

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України

Кваліфікаційна наукова
праця на правах рукопису

МОСКАЛЕНКО ЮРІЙ ВОЛОДИМИРОВИЧ

УДК 004.032.26

ДИСЕРТАЦІЯ
МЕТОДИ РОЗПІЗНАВАННЯ ЗА ДІАГНОСТИЧНИМ
СИГНАЛОМ НА ОСНОВІ ГІБРИДНИХ НЕЙРОННИХ МЕРЕЖ

12 — Інформаційні технології

122 — Комп'ютерні науки

Подається на здобуття наукового ступеня доктор філософії
Дисертація містить результати власних досліджень. Використання ідей,
результатів і текстів інших авторів мають посилання на відповідне джерело
_____ Ю. В. Москаленко

Науковий керівник:

Шаповалова Світлана Ігорівна, кандидат технічних наук, доцент

Київ — 2020

АНОТАЦІЯ

Москаленко Ю. В. Методи розпізнавання за діагностичним сигналом на основі гібридних нейронних мереж. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії з галузі знань 12 Інформаційні технології за спеціальністю 122 Комп'ютерні науки. – Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, 2020.

Підготовка здійснювалась на кафедрі автоматизації проектування енергетичних процесів і систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського».

Дисертація присвячена розробці засобів підвищення точності розпізнавання за діагностичним сигналом.

Розпізнавання за діагностичним сигналом є базовою проблемою при реалізації систем автоматичного контролю та діагностики. Розповсюдженими задачами є безпосередньо класифікація за характеристичним сигналом, заданим як часовий ряд, та семантична сегментація за діагностичними знімками. З розвитком згорткових нейронних мереж відкрилися нові перспективи розв'язання таких задач швидко, зокрема в реальному часі, та з достатньою точністю.

Однак в багатьох випадках рішення задачі семантичної сегментації потребує дотримання піксельної точності тому, що – в залежності від масштабу – зображення може відповідати розмірам об'єктів різного порядку. Наприклад, у випадках аналізу медичних зображень — це міліметри, а в задачах сейсмічної розвідки Землі — метри. Для потрібної точності

розв'язання задач семантичної сегментації базових алгоритмів мереж глибокого навчання може бути не достатньо.

Крім цього, не для всіх випадків діагностування достатньо точності класифікації, яку можна досягти на згорткових мережах. Насамперед це відбувається при втраті значень характеристичного вектору, виникненні аномалій або початку перехідних процесів в досліджуваному об'єкті. Водночас інший різновид нейронних мереж – карти самоорганізації Кохонена – мають властивості до навчання за невизначеною множиною класів та формування нових кластерів для заздалегідь не відомих класів. Тим самим самим, мережі Кохонена надають можливість розв'язувати задачі зазначеного типу.

Тому в роботі проведено дослідження обох нейромережових підходів для підвищення ефективності кожного з них окремо та їх інтеграції в гібридну нейронну мережу.

Теоретичною базою досліджень стали роботи науковців:

- у галузі мереж глибокого навчання: G. Hinton, Y. LeCun, A. Krizhevsky, I. Goodfellow та інші;

- у галузі нейронних мереж, заснованих на принципах самоорганізації: T. Kohonen, S. Furoo, M. Cao та інші.

Сучасні дослідження спрямовані насамперед на знаходження засобів підвищення точності розв'язання прикладних задач на основі існуючих механізмів нейронних мереж. Однак для багатьох задач класифікації та семантичної сегментації досі не досягнута необхідна точність, тому вдосконалення архітектур та визначення додаткових засобів збільшення точності діагностування є актуальною прикладною задачею.

Метою досліджень є розробка методів підвищення точності розпізнавання за діагностичним сигналом на основі нейронних мереж. Для розв'язання поставленої мети поставлено такі завдання:

- провести аналіз математичного та алгоритмічного забезпечення нейронних мереж, їх архітектур та властивих ним проблем навчання;

- вдосконалити мережі глибокого навчання для підвищення точності розв'язання задачі семантичної сегментації;
- підвищити точність класифікації сигналів великої розмірності на мережах глибокого навчання за умови обмежених обчислювальних ресурсів;
- розробити метод класифікації за характеристичним сигналом з втратами у випадкові моменти часу на основі карт самоорганізації Кохонена;
- розробити концепцію асоційованої пам'яті для мережі глибокого навчання;
- здійснити програмну реалізацію всіх запропонованих методів вдосконалення нейронних мереж;
- експериментально перевірити запропоновані методи при розробці програмного забезпечення розв'язання прикладних задач діагностування.

