходами реалізується за рахунок розгалуження нейронної мережі на певному етапі, що призводить до утворення двох або більше гілок. Такі розгалуження можна утворювати на будь-якому етапі нейронної мережі (рис. 7). Зазвичай таку мережу використовують тоді, коли виникає необхідність отримати дані різних форматів за допомогою різних підходів до їх обробки, наприклад, класифікацію та результат регресійного аналізу, проте вона є доречною і для зображень, які можуть мати декілька непов'язаних характеристик.

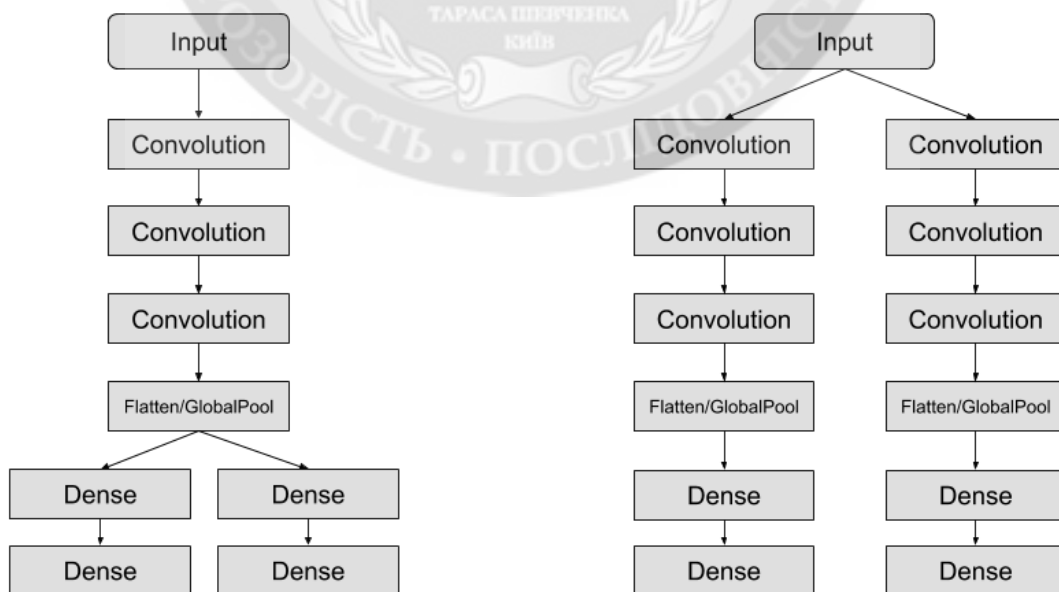


Рисунок 7 – Приклади нейронних мереж із декількома виходами

Під час розпізнавання обігових монет за допомогою згорткової нейронної мережі із декількома виходами можна отримати три вектори відповідностей, по одному для номіналу,

грошової одиниці та країни, при цьому точність розпізнавання для кожної із характеристик буде суттєво більшою за точність розпізнавання звичайною мережею із одним виходом з застосуванням складених характеристик у якості класів. Ще однією з переваг такого підходу є можливість більш тонкого налаштування нейронної мережі та економії ресурсів при навчанні, якщо після розгалуження у якості шарів гілок застосувати попередньо навчені моделі, адже інколи моделі для різних характеристик можуть навіть потребувати різної кількості епох тренування.

У даній роботі використано нейронну мережу із розгалуженням на три гілки на початку мережі, причому кожна із гілок представлена одним шаром, який являє собою звичайну згорткову модель. Для загальної нейронної мережі відмінності між певною послідовністю шарів та моделлю, яка являє собою таку саму послідовність шарів не виникне, проте це дозволить окремо навчити певну гілку та використати отримані ваги у загальній нейронній мережі. Код загальної нейронної мережі подано на рис. 8, а представлення у вигляді графу, на рис. 9.

```
input = keras.layers.Input((150, 300, 3))
value_layer = value_model(input)
currency_layer = currency_model(input)
country_layer = country_model(input)
new_model = keras.models.Model(inputs = input, outputs = [value_layer, currency_layer, country_layer])
```

