КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ РАДІОФІЗИКИ, ЕЛЕКТРОНІКИ ТА КОМП'ЮТЕРНИХ СИСТЕМ

Кафедра медичної радіофізики

До захисту допущено: «На правах рукопису»
Завідувач кафедри _____ Сергій РАДЧЕНКО
« __ » червня 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему:

« АНАЛІЗ ДАНИХ З ВИКОРИСТАННЯМ АПАРАТНОГО АКСЕЛЕРАТОРА ЗГОРТКОВИХ НЕЙРОМЕРЕЖ Intel NCS2 »

Виконав:
студент 4-го курсу
денної форми навчання
спеціальності 105 – Прикладна фізика та наноматеріали
ОП «Електроніка та інформаційні технології в медицині»
Метельський Ілля Анатолійович _____

Науковий керівник:
канд. фіз.-мат. наук, доцент
Судаков Олександр Олександрович _____

Рецензент:
канд. фіз.-мат. наук, доцент
Бойко Юрій Володимирович _____

Засвідчую, що у цій бакалаврській роботі
немає запозичень з праць інших авторів без
відповідних посилань
Студент _____

Робота допущена до захисту в ЕК рішенням кафедри медичної радіофізики
від «___» червня 2023 р., протокол № __.

Завідувач кафедри медичної радіофізики,
канд. фіз.-мат. наук, доцент
Радченко Сергій Петрович _____

Київ – 2023

РЕФЕРАТ

Бакалаврська робота: 35 с., 20 рис., 13 джерел.

Наведено огляд літератури за тематикою згорткових нейромереж. Виконано тестування апаратного акселератора згортокових неромереж “Intel Neural Compuer Stick 2” для задачі розпізнавання зображень

КЛЮЧОВІ СЛОВА: НЕЙРОМЕРЕЖА, ЗГОРТКОВІ НЕЙРОМЕРЕЖІ, ШТУЧНИЙ ІНТЕЛЕКТ, РОЗПІЗНАВАННЯ ОБРАЗІВ, АНАЛІЗ ЗОБРАЖЕННЯ, INTEL, OPENVINO, PYTHON, INTEL, NEURAL COMPUTE STICK.

ЗМІСТ

ЗМІСТ 2
ВСТУП 3
1. НЕЙРОННІ МЕРЕЖІ 4
1.1. Штучні нейронні мережі 4
1.2 Нейронна мережа прямого зв’язку 4
1.3 Згорткові нейронні мережі 6
2. Налаштування апартного акселератора 15
2.1 Основні характеристики Intel Neural Compute Stick 2: 15
2.2 Архітектура 17
OpenVINO 20
3 Вимірювання характеристик апаратного акселератора 21
3.1 Знаходження об’єктів 21
3.2 SqueezeNet 21
3.2 Продуктивність 26
Висновки 30
Перелік використаних джерел 31

ВСТУП

Штучний інтелект вже давно міцно увійшов в наше життя. Найчастіше ми навіть не замислюємося над цим питанням, коли звично користуємося голосовим помічником в смартфоні або автоматичним розпізнаванням зображень в програмі.

Нейронні мережі використовуються для вивчення та ідентифікації шаблонів у даних. У контексті розпізнавання об'єктів нейронні мережі навчаються визначати певні властивості об'єктів, які дозволяють відрізняти їх один від одного. Ці властивості можуть включати колір, форму, розмір або будь-яку іншу відмітну характеристику. Можливість нейронних мереж «засвоїти» ці властивості - це те, що робить їх добре придатними для таких завдань, як класифікація зображень та виявлення об'єктів.

На сьогодні головними проблемами застосування згорткових нейромереж, особливо на мобільних пристроях, є значні вимоги до обчислювальної продуктивності апаратного забезпечення. Для цього останнім часом розробляються портативні обчислювальні акселератори, однак такі пристрої є вузькоспеціалізованими і необхідно вивчати можливість їх застосування для аналізу різних типів даних, таких як різні медичні діагностичні дані.

У даній роботі, було експериментально досліджено характеристики акселератора згорткових нейромереж Intel Neural Compute Stick 2 для розпізнавання зображень. Були проаналізовані приклади коду та нейронних мереж, а саме squeezenet1.0, яка створена для розпізнавання та класифікації образів, за допомогою Intel NCS 2.

1. НЕЙРОННІ МЕРЕЖІ

1.1. Штучні нейронні мережі

Штучні нейронні мережі – це математична програмна модель, побудована за принципом функціонування біологічних нейронних мереж — мереж нервових клітин живого організму.

Штучні нейронні мережі(ШНМ) навчаються за допомогою навчального набору. Наприклад, припустімо, що ви хочете навчити ANN розпізнавати кота. Потім йому показують тисячі різних зображень кішок, щоб мережа навчилася ідентифікувати кота. Після того, як нейронна мережа достатньо навчена використанню зображень котів, нам потрібно перевірити, чи може вона правильно ідентифікувати зображення котів. Це робиться шляхом того, що ШНМ класифікує надані зображення, вирішуючи, чи є вони зображеннями котів чи ні. Вихідні дані, отримані ШНМ, підтверджуються наданим людиною описом того, чи є зображення зображенням кота чи ні. Якщо ШНМ ідентифікує неправильно, тоді використовується зворотне розповсюдження для коригування всього, що він дізнався під час навчання. Зворотне поширеннявиконується шляхом точного налаштування ваг з’єднань в одиницях ШНМ на основі отриманої частоти помилок. Цей процес триває до тих пір, поки штучна нейронна мережа не зможе правильно розпізнати кота на зображенні з мінімальною можливістю помилок. [1]

Розглянемо типи штучних нейронних мереж які нас цікавлять:

1.2 Нейронна мережа прямого зв’язку

Нейронна мережа прямого зв’язку є однією з найпростіших штучних нейронних мереж. У цій ШНМ надані дані або вхідні дані передаються в одному напрямку від вхідних до вихідних нейронів.