В дисертаційні роботі вперше отримано такі наукові результати:

Вперше запропоновано:

- метод збільшення рецептивного поля нейронів згорткових нейромереж на основі агрегації карт признаков різної розмірності для підвищення точності класифікації сигналів великої розмірності;
- метод визначення відповідності нейронів решітки навченої карти самоорганізації Кохонена вхідному вектору з втраченими невизначеними компонентами для підвищення точності класифікації;
- модель формування усереднених карт признаков в згорткових нейронних мережах на основі карт самоорганізації Кохонена для підвищення точності розв'язання задач класифікації та семантичної сегментації.

Отримав подальший розвиток:

- метод семантичної сегментації на мережах глибокого навчання за рахунок примусового виділення контурів зображення в декодері мереж:

FPN, PSPNet, DeepLab v3, U-Net та за рахунок агрегації різнорозмірних підвиборок карти признаков для підвищення точності розпізнавання.

У **першому** розділі «Проблеми розв'язання задач діагностування нейронними мережами» формалізовано постановки задач класифікації за діагностичним сигналом та семантичної сегментації за знімками, проведено аналіз нейронних мереж, їх концепцій, архітектур та властивих проблем навчання.

В **другому** розділі «Семантична сегментація на мережах глибокого навчання» розглянуто базові блоки сучасних архітектур нейронних мереж, методи виділення контурів на зображенні. На основі цих досліджень здійснено власну розробку і вдосконалено математичне та алгоритмічне забезпечення сучасних мереж глибокого навчання: U-Net, DeepLabv3, FPN, PSPNet.

В **третьому** розділі «Класифікація на картах самоорганізації за сигналом з втратами» розроблено спосіб збільшення точності розпізнавання вхідного сигналу з втратами на основі карт самоорганізації. Розроблено власні програмні реалізації карт самоорганізації за класичним алгоритмом та алгоритмом, доповненим запропонованим способом. Обґрунтовано вибір існуючих програмних реалізацій мереж Кохонена, реалізованих як на базових, так і на оптимізованих алгоритмах. На основі обчислювальних експериментів з всіма програмними реалізаціями нейронних мереж доведено ефективність запропонованого способу.

В **четвертому** розділі «Середовище моделювання нейронних мереж для розв'язання задачі кластеризації» розроблено середовище моделювання нейронних мереж Кохонена для порівняльного аналізу існуючих програмних реалізацій карт самоорганізацій при розв'язанні задач кластеризації та доведення коректності нових розробок. Запропоноване програмне середовище має практику значущість для обґрунтування вибору програмного інструментарію ефективного розв'язання поточної прикладної задачі, або доведення ефективності нових розробок карт самоорганізацій.

В п'ятому розділі «Класифікація та сегментація на згорткових мережах з використанням карт самоорганізації» представлена концепція інтеграції згорткової мережі з картою самоорганізації Кохонена в єдиному нейромережевому комплексі для підвищення точності. Розроблено блок гібридної згорткової мережі з інтегрованою картою самоорганізації Кохонена. На основі обчислювальних експериментів доведено ефективність запропонованої архітектури.

За матеріалами дисертації **опубліковано** 12 робіт, з яких 5 – це статті у періодичних журналах, що входять до переліку фахових видань, затверджених МОН України за спеціальністю дисертації, в тому числі 1 включена до міжнародної наукометричної бази SCOPUS, та 7 – публікації у матеріалах конференцій (у тому числі, міжнародних).

Ключові слова: згорткові нейронні мережі, карти самоорганізації Кохонена, глибинне навчання, класифікація, семантична сегментація.

SUMMARY

Yu.V. Moskalenko Diagnostic signal recognition methods based on hybrid neural networks. - Qualified scientific work in manuscript copyright.

Dissertation for the earning a degree of Doctor of Philosophy in the knowledge domain 12 Information technology in the specialty 121 Computer Science. - National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute,” Kyiv, 2020.

The preprocessing was carried out at the Department of Automation of energy processes and systems design of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

The dissertation is devoted to the development of ways for improving the accuracy of recognition by a diagnostic signal.

Recognition by a diagnostic signal is a basic problem in the implementation of automatic monitoring and diagnostics systems. Common tasks include the classification based on a characteristic signal specified as a time series, and semantic segmentation based on diagnostic images. With the development of convolutional neural networks, new prospects have opened for solving such problems quickly, in particular in real time and with sufficient accuracy.

However, in many cases, the solution of the conceptual segmentation problem requires pixel accuracy because based on the scale, the image can correspond to real sizes of different order. For example, in cases of analysis of medical images it is represented in millimeters, and in objectives of seismic method of prospecting — in meters. Basic algorithms of deep learning networks may not be enough for the required accuracy of solving conceptual segmentation problems.

Besides, the classification accuracy that can be achieved on these networks is not sufficient enough for convolutional networks. First of all, this occurs when

the values of the characteristic vector are lost, anomalies occur, or transients begin in the object under study. At the same time, another type of neural networks, self-organization maps, have properties for learning by an indefinite set of classes and forming new clusters for previously unknown classes. In the same way, such networks provide the ability to solve problems of the specified type.