Рисунок 8 – Задання нейронної мережі з трьома виходами

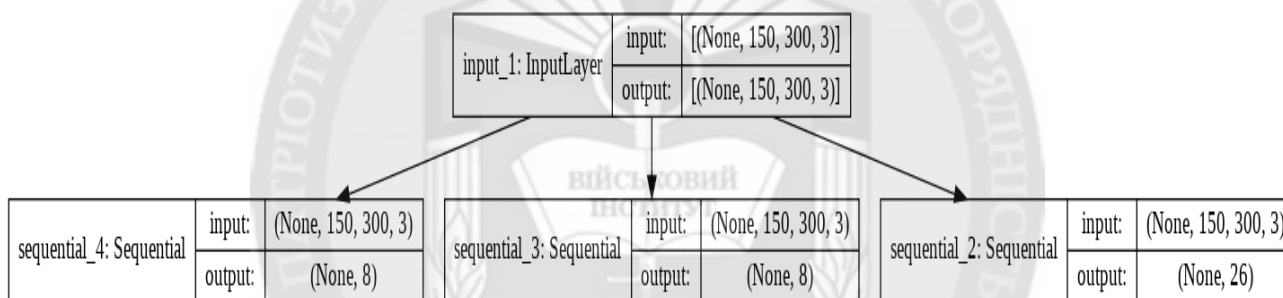


Рисунок 9 – Граф нейронної мережі з трьома виходами

Моделі value_model, currency_model та country_model використані у якості згрупованих шарів і являються моделями для розпізнавання номіналу, грошової одиниці та країни відповідно. Структура цих моделей подібна до традиційної моделі згорткової нейронної мережі (рис. 2) і відрізняються зменшеною кількістю нейронів останнього шару для класифікації – 8 для моделей номіналу та грошової одиниці і 26 для моделі країни.

Навчання кожної із моделей відбувалося окремо, причому розподіл даних на тренувальні та тестові залишився не змінним, в той час як розподіл даних на виключно тренувальні та валідаційні відрізнявся для різних моделей. Результати навчання моделі для розпізнавання номіналу, грошової одиниці та країни подані на рис. 10, 12 та 14 відповідно у вигляді графіків точності розпізнавання та функції втрат на тренувальних та валідаційних даних. Точність розпізнавання на тестових даних становила 0,9941 для номіналу, 0,9989 для грошової одиниці та 0,9641 для країни (рис. 11, 13 та 15 відповідно).

За отриманими графіками видно, коли для кожної окремої моделі починається перенавчання, що дозволяє визначити необхідну кількість епох навчання для кожної окремої моделі. Проаналізувавши отримані результати для тестових даних можна помітити, що точність розпізнавання номіналу та грошової одиниці є досить високою, приблизно 99%, а точність розпізнавання країн менша, у межах 96-97%. Менша точність пояснюється, зокрема більшою кількістю класів та специфікою вхідних даних, зокрема наявністю у датасеті зображень пам'ятних монет номіналом 2 євро ускладнює визначення приналежності цих монет до певної країни Євросоюзу. Неважко побачити, що точність розпізнавання номіналу,

грошової одиниці та країни окремо вища за результат, отриманий при застосуванні традиційної мережі, при цьому ми із досить високою вірогідністю отримуємо правильно визначений номінал та грошову одиницю. Якщо визначити загальну точність розпізнавання зображень як добуток результатів кожної з гілок, ми отримаємо підсумкову точність величиною 0,9574, що перевищує точність традиційної згорткової нейронної мережі на 0,0209, що є досить непоганим приростом. При більш гнучкому налаштуванні і тренуванні кожної окремої моделі, що входять до складу згорткової нейронної мережі з трьома виходами, можна і надалі покращити отриманий результат.

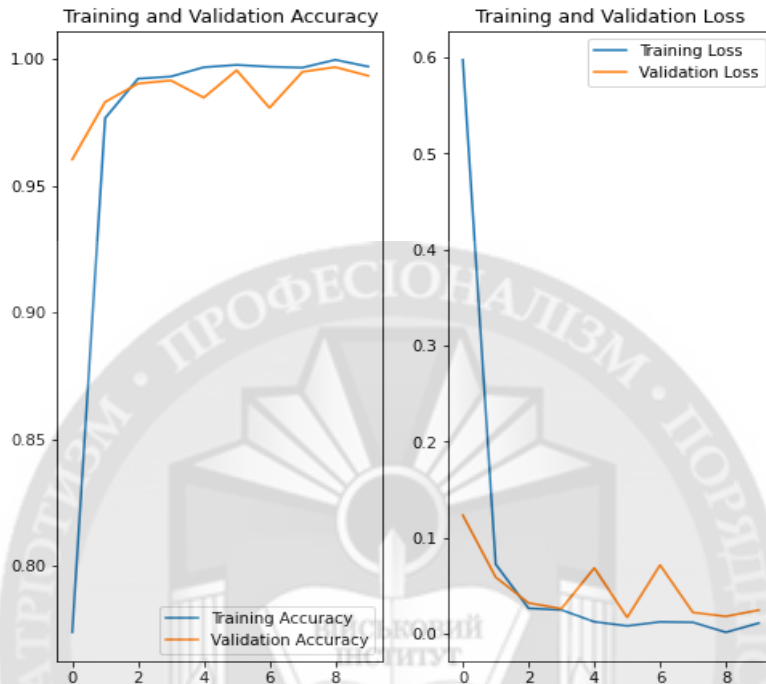


Рисунок 10 – Графіки тренування моделі розпізнавання номіналу

254/254 [=====] - 18s 69ms/step - loss: 0.0236 - accuracy: 0.9941
 [0.023559754714369774, 0.994071900844574]