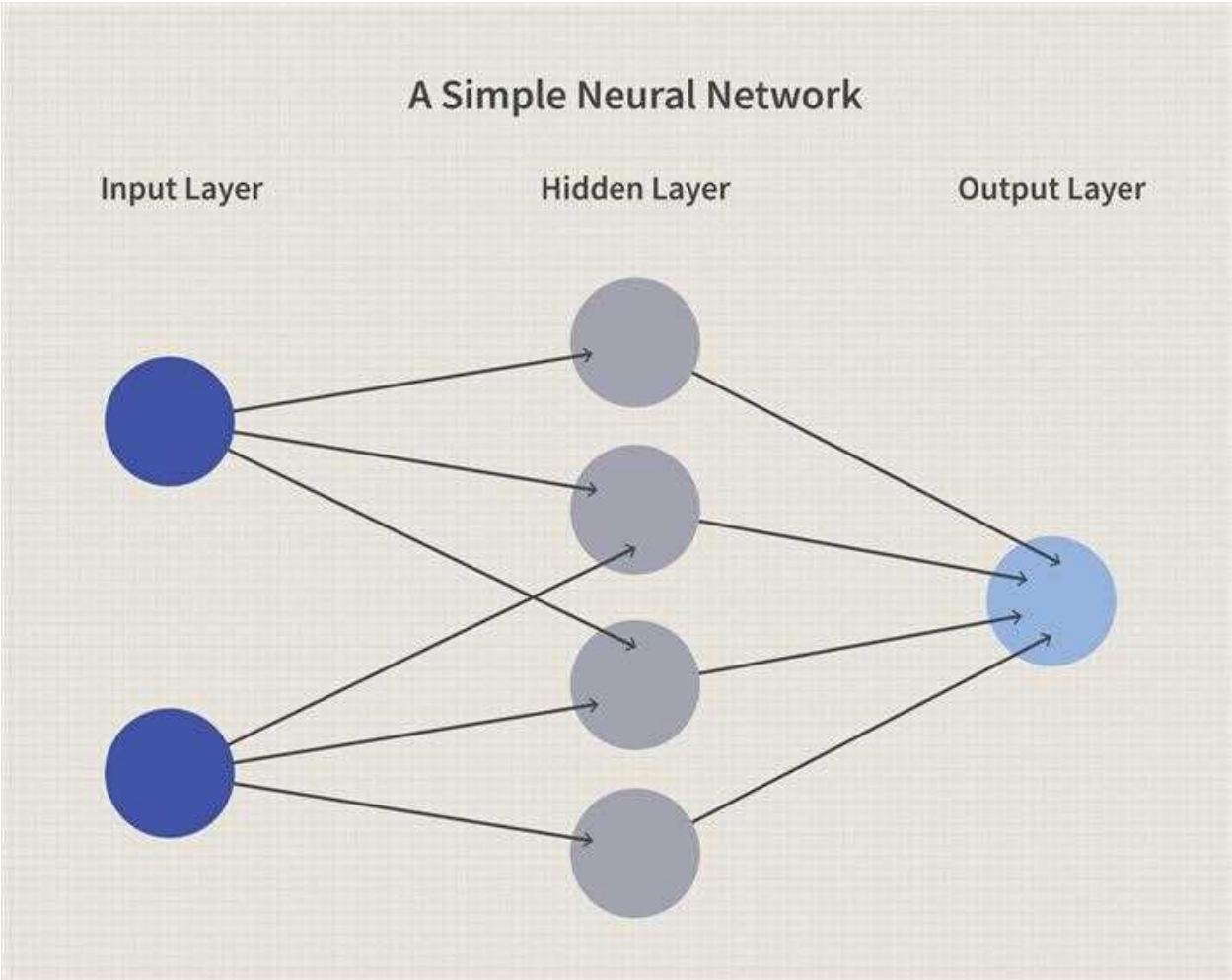

Рис. 1.1 Модель простої штучної нейромережі

У звичайній нейронній мережі є три типи шарів:

1. Вхідні шари: це шар, на якому ми надаємо вхідні дані для нашої моделі. Кількість нейронів у цьому шарі не менша, ніж загальна кількість ознак у наших даних. У якості ознак часто (виступають всі пікселі зображення).

2. Прихований шар або шари: дані з вхідного шару подаються до прихованого шару. Може бути багато прихованих шарів залежно від моделі та обсягу даних. Кожен прихований шар може мати різну кількість нейронів, яка, як правило, перевищує кількість ознак. Вихідні дані кожного шару обчислюються шляхом множення вектора вихідних

даних попереднього шару на матрицю вагових коефіцієнтів цього шару. До результуючого вектора додається вектор зміщень, і до результату застосовується функція активації, яка робить мережу нелінійною.

3. Вихідний рівень: вихідні дані з прихованого шару подаються в логістичну функцію, наприклад softmax, яка перетворює вихідні дані кожного класу в оцінку ймовірності кожного класу. [2]

Найбільшою проблемою звичайних нейронних мереж (НМ) є відсутність масштабованості. Для менших зображень із меншою кількістю кольорових каналів звичайний НМ може дати задовільні результати, але зі збільшенням розміру та складності зображення потреба в обчислювальній потужності та ресурсах також зростає, що вимагає більшої мережі.[3]

Крім того, з часом також виникає проблема перенавчання, коли нейронна мережа намагається вивчити занадто багато деталей у навчальних даних. Воно також може закінчитися «вивченням» шуму в даних, що впливає на його продуктивність на тестових наборах даних. Зрештою, НМ не в змозі ідентифікувати характеристики або шаблони в наборі даних і, таким чином, сам об’єкт. [3]

1.3 Згорткові нейронні мережі

Convolutional Neural Networks (CNNs) або звичайно відомі як **згорткові нейронні мережі** - це потужний тип штучних нейронних мереж, що знайшов широке застосування у багатьох областях комп'ютерного зору, обробки зображень, розпізнавання об'єктів та багатьох інших сферах.

Згорткові нейронні мережі були вперше впроваджені в 1980-х роках і вони були великим кроком вперед у сфері комп'ютерного зору. Їхній успіх обумовлений здатністю ефективно розпізнавати та аналізувати візуальну

інформацію, використовуючи особливість обробки зображень шляхом застосування фільтрів, відомих як згортки.

Для задач розпізнавання зображень, класифікації зображень і комп’ютерного зору (CV) CNN особливо корисні, оскільки вони забезпечують високоточні результати, особливо коли задіяно багато даних. CNN також вивчає особливості об’єкта в послідовних ітераціях, коли дані об’єкта переміщуються через численні шари CNN. Це пряме (і глибоке) навчання усуває потребу в ручному вилученні ознак (створення ознак). [3]

Основні переваги CNN полягають у їхній здатності працювати з великими обсягами даних, навчатись загальним моделям, виявляти локальні залежності та використовувати ієрархічну архітектуру для розпізнавання об'єктів.

Однією з ключових особливостей CNN є їх здатність виявляти локальні залежності на зображеннях за допомогою згортки. Це дозволяє їм фокусуватись на невеликих областях зображення та виявляти важливі риси. Крім того, CNN використовують подільність параметрів, що дозволяє ефективно використовувати обмежену кількість параметрів та знижує ризик перенавчання моделі.

Ієрархічна архітектура CNN дозволяє їм поступово виявляти все більш абстрактні та складні ознаки на зображеннях. Починаючи з нижніх шарів, які виявляють прості риси, вони поступово переходять до вищих шарів, які розпізнають складніші об'єкти та контекст.

CNN широко використовуються для розпізнавання об'єктів на зображеннях. Вони можуть виконувати класифікацію зображень та виявлення об'єктів з високою точністю. Це робить їх корисними в різних сферах, включаючи комп'ютерне зору, медицину, автономні автомобілі та багато інших застосувань. Згорткові нейронні мережі є потужним інструментом, який

допомагає автоматизувати процес обробки зображень та розпізнавання об'єктів.

Згорткова нейронна мережа, складається з кількох рівнів, таких як: вхідний рівень, згортковий рівень, та повністю зв’язані рівні.

Рівень згортки використовує фільтри, які виконують операції згортки під час сканування вхідних даних, щодо його розмірів. [4]

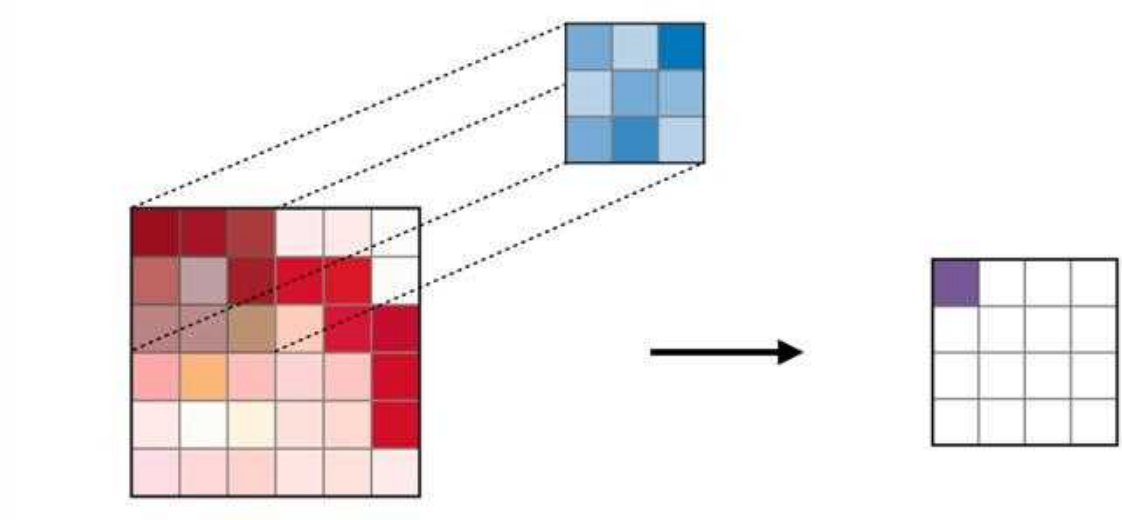

Рис. 1.2 Операція згортки

Ядро CNN працює на основі наступної формули:

Розміри зображення

$R = n_1 * n_2 * 1$, де n1 - висота, n2 - ширина та 1 - кількість каналів, наприклад RGB.