Therefore, the study of both neural network approaches to improve the efficiency of each of them separately and their integration into a hybrid neural network.

The theoretical basis of the research includes the works of scientists:

- in the domain of deep learning networks: G. Hinton, Y. LeCun, A. Krizhevsky, I. Goodfellow etc.;
- in the domain of neural networks based on the principles of self-organization: T. Kohonen, S. Furao, M. Sao etc.;

Modern studies are primarily focused on searching ways to improve the accuracy of solving applied problems based on existing mechanisms of neural networks. However, for many problems of classification and semantic segmentation, the required accuracy has not yet been achieved, hence, improving the architecture and determining additional means to increase the accuracy of diagnostics is an urgent experimental problem.

The aim of the research is to improve deep learning neural networks and self-organization maps to improve the accuracy of recognition based on a diagnostic signal. To achieve this aim, the following tasks were set:

- to analyze the mathematical and algorithmic support of neural networks, their architectures and their inherent learning problems;
- to enhance deep learning networks with the purpose of improving the accuracy of solving semantic segmentation problems;
- to enhance deep learning networks with the purpose of improving the accuracy of solving semantic segmentation problems;
- to improve the accuracy of high-dimensional signals classification in deep learning networks with limited computing resources;

- to develop a classification method by a characteristic signal with random losses based on Kohonen's self-organizing maps;
- to develop a concept of associated memory for a deep learning network;
- to effectuate software implementation of all the suggested methods for enhancing neural networks;
- to experimentally verify the suggested methods in the development of software for solving applied diagnostic problems.

The following scientific results were achieved for the first time in the dissertation work:

- a method for implementing an additional block for deep learning networks based on the Sobel operator for forced detection of contours in an image;
- method of classification by characteristic signal with random losses based on self-organization maps by determining incomplete matching of the input vector of the input layer of neurons to increase the proportion of correct recognition;
- a method of association of the self-organizing grid shape reflection with fragments of the input signal for embedding the SOM block in a hybrid convolutional neural network;
- architectural solution for integration of the network with the SOM block based on the association of self-organization grid mapping with fragments of the input signal to improve the accuracy of classification and semantic segmentation.

A further development was achieved:

- architectural solutions for FPN, PSPNet, DeepLab v3, U-Net networks by embedding the proposed Sobel block to improve the accuracy of semantic segmentation;
- architectural solution of U-Net networks by integrating them with the PSP block to improve the accuracy of semantic segmentation.

In the **first** section "Problems of solving problems of diagnostics by neural networks", the problem of classification and semantic segmentation with diagnostic images is formalized and the analysis of neural networks, its concepts, architectures, and its inherent learning problems is performed.

In the **second** section, "Semantic segmentation on deep learning networks", the basic blocks of modern neural networks and methods of contour selection in the image are considered. Based on these studies, mathematical and algorithmic support for deep learning networks has been improved: U-Net, DeepLabv3, FPN, PSPNet.

In the **third** section "Classification on self-organization maps based on the signal with losses", a method for increasing the accuracy of recognition of the input signal with losses based on self-organization maps is developed. A method of classification by characteristic signal with random losses based on self-organization maps has been designed. We also chose and developed our own software implementations of self-organization maps based on the classical algorithm and the algorithm supplemented by the proposed method. Based on computational experiments, the effectiveness of the proposed method is proved.

In the **fourth** section "Neural network modeling environment for solving the clustering problem", the neural network modeling environment is developed for comparative analysis of existing SOM software implementations for solving clustering problems and proving the correctness of new developments. Computational experiments have proved the effectiveness of the proposed environment.

In the **fifth** section, "Classification and segmentation on convolutional networks using self-organization maps", the concept of integrating a network with a self-organization map in a single neural network complex is presented to improve accuracy. The block of hybrid convolutional network architecture with an integrated self-organizing map is developed. Based on computational experiments, the effectiveness of the proposed method is proved.

Based on materials of dissertation 12 papers were **published**, 5 of which are articles in scientific journals included in the list of professional editions approved by MES of Ukraine on the specialty of the dissertation, including a 1 included in the international scientometric databases SCOPUS, and 7 publications in materials of conferences (including international).

Keywords: convolutional neural networks, self-organization maps, deep learning, classification, semantic segmentation.