Рисунок 11 – Результати тестування моделі розпізнавання номіналу

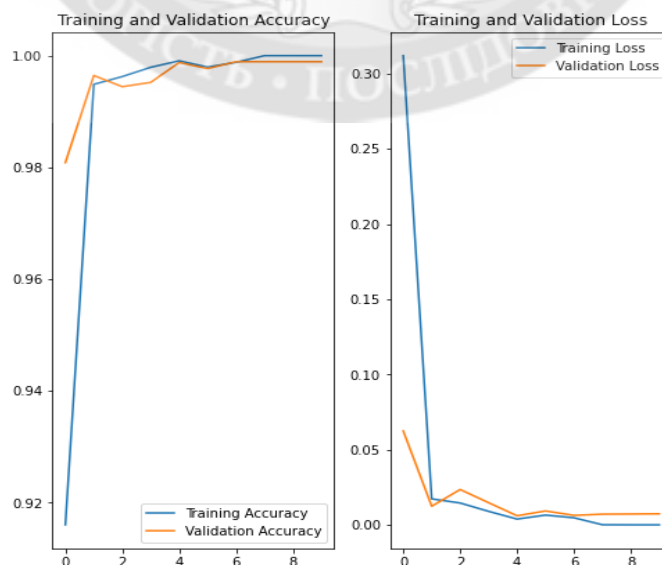


Рисунок 12 – Графіки тренування моделі розпізнавання грошової одиниці

254/254 [=====] - 18s 70ms/step - loss: 0.0035 - accuracy: 0.9989
[0.003485651919618249, 0.9988884925842285]

Рисунок 13 – Результати тестування моделі розпізнавання грошової одиниці

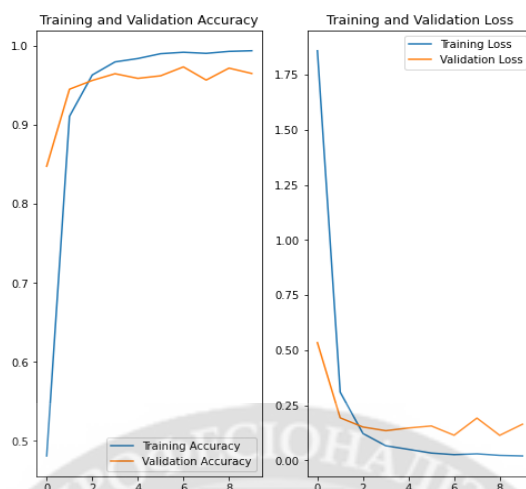


Рисунок 14 – Графіки тренування моделі розпізнавання країни

254/254 [=====] - 18s 70ms/step - loss: 0.1752 - accuracy: 0.9641
[0.1752026081085205, 0.9640607833862305]

Рисунок 15 – Результати тестування моделі розпізнавання країни

Висновки. Задача розпізнавання монет за зображенням через свою специфіку дозволяє застосувати декілька підходів до розбиття зображень на класи, а отже і до побудови згорткової нейронної мережі. У роботі було наведено традиційний підхід, коли кожне зображення визначається одним складеним класом, що одночасно вказує на номінал, грошову одиницю та країну і нейронна мережа має один вихід та підхід із використанням нейронної мережі з декількома виходами, коли кожна із характеристик зображення являє собою окрему класифікацію і нейронна мережа має три виходи, по одному для кожної із них.

Через велику кількість малих за обсягом зображень класів при традиційному підході результат розпізнавання виявився не найкращим, 93-94%, при цьому розгалуження нейронної мережі на три гілки для кожної окремої характеристики надало хороший результат у 99% для номіналу та грошової одиниці і 95-96% для країни. Перевагою такого підходу є можливість більш гнучко налаштувати та навчити окремі моделі для кожної характеристики та використати їх у якості шарів окремих гілок нейронної мережі. Недоліком такого підходу є в декілька разів більша кількість параметрів нейронної мережі, що означає суттєве збільшення часу на її навчання та використання ресурсів, при цьому окреме тренування вкладених моделей дозволить зменшити кількість ресурсів, що одночасно задіяні під час навчання мережі.

ЛІТЕРАТУРА:

1. LeCun Y. Convolutional Networks for Images, Speech, and Time-Series / Y. LeCun, Y. Benigo. – 1995. – 14 с.
2. Ямпольский Леонід Стефанович. Нейротехнології та нейрокомп'ютерні системи : підручник / Л.С. Ямпольский, О.І. Лісовиченко, В.В. Олійник. – К. : Дорадо-Друк, 2016. – 576 с.
3. ImageNet [Електронний ресурс] – Режим доступу до ресурсу: <https://image-net.org/challenges/LSVRC/index.php>.
4. Simonyan K. Very Deep Convolutional Networks for Large-Scale Image Recognition / K. Simonyan, A. Zisserman., 2015. – 14 с.
5. Deep Residual Learning for Image Recognition / К. He, X. Zhang, S. Ren, J. Sun., 2015. – 12 с.