Отже, як приклад, формула матиме вигляд $R = 5 * 5 * 1$ [5]

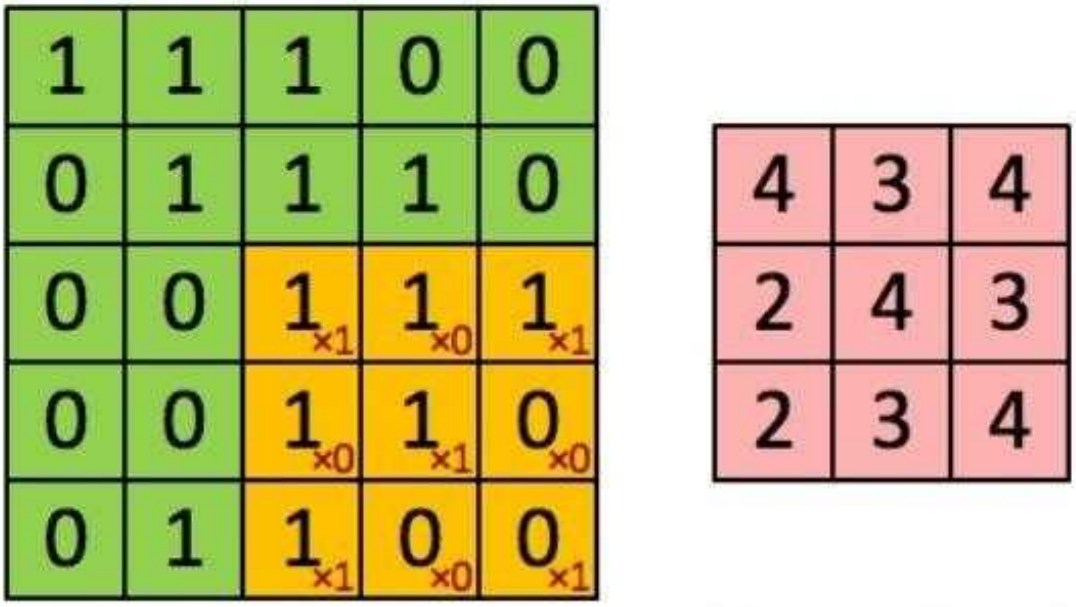

Рис. 1.2 Виконання операції згортки

На (Рис.2) ядро (позначено жовтим квадратом) ящик переміщується від першої комірки до останньої, виконуючи операцію згортки з матрицею 3x3. Ця операція називається Kernel (K).[5]

Якщо наявні декілька каналів, наприклад, у зображеннях RGB, то ядро має таку ж глибину, як у вихідному зображенні. Матричне множення виконується за допомогою числа Ks, Is. Ця процедура реалізується у форматі стеку, наприклад, {K1, I1}, {K2, I2} і так далі. Результати генеруються шляхом підсумовування зміщення. Отриманий результат представляється у вигляді стиснутого "каналу з однією глибиною", який містить скомпоновані функції.

Метою описаної операції згортки є отримання всіх високорівневих характеристик зображення. Проте, вона не обмежується лише високорівневими функціями, а також виконує операції над низькорівневими функціями, такими як колір і орієнтація градієнта.

Мета цього рівня — зменшити розмірність зображення, яке міститься у вхідному зображенні, і збільшити розмірність вихідного, у деяких випадках, залишити її незмінною, залежно від необхідного результату. [5]

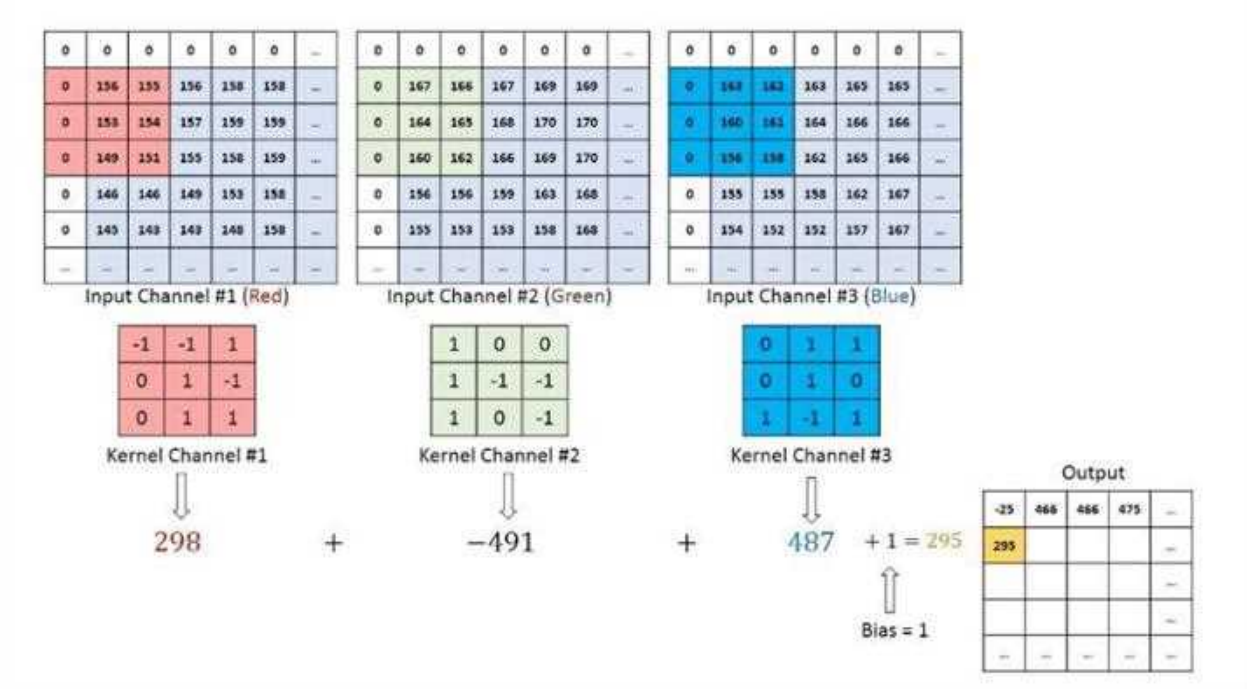

Рис. 1.3 Результат згортки

Агрегувальні шари - є одним із будівельних блоків згорткових нейронних мереж. Там, де згорткові шари застосовують функції до зображень, шари об’єднання об’єднують ознаки, отримані CNN. Його метою є поступове скорочення просторового виміру представлення, щоб мінімізувати кількість параметрів і обчислень у мережі.

Карта функцій, створена фільтрами згорткових шарів, залежить від розташування. Наприклад, якщо об’єкт на зображенні дещо змістився, він може бути нерозпізнаним згортковим шаром. Отже, це означає, що карта об’єктів записує точні позиції об’єктів у вхідних даних. Рівні об’єднання забезпечують «трансляційну інваріантність», яка робить CNN інваріантним до зміщень, тобто навіть якщо вхідні дані CNN зміщені, CNN все одно зможе розпізнавати функції вхідних даних. [6]

Використовують два типи об’єднань: максимальне та середнє — це спеціальні види об’єднання, де береться максимальне та середнє значення відповідно.