Зміст

| | |
|---|----|
| АНОТАЦІЯ..... | 2 |
| SUMMARY..... | 7 |
| Перелік умовних позначень..... | 16 |
| ВСТУП..... | 19 |
| РОЗДІЛ 1 ПРОБЛЕМИ РОЗВ’ЯЗАННЯ ЗАДАЧ ДІАГНОСТУВАННЯ НЕЙРОННИМИ МЕРЕЖАМИ..... | 26 |
| 1.1. Постановка задач розпізнавання..... | 27 |
| 1.2. Згорткові нейронні мережі..... | 35 |
| 1.2.1. Концепція згорткових нейронних мереж..... | 35 |
| 1.2.2. Проблеми згорткових нейронних мереж та засоби їх вирішення...36 | 36 |
| 1.2.3. Проблема підвищення точності згорткових нейронних мереж.....45 | 45 |
| 1.3. Архітектури нейронних мереж призначених для семантичної сегментації..... | 46 |
| 1.3 Нейронні мережі Кохонена..... | 48 |
| Висновки до розділу 1..... | 52 |
| РОЗДІЛ 2 СЕМАНТИЧНА СЕГМЕНТАЦІЯ ТА КЛАСИФІКАЦІЯ НА МЕРЕЖАХ ГЛИБОКОГО НАВЧАННЯ..... | 53 |
| 2.1. Згорткові нейронні мережі..... | 54 |
| 2.1.1. Базові блоки згорткових нейронних мереж..... | 54 |
| 2.1.2 Механізми оптимізації згорткових нейронних мереж..... | 60 |
| 2.2. Семантична сегментація..... | 62 |
| 2.2.1. Підвищення точності семантичної сегментації..... | 62 |
| 2.2.2. Покращення блоку декодери мережі U-Net..... | 67 |
| 2.2.3. Метрики якості в задачах семантичної сегментації..... | 69 |
| 2.3. Обчислювальні експерименти з розв’язання задачі семантичної сегментації..... | 72 |
| 2.3.1. Тестові задачі..... | 72 |
| 2.3.2. Обчислювальні експерименти на мережах U-Net та FPN..... | 74 |
| 2.3.3. Обчислювальні експерименти на мережах PSPNet та DeepLab v3..... | 77 |

| | |
|---|-----------|
| 2.3.4. Обчислювальні експерименти на мережах з модифікованим декодером..... | 78 |
| 2.4. Класифікація сигналів великої розмірності..... | 81 |
| 2.4.1. Згорткові нейронні мережі, пристосовані для розв'язання задач класифікації..... | 82 |
| 2.4.2. Блок агрегації сигналів різної розмірності..... | 83 |
| 2.4.3. Архітектура модифікованої мережі..... | 84 |
| 2.5. Обчислювальні експерименти з розв'язання задачі класифікації сигналів великої розмірності..... | 85 |
| 2.5.1. Мережі для проведення експериментів..... | 85 |
| 2.5.2. Задача класифікації 1D-сигналів..... | 87 |
| 2.5.2. Задача класифікації 2D-сигналів..... | 90 |
| Висновки до розділу 2..... | 91 |
| РОЗДІЛ 3 КЛАСИФІКАЦІЯ СИГНАЛУ З ВТРАТАМИ НА КАРТАХ САМООРГАНІЗАЦІЇ..... | 93 |
| 3.1. Постановка проблеми..... | 94 |
| 3.1.2. Проблеми розпізнавання за сигналом з втратами..... | 95 |
| 3.2. Класифікація за характеристичним сигналом з випадковими втратами на основі SOM..... | 101 |
| 3.2.1. Постановка задачі для нейромережевого комплексу..... | 101 |
| 3.2.2. Визначення відповідності компонентів вхідного вектору нейронам вхідного шару SOM..... | 104 |
| 3.3. Етапи розв'язання задачі класифікації за неповним сигналом..... | 107 |
| 3.4. Математична модель формування прикладів повних сигналів навчальної вибірки..... | 108 |
| 3.5. Реалізації нейронних мереж SOM..... | 108 |
| 3.6. Верифікація реалізацій базових алгоритмів SOM та MLP..... | 110 |
| 3.7. Обчислювальні експерименти з класифікації за сигналом з випадковими втратами..... | 112 |
| 3.8. Аналіз результатів обчислювальних експериментів..... | 114 |