6. Bozinovski S. Reminder of the First Paper on Transfer Learning in Neural Networks / Stevo Bozinovski., 2020. – 12 с.
7. Module: tf | TensorFlow Core v2.6.0 [Электронный ресурс] – Режим доступа до ресурсу: https://www.tensorflow.org/api_docs/python/tf?hl=en.
8. Keras API reference [Электронный ресурс] – Режим доступа до ресурсу: <https://keras.io/api/>.
9. A Survey on Multi-output Learning / [D. Xu, Y. Shi, I. Tsang та ін.]. – 2019. – 21 с.
10. A survey on multi-output regression / H.Borchani, G. Varando, C. Bielza, P. Larranaga. – 2015. – 27 с.
11. Хайкин Саймон. Нейронные сети: полный курс, 2-е изд. : пер. с англ. / Саймон Хайкин. – М. : Издательский дом «Вильямс», 2006. – 1104 с.
12. CS231n Convolutional Neural Networks for Visual Recognition [Электронный ресурс] – Режим доступа до ресурсу: <https://cs231n.github.io>.

REFERENCES:

1. LeCun Y. (1995), “Convolutional Networks for Images, Speech, and Time-Series”, 14 p.
2. Yampolskyi L.S., Lisovychenko O.I., Oliinyk V.V. (2016), “Neirotekhnolohii ta neurokompiuterni systemy: pidruchnyk ” [Neurotechnology and neurocomputer systems], Dorado-Druk, Kyiv, 576 p.
3. “ImageNet”, available at: image-net.org/challenges/LSVRC/index.php.
4. Simonyan K., Zisserman A. (2015), “Very Deep Convolutional Networks for Large-Scale Image Recognition”, 14 p.
5. He K., Zhang X., Ren S., Sun J. (2015), “Deep Residual Learning for Image Recognition”, 12 p.
6. Bozinovski S. (2020), “Reminder of the First Paper on Transfer Learning in Neural Networks”, 12 p.
7. “Module: tf | TensorFlow Core v2.6.0”, available at: www.tensorflow.org/api_docs/python/tf?hl=en.
8. “Keras API reference”, available at: keras.io/api/.
9. Xu D., Shi, Y., Tsang I.W., Ong Y.-S., Gong C., Shen X. (2019), “A Survey on Multi-output Learning”, 21 p.
10. Borchani H., Varando G., Bielza C., Larranaga P. (2015), “A survey on multi-output regression”, 27 p.
11. Haykin S. (2006) “Neyronnyye seti: polnyy kurs, 2-e izd.: per. s angl.” [Neural networks: full course], Izdatelskiy dom Viliams, Moscow, 1104 p.
12. “CS231n Convolutional Neural Networks for Visual Recognition”, available at: cs231n.github.io.

Vaivala Ye.Yu., PhD Tsopa N.V., Shmidke V.S.

APPLICATION OF A CONVOLUTIONAL NEURAL NETWORK WITH MULTIPLE OUTPUTS FOR CIRCULATING COINS RECOGNITION

The problem of circulation coins recognition can be faced both by experienced collectors and by ordinary tourists or novice collectors. The easiest solution for those who do not have enough knowledge for coins recognition on one’s own may be coins recognition using software based on a neural network. Since most coin characteristics can be identified by obverse and reverse images and the process of creating coin photographs is fast and does not require additional measuring devices, the most convenient approach is to recognize coins by images.

The article considers the problem of circulating coins recognition by images using convolutional neural networks, which contain layers designed specifically for working with images. The traditional approach to solving the problem of image recognition is described, which involves the use of a regular convolutional neural network with one output, when each image is assigned to a class that uniquely defines it. The analysis of this approach and obtained results shows that to solve the problem for the collected dataset, when there is a large number of classes, such an architecture is not the best. To increase the accuracy of recognition, the architecture of a convolutional neural network with several outputs was used, which includes branching of the neural network structure into several branches at a certain stage. When using such a neural network, the image is matched by a list of several independent characteristics instead of one composite class. Thus, the task of coin recognition can be divided into several subtasks, each of which corresponds to a separate branch of the neural network. The results obtained using a regular neural network and a neural network with multiple outputs were compared with each other, after that the reasons for the differences in results and the advantages and disadvantages of each of the approaches were given.

Keywords: convolutional neural network, multi-output neural network, image recognition, machine learning, artificial intelligence.