1. Максимізаційне агрегування використовує максимальне значення з кожного з кластерів нейронів попереднього шару

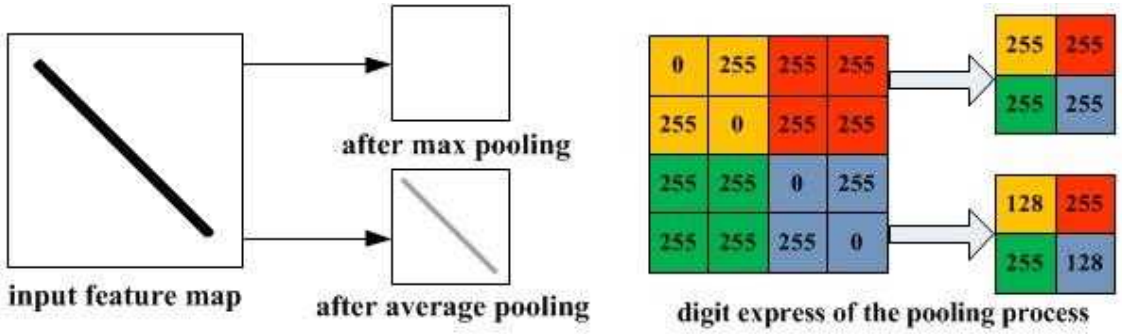

Рис. 1.4 Максимальне агрегування

2. Усереднювальне агрегування - використовує усереднене значення з кожного з кластерів нейронів попереднього шару. Це згладжує зображення, зберігаючи суть зображення.[6]

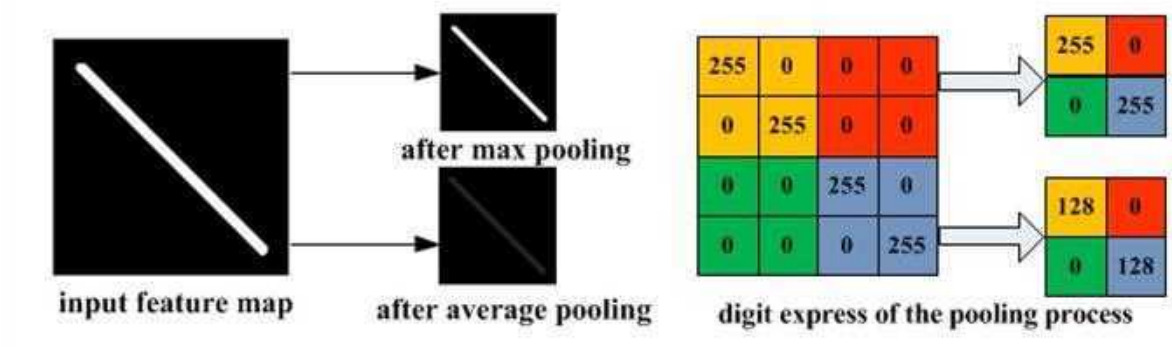

Рис. 1.5 Агрегування середніх значень

3. Повноз'єднані шари

   Нейронні мережі складаються з нелінійних функцій, кожна з яких представлена нейроном або перцептроном. У шарах повного зв'язку, нейрон застосовує лінійне перетворення до вхідного вектора шляхом множення на матрицю ваг. Після цього до отриманого добутку

застосовується нелінійне перетворення за допомогою функції активації *f*.

$$y_{jk}(x) = f(\sum_{i=1}^{n_H} w_{jk} x_i + w_{j0}$$

[7]

Функція активації "f" застосовується до скалярного добутку між вхідними даними шару та матрицею ваг цього шару. Кожен стовпець в матриці ваг має власний номер та оптимізується під час навчання моделі.

У даному випадку, вхідні дані є вектором розмірністю 1x9, а матриця ваг має розмірність 9x4. Шляхом знаходження скалярного добутку та застосування нелінійного перетворення з функцією активації, отримуємо вихідний вектор розмірністю 1x4.

Якщо ми розглянемо шар у повністю зв’язаній нейронній мережі з вхідним розміром дев’ять і вихідним розміром чотири, операцію можна уявити

наступним чином:

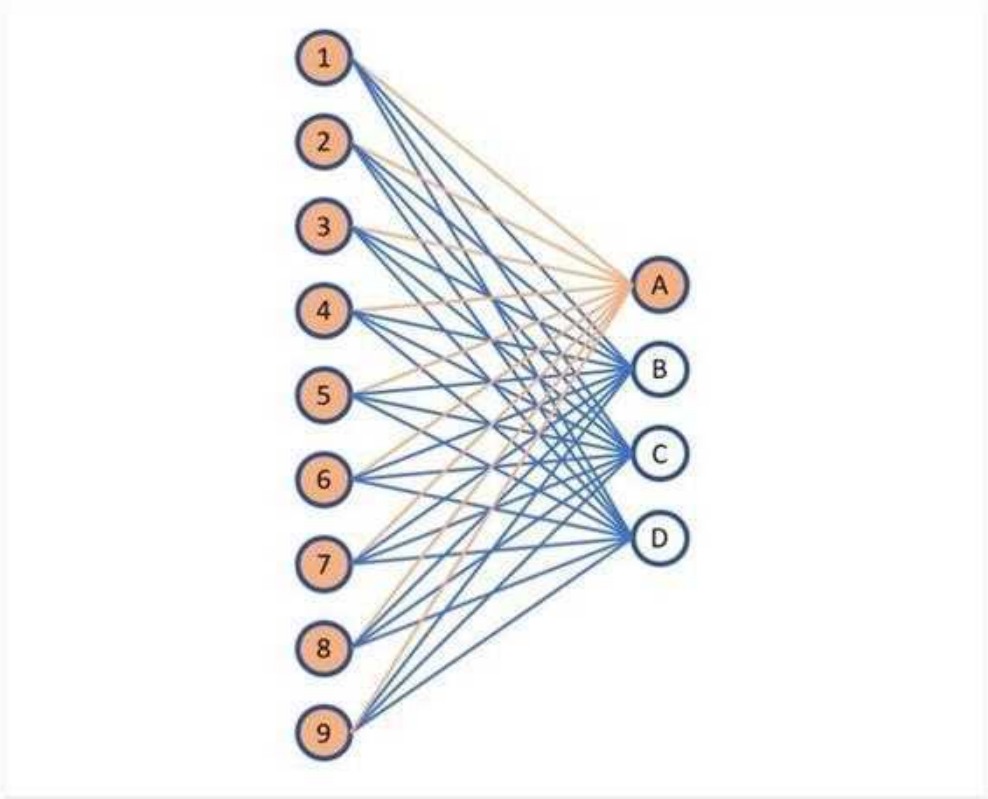

Рис. 1.6 Ілюстрація результату шару з'єднання

Застосування:

- Згорткові нейронні мережі широко використовуються в задачах комп'ютерного зору, таких як розпізнавання облич, класифікація зображень, виявлення об'єктів, сегментація та багато інших. Вони демонструють вражаючі результати у багатьох конкурсах та реальних застосуваннях.
- CNN знайшли широке застосування у сфері медицини, де вони використовуються для діагностики захворювань на основі зображень, виявлення патологій на рентгенівських знімках, аналізу зображень з магнітно-резонансної томографії тощо.
- Інші застосування включають використання CNN в автономних автомобілях для розпізнавання дорожніх знаків та об'єктів, в обробці

відеозаписів, в системах безпеки для виявлення вторгнень або небезпечних ситуацій.

- Окрім обробки зображень, CNN також використовуються для обробки інших типів даних, наприклад, у задачах обробки природних мов, розпізнавання мовлення або аналізу часових рядів.

Згорткові нейронні мережі є потужним інструментом, який змінює підхід до обробки зображень та аналізу даних. Вони пропонують широкий спектр застосувань та можуть досягати вражаючих результатів у багатьох важливих сферах.

2. Налаштування апартного акселератора

Акселератор нейронних мереж Intel Neural Compute Stick 2

Intel Neural Compute Stick 2 (NCS2) - це пристрій для апаратного прискорення інтелектуальних систем, розроблений компанією Intel. Він є невеликим, портативним пристроєм, який працює як апаратний прискорювач нейронних мереж, дозволяючи виконувати обчислення на пристрої замість використання центрального процесора.