| | |
|--|-----|
| | 14 |
| Висновки до розділу 3..... | 116 |
| Розділ 4 Середовище моделювання нейронних мереж для розв’язання задачі кластеризації..... | 118 |
| 4.1. Програмні реалізації SOM..... | 119 |
| 4.2. Постановка задачі моделювання для нейронної мережі SOM..... | 121 |
| 4.3. Архітектура середовища..... | 123 |
| 4.4. Обчислювальні експерименти..... | 125 |
| 4.4.1. Графічний інтерфейс користувача для розв’язання задач на класичних наборах даних..... | 125 |
| 4.4.2. Обчислювальні експерименти на класичних наборах даних..... | 127 |
| 4.4.3. Графічний інтерфейс для моделювання втрат вхідного сигналу..... | 130 |
| 4.4.4. Обчислювальні експерименти з моделювання вхідного сигналу з втратами..... | 132 |
| Висновки до розділу 4..... | 134 |
| РОЗДІЛ 5 КЛАСИФІКАЦІЯ ТА СЕГМЕНТАЦІЯ НА ЗГОРТКОВИХ МЕРЕЖАХ З ВИКОРИСТАННЯМ КАРТ САМООРГАНІЗАЦІЇ..... | 135 |
| 5.1. Проблема інтеграції згорткових нейронних мереж з картами самоорганізації Кохонена..... | 136 |
| 5.2. Базова концепція SOM..... | 139 |
| 5.3. Реалізація блоку SOM в мережі глибокого навчання..... | 143 |
| 5.4. Архітектура гібридної мережі для розв’язання задачі класифікації..... | 144 |
| 5.5. Архітектура гібридної мережі для розв’язання задач сегментації..... | 146 |
| 5.6. Обчислювальні експерименти для розв’язання задачі класифікації та сегментації..... | 150 |
| 5.6.1. Тестові задачі..... | 150 |
| 5.6.2. Обчислювальні експерименти розв’язання задачі класифікації..... | 151 |
| 5.6.3 Обчислювальні експерименти розв’язання задачі сегментації..... | 153 |
| Висновки до розділу 5..... | 158 |
| ВИСНОВКИ..... | 159 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... | 161 |

| | |
|--------------|-----|
| Додатки..... | 177 |
|--------------|-----|

Перелік умовних позначень

Абревіатури

| | |
|--------|--|
| SOM | Карта самоорганізації Кохонена (Self-organizing map) |
| MLP | Багатошаровий перцептрон (Multilayer perceptron) |
| CNN | Згорткова нейронна мережа (Convolutional neural network) |
| GAP | Глобальний середній пулінг (Global average pooling) |
| IoU | Відношення площі перекриття областей до площі об'єднань областей (Intersection-over-Union) |
| FPN | Різновид мережі — Feature Pyramid Network |
| PSPNet | Різновид мережі — Pyramid Scene Parsing Network |
| BCE | Бінарна крос-ентропія (binary cross entropy) |
| CCE | Категорійна крос-ентропія (categorical cross entropy) |
| TTA | Аугментація під час тесту (test time augmentation) |
| ReLU | функцію активації “випрямляч” (Rectified Linear Units) |

Задачі класифікації та сегментації

| | |
|--------------------|--|
| S_X | Множина вхідних сигналів |
| N_{Total} | Кількість сигналів |
| X | Вхідний вектор |
| x_i | i -ий елемент вхідного сигналу |
| S_X^m | Навчальна вибірка |
| N_{Train} | Кількість прикладів в навчальній вибірці |
| S_A | Кінцева множину ідентифікаторів класів |

| | |
|------------|--|
| A | Множина класів |
| A_i | i-й клас |
| d | Кількість класів |
| \bar{X}' | Результат розв'язання задачі семантичної сегментації |
| ShapeSegm | Розмірність \bar{X}' і \bar{X} сигналу сегментації |
| FoldCnt | Кількість вибірок розбиття |
| P | Кількість класів, які класифікатор вірно розпізнав |
| Accuracy | Метрика точності |
| Cor | Частка коректного розпізнавання |
| Y_p | Ймовірність відповідності p-ого класу поточному вхідному вектору |

Карта самоорганізації

| | |
|---------------|--|
| N | Вхідний шар мережі Кохонена |
| N_j | j-ий вхід нейромережі SOM |
| D | Значення помилки (різниця між векторами) |
| t_i | i-й момент часу вхідного сигналу |
| Wsom | Матриця виходів SOM |
| W_j | Ваги j-ого нейрону |
| C | Комбінація всіх x_i , $i=1..l$, та n_j , $j=1..k$. |
| C_p | Поточна комбінація всіх x_i , $i=1..l$, та n_j , $j=1..k$. |
| ε | Значення помилки |
| x_i^g | Нероздільна група значень вхідного сигналу |
| l | Довжина групи значень вхідного сигналу |
| $\eta(n)$ | параметр швидкості навчання |
| Δ | поріг, при якому буде створено новий кластер решітки SOM |
| $h_{j,i}$ | топологічна околиця кластеру на решітці SOM |

Перцептрон

 X_{MLP}

Вхідний вектор MLP

ВСТУП

Актуальність роботи. Розпізнавання за діагностичним сигналом є базовою проблемою при реалізації систем автоматичного контролю та діагностики. Розповсюдженими задачами є безпосередньо класифікація за характеристичним сигналом, заданим як часовий ряд, та семантична сегментація за діагностичними знімками. З розвитком згорткових нейронних мереж відкрилися нові перспективи розв'язання таких задач швидко, зокрема в реальному часі, та з достатньою точністю.