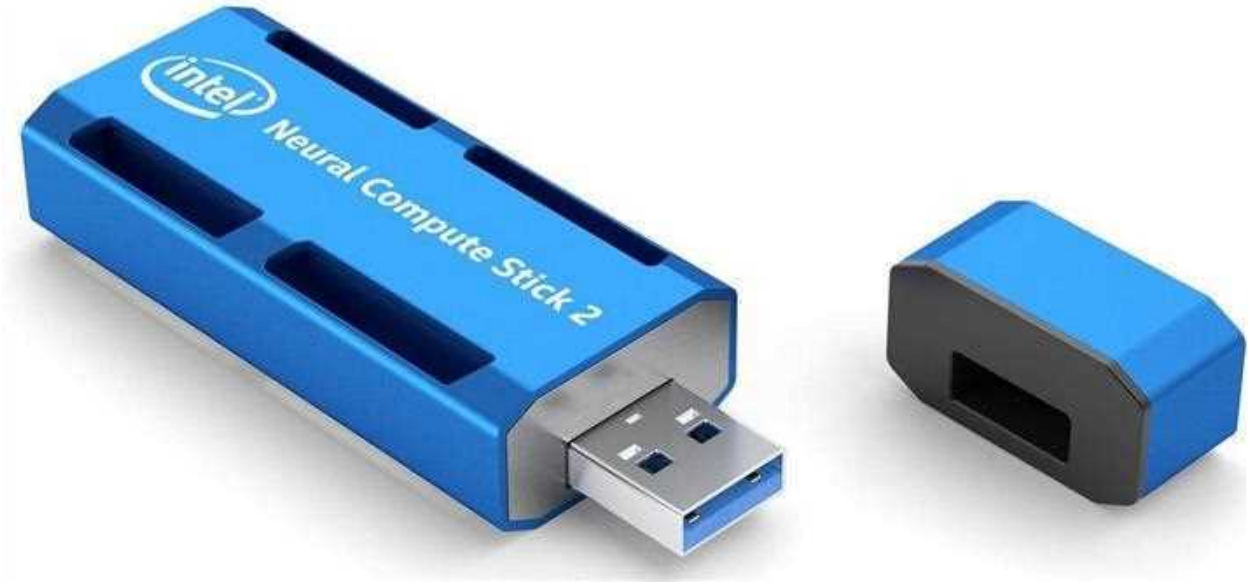

Рис. 2.1 Intel Neural Compute Stick 2

2.1 Основні характеристики Intel Neural Compute Stick 2:

1. Архітектура: Внутрішньо NCS2 базується на архітектурі Intel Movidius Myriad X Vision Processing Unit (VPU), яка є спеціалізованою для виконання операцій штучного інтелекту. Вона має високу енергоефективність і підтримує прискорення машинного навчання та інференсу моделей.

2. Продуктивність: NCS2 має високу продуктивність (4 TOPS (трильйони операцій нейромережі в секунду) та до 10 мільйонів векторних операцій у секунду (VPU GFLOPS)) і може виконувати інтенсивні обчислення

нейронних мереж в реальному часі. Він здатний обробляти велику кількість операцій штучного інтелекту, таких як конволюції, пулінг, активації та інші. Серцем Intel NCS 2 є Intel Movidius Myriad X VPU, нове покоління малопотужного процесора штучного інтелекту, розробленого спеціально для роботи глибоких нейронних мереж (DNN) із високою продуктивністю. Він оснащений Neural Compute Engine, спеціальним вбудованим апаратним прискорювачем. Масив із 16 програмованих ядер SHAVE (Streaming Hybrid Architecture Vector Engine) на 33 відсотки більше, ніж у минулому поколінні, що підтримується інтелектуальною структурою пам’яті надвисокої пропускної здатності. [8]

3. З'єднання: NCS2 підключається до комп'ютера або пристрою за допомогою USB 3.0 або USB 3.1 Gen 1 Type-A порту. Це забезпечує зручність використання і портативність, оскільки пристрій можна підключити до різних систем.
4. Підтримка фреймворків: NCS2 підтримує різні фреймворки та бібліотеки машинного навчання, включаючи TensorFlow, Caffe, MXNet, OpenVINO та інші. Це дає можливість розробникам використовувати їх улюблені інструменти для створення та оптимізації моделей.

5. Інтеграція з OpenVINO: NCS2 щільно інтегрований з Intel OpenVINO Toolkit, який надає набір інструментів для розробки, оптимізації та виконання моделей штучного інтелекту. Набір інструментів Intel® Distribution of OpenVINO™ дозволяє швидше та легше створювати програмне забезпечення, яке емулює людський зір. Бібліотеки попередньо навчених моделей, оптимізовані алгоритми комп’ютерного зору та зразки коду позбавлять вас від необхідності будувати основу рішення з нуля. Набір інструментів також є ключовим для розробки за принципом «напиши один раз, розгорни всюди» за допомогою Intel NCS 2. Загальний API виконує робочі навантаження на всіх типах платформ

комп’ютерного бачення та прискорювачів від Intel, включаючи ЦП, інтегровану графіку, FPGA та VPU всередині Intel NSC 2.[8]

6. Підтримка різних завдань: NCS2 підтримує виконання різних завдань машинного навчання, включаючи класифікацію зображень, детекцію об'єктів, визначення ключових точок, візуальне відстеження тощо. Він може бути використаний у багатьох сферах, таких як комп'ютерне зорове відстеження, розпізнавання образів та автономна навігація.

Intel Neural Compute Stick 2 є потужним інструментом для апаратного прискорення інтелектуальних систем, дозволяючи виконувати обчислення на пристрої замість центрального процесора. Він підходить для розробки та розгортання різних застосунків штучного інтелекту, які вимагають високої продуктивності та низького енергоспоживання.

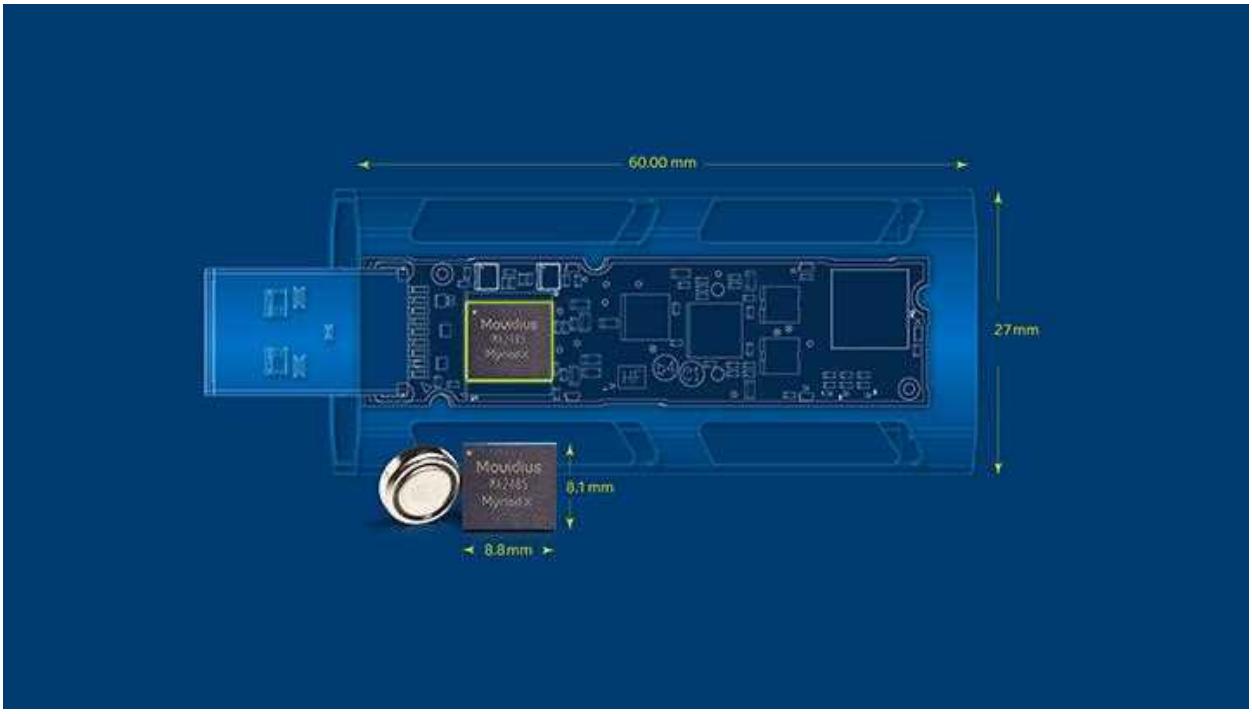

Рис. 2.2 Схема Пристрою Intel NCS 2

2.2 Архітектура

- Intel Movidius NCS 2 містить процесор Intel Movidius Myriad 2 vision, включаючи 4 Гбіт LPDDR
- Intel Movidius NCS 2 підключається до прикладного процесора (AP), наприклад, Rasberry PI або плати UP Squared
- Виконання контролюється мікропроцесором LEON, а обчислення виконуються на процесорах SHAVE