Однак в багатьох випадках рішення задачі семантичної сегментації потребує дотримання піксельної точності тому, що – в залежності від масштабу – зображення може відповідати розмірам об'єктів різного порядку. Наприклад, у випадках аналізу медичних зображень — це міліметри, а в задачах сейсмічної розвідки Землі — метри. Для потрібної точності розв'язання задач семантичної сегментації базових алгоритмів мереж глибокого навчання може бути не достатньо.

Крім цього, не для всіх випадків діагностування достатньою точності класифікації, яку можна досягти на згорткових мережах. Насамперед це відбувається при втраті значень характеристичного вектору, виникненні аномалій або початку перехідних процесів в досліджуваному об'єкті. Водночас інший різновид нейронних мереж – карти самоорганізації Кохонена – мають властивості до навчання за невизначеною множиною класів та формування нових кластерів для заздалегідь не відомих класів. Тим самим самим, мережі Кохонена надають можливість розв'язувати задачі зазначеного типу.

Тому в роботі проведено дослідження обох нейромережових підходів для підвищення ефективності кожного з них окремо та їх інтеграції в гібридну нейронну мережу.

Теоретичною базою досліджень стали роботи науковців:

- у галузі мереж глибокого навчання: G. Hinton, Y. LeCun, A. Krizhevsky, I. Goodfellow та інші;

- у галузі нейронних мереж, заснованих на принципах самоорганізації: T. Kohonen, S. Furoo, M. Cao та інші.

Сучасні дослідження спрямовані насамперед на знаходження засобів підвищення точності розв'язання прикладних задач на основі існуючих механізмів нейронних мереж. Однак для багатьох задач класифікації та семантичної сегментації досі не досягнута необхідна точність, тому вдосконалення архітектур та визначення додаткових засобів збільшення точності діагностування є актуальною прикладною задачею.

Зв'язок роботи з науковими програмами, планами, темами.

Дисертаційна робота виконувалася у Національному технічному університеті України «Київський політехнічний інститут імені Ігоря Сікорського» згідно з планом науково-дослідних робіт кафедри автоматизації проектування енергетичних процесів та систем в рамках науково-дослідних робіт: «Автоматизація моніторингу гідрохімічного стану підземних вод АЕС», номер державної реєстрації 0115U000329; «Інтелектуальна обробка графічної інформації», номер державної реєстрації 0117U006081.

Мета і завдання дослідження. є підвищення точності розпізнавання за діагностичним сигналом на основі нейронних мереж.

Для розв'язання поставленої мети у дисертаційній роботі розв'язуються такі завдання:

- провести аналіз математичного та алгоритмічного забезпечення нейронних мереж, їх архітектур та властивих ним проблем навчання;
- вдосконалити мережі глибокого навчання для підвищення точності розв'язання задачі семантичної сегментації;
- підвищити точність класифікації сигналів великої розмірності на мережах глибокого навчання за умови обмежених обчислювальних ресурсів;

- розробити метод класифікації за характеристичним сигналом з втратами у випадкові моменти часу на основі карт самоорганізації Кохонена;
- розробити концепцію асоційованої пам'яті для мережі глибокого навчання;
- здійснити програмну реалізацію всіх запропонованих методів вдосконалення нейронних мереж;
- експериментально перевірити запропоновані методи при розробці програмного забезпечення розв'язання прикладних задач діагностування.

Об'єктом дослідження є розпізнавання на мережах глибокого навчання та картах самоорганізації Кохонена.

Предметом дослідження є методи вирішення задачі класифікації та семантичної сегментації на мережах глибокого навчання та картах самоорганізації Кохонена.

Методи дослідження. Для проведення досліджень застосовуються методи машинного навчання за концепціями навчання з вчителем та без вчителя, методи виокремлення контурів на зображенні, методи семантичної сегментації.

Наукова новизна одержаних результатів полягає в наступному:

Вперше запропоновано:

- метод збільшення рецептивного поля нейронів згорткових нейромереж на основі агрегації карт признаков різної розмірності для підвищення точності класифікації сигналів великої розмірності;
- метод визначення відповідності нейронів решітки навченої карти самоорганізації Кохонена вхідному вектору з втраченими невизначеними компонентами для підвищення точності класифікації;
- модель формування усереднених карт признаков в згорткових нейронних мережах на основі карт самоорганізації Кохонена для підвищення точності розв'язання задач класифікації та семантичної сегментації.

Отримав подальший розвиток:

- метод семантичної сегментації на мережах глибокого навчання за рахунок примусового виділення контурів зображення в декодері мереж: FPN, PSPNet, DeepLab v3, U-Net та за рахунок агрегації різнорозмірних підвиборок карти признаков для підвищення точності розпізнавання.

Практичне значення одержаних результатів полягає в тому, що розроблено:

- способи підвищення точності розпізнавання на нейронних мережах за характеристичними показниками в системах автоматичного діагностування для таких галузей, як медицина, сейсмологія, автомобілебудування та інші;
- середовище моделювання карт самоорганізацій Кохонена для визначення програмного інструментарію розв'язання поточної задачі діагностування, оптимального за критеріями точності та часу розв'язання задачі.

способи підвищення точності розпізнавання на нейронних мережах за характеристичними показниками в системах автоматичного діагностування для таких галузей, як медицина, сейсмологія, автомобілебудування та інші;

Впровадження одержаних результатів виконані в: 1) КПІ ім. Ігоря Сікорського для виконання науково-дослідних робіт; 2) платформа змагань з машинного навчання Kaggle (<https://www.kaggle.com/yuramuv>) для участі в змаганнях: Elo Merchant Category Recommendation (віртуальна срібна медаль), Quick, Draw! Doodle Recognition Challenge (віртуальна бронзова медаль), TGS Salt Identification Challenge (віртуальна бронзова медаль); University of Liverpool Ion Switching (віртуальна срібна медаль), SIIM-ISIC Melanoma Classification (віртуальна бронзова медаль); 3) КПІ ім. Ігоря Сікорського в навчальному процесі на кафедрі автоматизації проектування енергетичних процесів і систем при викладанні дисципліни «Програмування паралельних та розподілених обчислень - 1» підготовки магістрів

спеціальності 122 «Комп'ютерні науки» освітньо-наукової програми «Комп'ютерний моніторинг та геометричне моделювання процесів та систем».

Особистий внесок здобувача. Всі основні результати дисертаційної роботи отримані автором особисто. Одноосібно опубліковані праці [53, 85, 89, 96]. У друкованих працях, опублікованих у співавторстві, автору належить: [65, 66] — способи підвищення точності семантичної сегментації, [83-85, 90] — спосіб розпізнавання сигналу зі втратами, [95] - спосіб інтеграції карти самоорганізації в згорткову нейронну мережу.

Апробація результатів дисертації. Результати та основні положення роботи доповідалися та обговорювалися на: XIII, XVI, XVII міжнародній науковій конференції 'Сучасні проблеми наукового забезпечення енергетики', Київ, 2015, 2017, 2019; II, VI науково-практична конференція молодих вчених і спеціалістів 'Сучасні аспекти розробки програмного забезпечення', Київ, 2015, 2019; 19-а Міжнародна науково-практична конференція SAIT-2017 'Системний аналіз та інформаційні технології'; V-а Міжнародна Науково-технічна конференція 'Комп'ютерне моделювання та оптимізація складних систем'.

Публікації. За матеріалами дисертації опубліковано 12 робіт, з яких 4 – це статті у періодичних журналах, що входять до переліку фахових видань, затверджених МОН України за спеціальністю дисертації, в тому числі а 1 включена до міжнародної наукометричної бази SCOPUS, 1 стаття у науковому періодичному виданні іншої держави, та 7 – публікації у матеріалах конференцій (у тому числі, міжнародних).

Статті в наукових фахових виданнях України

(які входять до переліку ВАК/МОН України):

1. Shapovalova S., Moskalenko Y. Semantic segmentation accuracy improvement based on forced edge detection. *Адаптивні системи*

автоматичного управління. 2019. Т. 2. №35. С.79-87. DOI: 10.20535/1560-8956.35.2019.197435

2. Москаленко Ю. В. Середовище моделювання нейронних мереж для розв'язання задачі кластеризації. *Математичне та комп'ютерне моделювання. Серія: Технічні науки*. 2019. Вип. 20. С. 68-78. DOI: <https://doi.org/10.32626/2308-5916.2019-20.79-87>

3. Shapovalova S., Moskalenko Y. Rate increase of the objects classification on the convolutional neural networks with the self-organization maps implementation. *Сучасні проблеми моделювання*. 2020. Вип.17. С.145-155. DOI: 10.33842/2313-125X/2019/17/145/155.

*Статті в наукових фахових виданнях України,
які входять до міжнародних наукометричних баз даних
(за умови, що на момент публікації статті видання вже внесені до
міжнародної наукометричної бази):*

4. Shapovalova, S., Moskalenko, Yu. Increasing the share of correct clustering of characteristic signal with random losses in self-organizing maps. *Eastern-European Journal of Enterprise Technologies*. 2019. № 2/4(98). P. 13-21. DOI: 10.15587/1729-4061.2019.160670 (**SCOPUS, категорія A**)

Статті в наукових періодичних виданнях інших держав:

5. Shapovalova S., Moskalenko Y. Methods for increasing the classification accuracy based on modifications of the basic architecture of convolutional neural networks. *ScienceRise*. 2020. No 6 (71). P. 10–16. DOI:10.21303/2313-8416.2020.001550

Статті у збірниках наукових праць:

6. Shapovalova S., Moskalenko Y. Segmentation based problem solving on convolutional neural network. *Modern Aspects of Software Development:*

Proceedings of VI International Scientific and Practical Conference of Software Specialists, June, 24, 2019. Kyiv: Igor Sikorsky KPI. 2019. P.47-53.