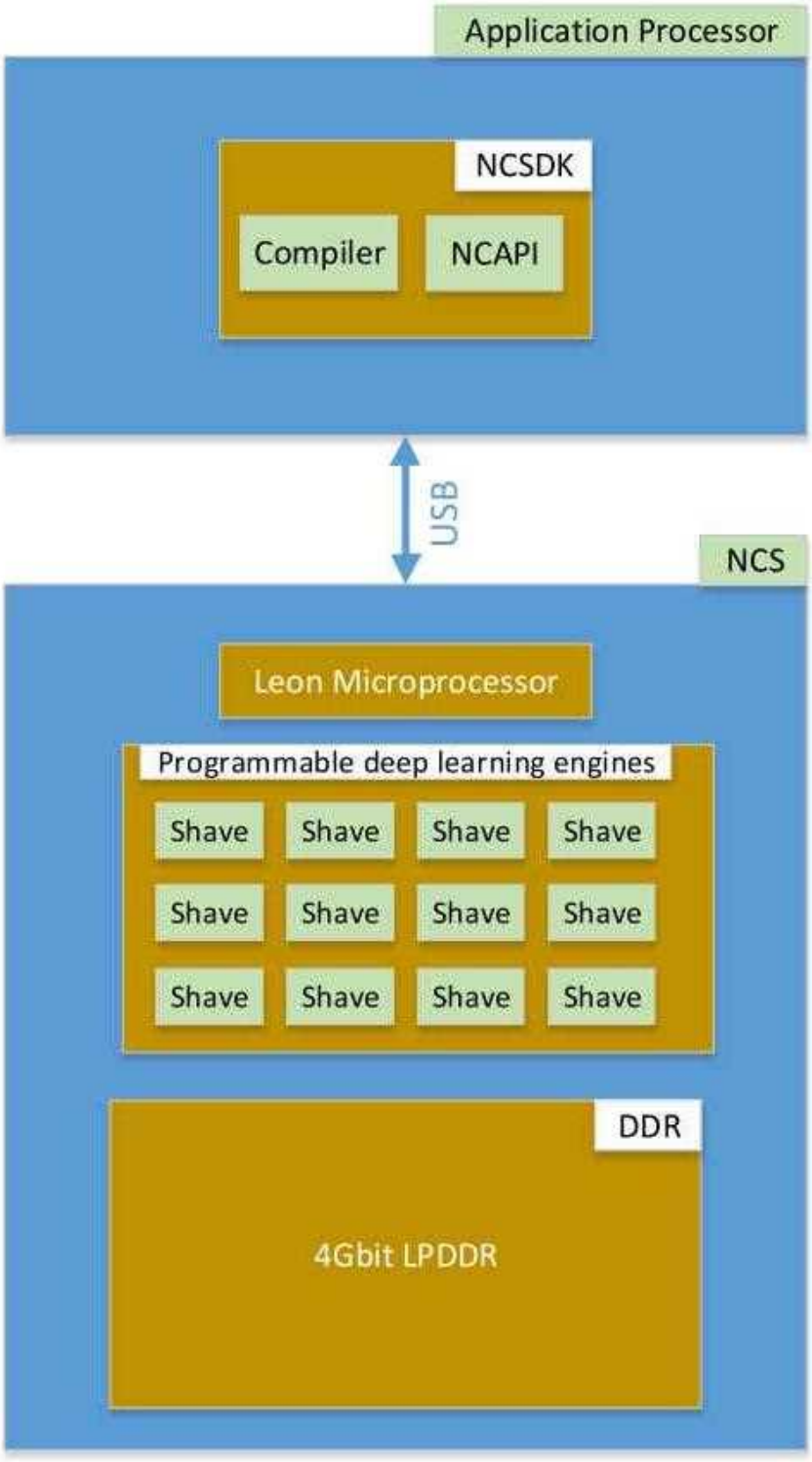

Рис. 2.3 Архітектура Intel NCS 2

OpenVINO

OpenVINO — це кросплатформний інструментарій глибокого навчання, розроблений Intel. OpenVINO зосереджується на оптимізації висновків нейронної мережі за допомогою підходу одноразового запису та розгортання будь-де для апаратних платформ Intel [9]

Особливості OpenVINO:

- Перетворення та оптимізація моделей: OpenVINO надає інструменти для конвертації моделей з популярних фреймворків машинного навчання, таких як TensorFlow, Caffe, MXNet, PyTorch, в оптимізований формат, який може бути ефективно виконаний на апаратних прискорювачах Intel. Це дозволяє забезпечити оптимальну продуктивність для моделей штучного інтелекту.
- Ми можемо виконувати ту саму програму на кількох пристроях. Нам просто потрібно передати цільовий пристрій як аргумент командного рядка, а Inference Engine подбає про решту, тобто ми можемо запустити той самий фрагмент коду на центральному процесорі, графічному процесорі, відеопроцесорі чи будь-якому іншому пристрої, сумісному з набором інструментів OpenVINO [10]
- Оптимізація продуктивності: OpenVINO використовує набір оптимізаційних технік, таких як квантизація, обрізка моделей, об'єднання шарів, для зменшення розміру моделі та збільшення швидкодії. Це дозволяє досягти більш ефективного використання ресурсів пристрою та знизити енергоспоживання.
- Хмарна інтеграція: OpenVINO підтримує хмарну інтеграцію, дозволяючи виконувати інференс моделей на віддалених серверах або хмарних платформах. Це дозволяє розгортати та масштабувати моделі в розподілених середовищах та отримувати доступ до великої обчислювальної потужності.

- Підтримуючи численні моделі глибокого навчання з коробки, набір інструментів прискорює процес створення додатків комп'ютерного бачення, скорочуючи час створення та налаштування. Попередньо підготовлені моделі, які варіюються від фреймворків глибокого навчання, таких як YOLO (You Only Look Once) до R-CNN і ResNet, дозволяють розробникам створювати моделі, які виконують складні програми комп'ютерного бачення, такі як розпізнавання обличчя, виявлення людей, виявлення транспортних засобів, і люди підраховують. Набір Myriad Development Kit (MDK) також містить необхідні інструменти розробки, фреймворки та API для реалізації на чіпі спеціального бачення, зображень і глибокої нейронної мережі. [11]

3 Вимірювання характеристик апаратного акселератора

3.1 Знаходження об'єктів

Першим етапом тестування був стандатний тест визначення, чи є на зображені кішка і, якщо є – класифікувати її.

Для прикладів було взяті такі зображення:

Рис. 3.1 Зображення №1

Рис. 3.27 Зображення №2

Рис. 3.38 Зображення №3

Для аналізу зображення використовується архітектура згорткової нейронної мережі squeezenet1.0

3.2 SqueezeNet

SqueezeNet — це згорткованейронна мережа з 18 рівнями. Попередньо навчена мережа може класифікувати зображення за 1000 категоріями об’єктів, наприклад клавіатура, миша, олівець і багато тварин. У результаті мережа навчилася багатим представленням функцій для широкого діапазону зображень.[12]

Модель squeezenet1.0 є однією з моделей топології SqueezeNet, призначена для класифікації зображень. Моделі SqueezeNet були попередньо навчені на базі даних зображень ImageNet.[13]

Вхідні дані моделі — це об’єкт, який складається з одного зображення BGR. Вихід моделі для squeezenet1.0 є типовим виходом класифікатора об’єктів для 1000 різних класифікацій, які відповідають класифікаціям у базі даних ImageNet. [13]

На основі аналізу зображень, згорткова нейронна мережа робить припущення:

Рис. 3.4 Результат розпізнавання та класифікації першої картинки

Рис. 3.5 Результат розпізнавання та класифікації другої картинки

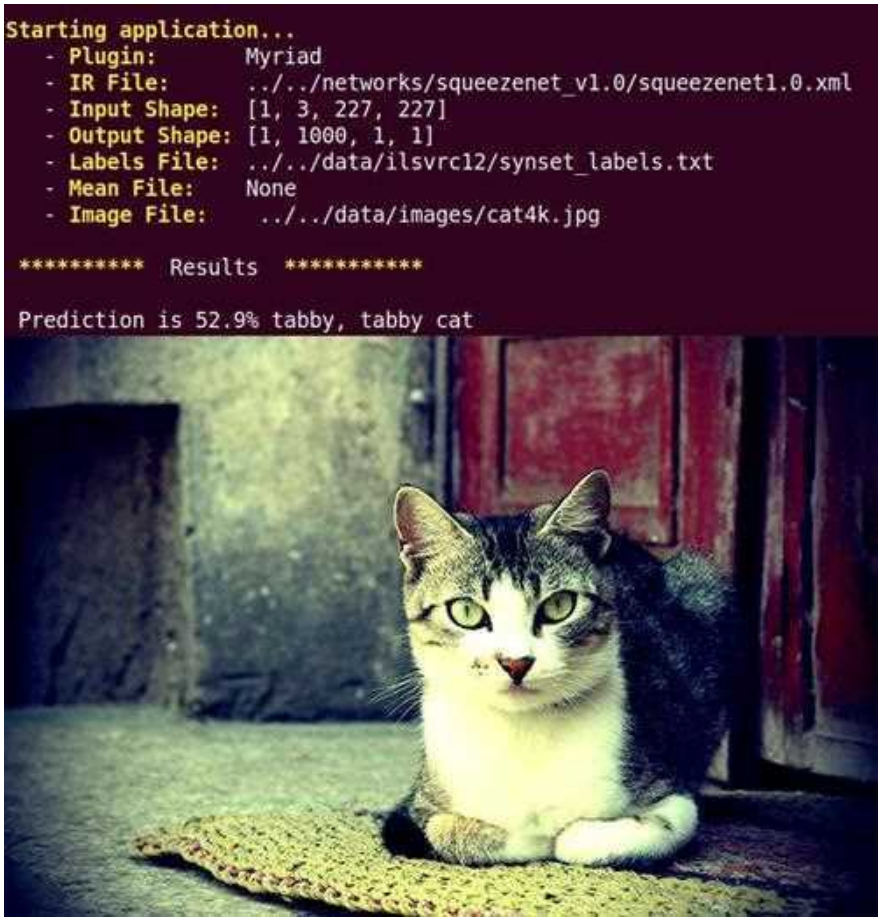

Рис. 3.6 Результат розпізнавання та класифікації третьої картинки

Взагалом, було опрацьовано 55 картинок, з різними характеристиками, результати показано на (Рис 16):

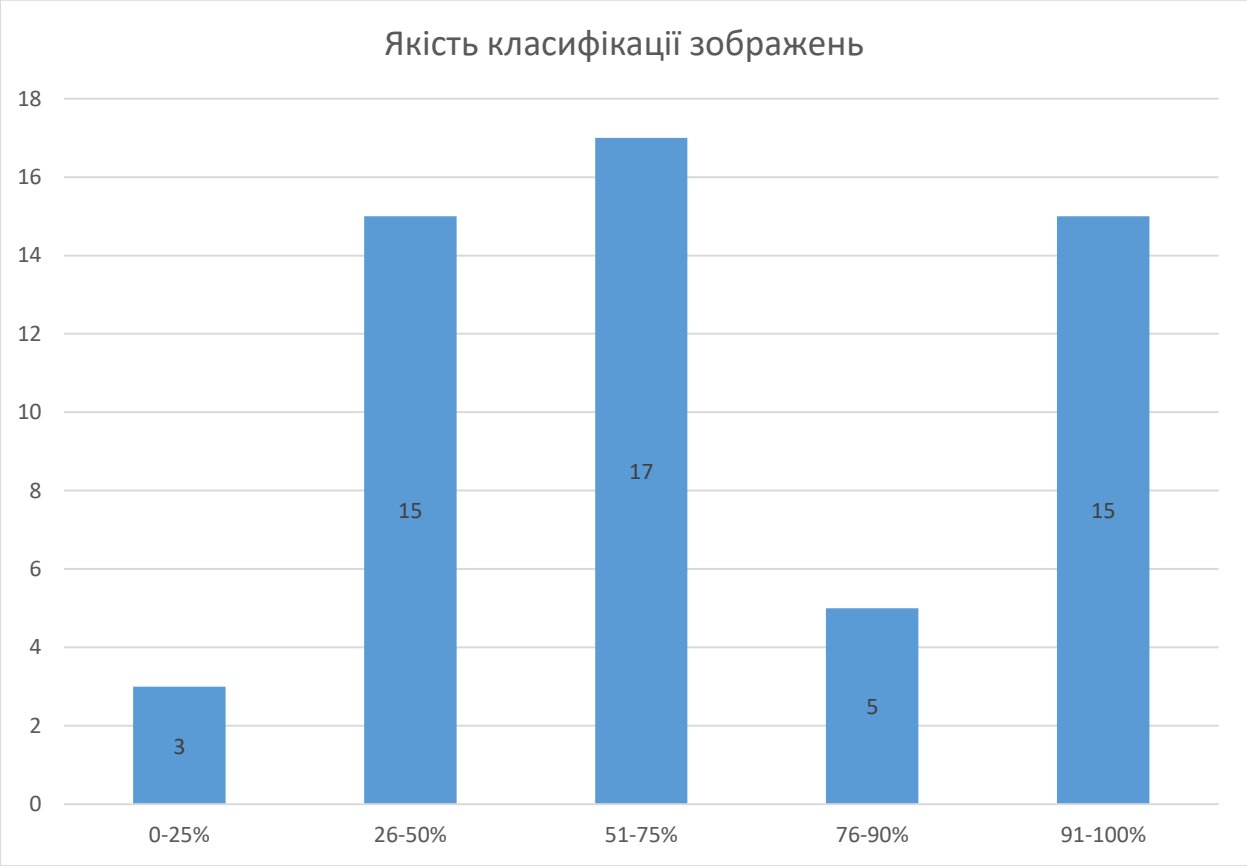

Рис. 3.7 Якість класифікації зображень

3.2 Продуктивність

Також можемо отримати час, за який виконуються усі етапи обробки зображення.

```
Prediction is 52.9% tabby, tabby cat    Prediction is 67.5% tiger cat    Prediction is 46.5% tabby, tabby cat

real    0m6.337s                        real    0m6.273s                 real    0m6.243s
user    0m4.285s                        user    0m4.268s                 user    0m4.139s
sys     0m1.161s                        sys     0m1.087s                 sys     0m1.185s
```

Рис. 3.8 Час виконання

Де real – загальний час виконання, user – час, який програма витратила на обчислення, та sys – час виконання системних операцій.

У випадку класифікації зображень час буде залежати напряму від розміру нашого зображення, на приклад, маємо два зображення з такими

характеристиками:

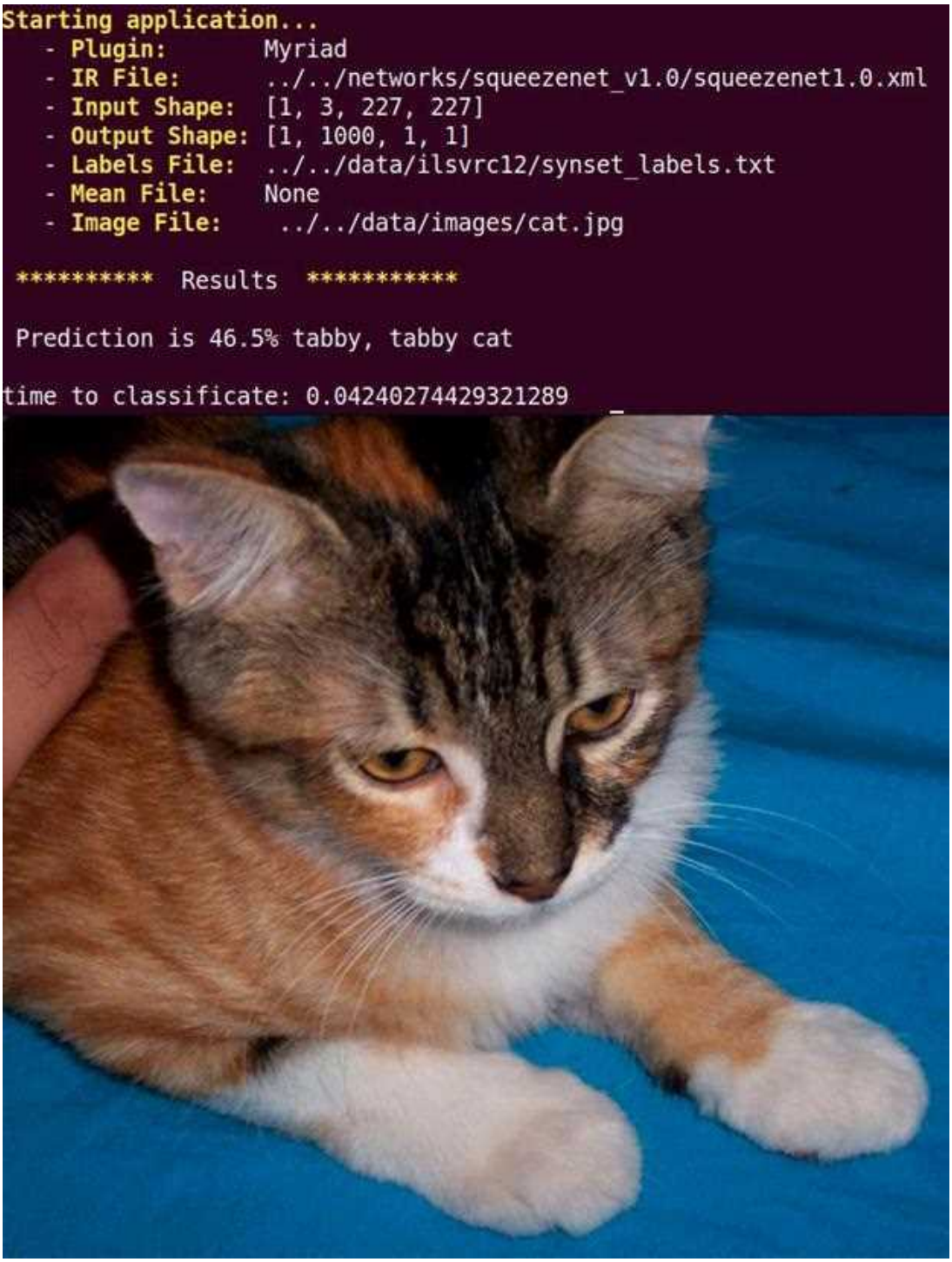

Рис. 3.9 Зображення розміром: 800x712p

Рис. 3.10 Зображення розміром 7952x5304p

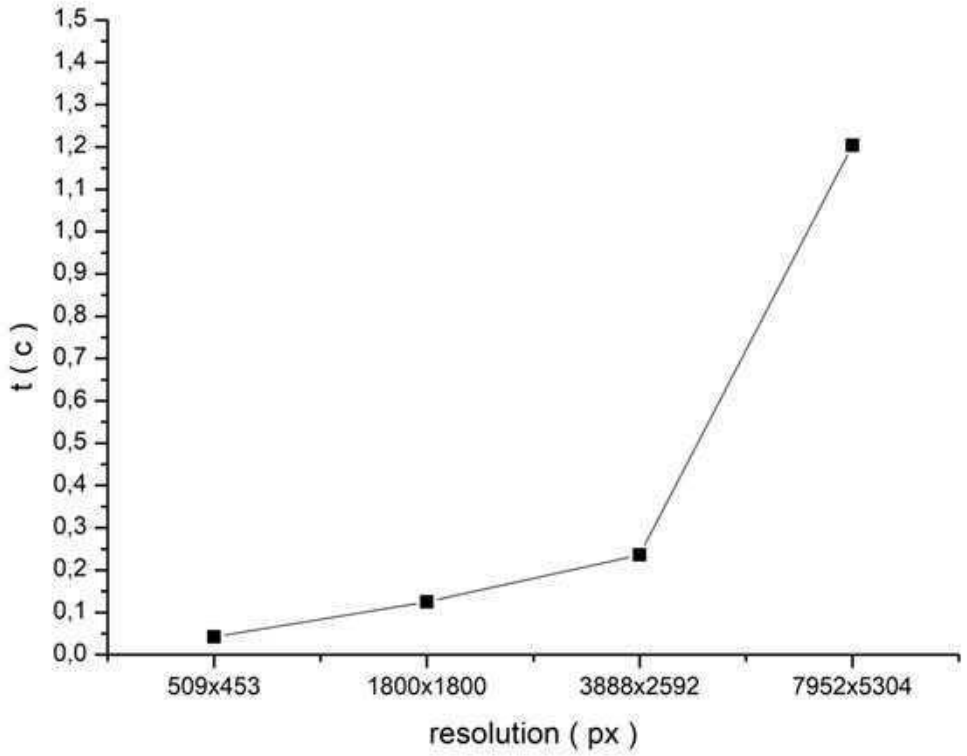

Рис. 3.11 Графік залежності часу розпізнавання від розміру зображення

Із наведеного графіку (Рис.19) бачимо, що час класифікації зображення розміром 800x712 пікселів у 30 разів швидший від зображення розміром 7952x5304. Проте варто відмітити, що зображення з більшою роздільною здатністю класифікувалось всього за 1.2 секунди

Висновки

1. Результати тестування показали, що апаратний акселератор нейромереж Intel NCS 2 дає можливість виконувати класифікацію у реальному часі зображень з роздільною здатність у діапазоні значень вкажіть діапазон. Час класифікації таких зображень лежить у діапазоні від 0.04 с до 1.2 с. Зображення з великою роздільною здатністю 7952x5304 класифікуються до 30 разів повільніше.
2. Результат класифікації сильно залежить від моделі нейронної мережі і у вападку моделі squeezenet1.0 стандартний текст знаходження кота дає 100% якість знаходження об’єктів. Якість класифікації порід котів не перевищує 50%.
3. Intel Neural Compute Stick 2 може бути застосований для класифікації інших даних, зокрема і медичних діагностичних зображень у реальному часі.

Перелік використаних джерел

1. Geeksforgeeks, «Artificial Neural Networks and its Applications»: https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/, 02.06.2023
2. Geeksforgeeks, «Introduction to Convolution Neural Network»: https://www.geeksforgeeks.org/introduction-convolution-neural-network/, 24.03.2023
3. Techtarget, «convolutional neural network (CNN)»: https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network, 04.2023
4. Stanford, «Convolutional Neural Networks»: https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks, 06.01.2019
5. Aigents, «Introduction to Convolutional Neural Networks CNNs»: https://aigents.co/data-science-blog/publication/introduction-to-convolutional-neural-networks-cnns, 08.12.2020
6. TowardsAI, «Introduction To Pooling Layers In CNN»: https://towardsai.net/p/l/introduction-to-pooling-layers-in-cnn#:~:text=A%20Pooling%20layer%20is%20added,or%20average%20of%20the%20input., 16.08.2022
7. Builtin, «Fully Connected Layer vs. Convolutional Layer: Explained»: https://builtin.com/machine-learning/fully-connected-layer, 18.10.2022
8. Intel, «Intel® Neural Compute Stick 2 (Intel® NCS2)»: https://www.intel.com/content/www/us/en/developer/articles/tool/neural-compute-stick.html, 09.05.2020
9. VisoAI, «What is OpenVINO? – The Ultimate Overview in 2023»: https://viso.ai/computer-vision/intel-openvino-toolkit-overview/

10. Towardsdatascience, «Introduction to OpenVINO»: https://towardsdatascience.com/introduction-to-openvino-897e705a1f0a, 28.11.2019

11. VisoAI Intel Neural Compute Stick 2 – AI Vision Accelerator Review 2021 https://viso.ai/edge-ai/intel-neural-compute-stick-2/ , 2021

12. Mathworks, «squeezenet»: https://www.mathworks.com/help/deeplearning/ref/squeezenet.html, 2023

13. Openvino.AI, «squeezenet1.0»: https://docs.openvino.ai/2022.3/omz_models_model_squeezenet1_0.html#use-case-and-high-level-description, 2023