Тези доповідей:

7. Москаленко Ю.В., Шаповалова С.І. Класифікація кривих другого порядку по неповному вхідному вектору. *Сучасні проблеми наукового забезпечення енергетики: матеріали XIII Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, Київ, 21-24 квітня 2015 р. у 2 т. - К: НТУУ „КПІ”, 2015.- Т.2.- С.129.*

8. Москаленко Ю.В. Шаповалова С.І. Середовище моделювання карт самоорганізації для розпізнавання геометрично схожих фігур *Сучасні аспекти розробки програмного забезпечення: Збірник тез II науково-практичної конференції молодих вчених і спеціалістів в області розробки програмного забезпечення, Київ, 22 травня 2015 р. Черкаси: видавець Чабаненко Ю.А., 2015.- С. 51-53.*

9. Москаленко Ю.В. Кластеризація навченої карти самоорганізації. *Сучасні проблеми наукового забезпечення енергетики: матеріали XVI Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, Київ, 25-28 квітня 2017 р. у 2 т. - К: НТУУ „КПІ”, 2017.- Т.2.- С.171.*

10. Шаповалова С.І., Мажара О.О., Москаленко Ю.В. Екстракція правил виведення нейронних мереж з представленням в форматі CLIPS. *Системний аналіз та інформаційні технології: матеріали 19-ї Міжнародної науково-практичної конференції SAIT-2017, Київ, 22-25 травня 2017 р. - К: ННК „ІПСА” НТУУ „КПІ ім. Ігоря Сікорського”, 2017. С.192-193.*

11. Москаленко Ю.В. Класифікація зашумлених діагностичних сигналів. *Сучасні проблеми наукового забезпечення енергетики: матеріали XVII Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, Київ, 23-26 квітня 2019 р. у 2 т. - К: НТУУ „КПІ”, 2019.- Т.2. С. 92.*

12. Москаленко Ю.В. Гібридні нейронні мереж на основі карт самоорганізації Кохонена *Комп'ютерне моделювання та оптимізація складних систем: матеріали V-ї Міжнар. Наук.-техн. Конф.*, 6-8 листоп. 2019. Дніпро: Баланс-клуб. 2019. С. 133-134. DOI: 10.32434/cmocs-2019

Структура та обсяг дисертації. Дисертація складається із анотації, вступу, п'яти розділів, висновків, списку використаних джерел та додатків. Робота містить 175 сторінок, у тому числі: 132 сторінки основного тексту, 54 рисунки, 20 таблиць, список використаних джерел налічує 105 найменувань.

РОЗДІЛ 1

ПРОБЛЕМИ РОЗВ'ЯЗАННЯ ЗАДАЧ ДІАГНОСТУВАННЯ НЕЙРОННИМИ МЕРЕЖАМИ

Дослідження цього розділу спрямовані на визначення підходів та властивих ним засобів розв'язання задач розпізнавання за діагностичним сигналом. Для цього вирішено такі задачі:

1. Проведено аналіз задач діагностування, що потребують збільшення точності розв'язання на основі існуючих нейромережових методів. Наведено формалізацію постановки задач класифікації та семантичної сегментації.

2. Проведено аналіз розв'язання базових проблем навчання та функціонування згорткових нейронних мереж.

3. Проведено аналіз архітектур та алгоритмічного забезпечення згорткових нейронних мереж.

4. Проведено аналіз модифікацій та вдосконалень нейронних мереж Кохонена.

В розділі обґрунтовано використання згорткових нейронних мереж та карт самоорганізації Кохонена для розв'язання задач діагностування за показниками відповідної апаратури.

1.1. Постановка задач розпізнавання

Базовою задачею діагностування є розпізнавання стану об'єкту або процесу. У випадку, коли заздалегідь відомо всі можливі діагнози-класи, задача розпізнавання зводиться до задачі класифікації за діагностичним сигналом з апаратури.

Діагностичні сигнали з апаратури обмежимо:

- показниками з датчиків, які утворюють часові ряди;
- діагностичними знімками.

Результатом розв'язку задачі діагностування є визначення діагностичного класу за поточним сигналом апаратури або виокремлення фрагменту вхідного сигналу, який відповідає певному класу.

Роботу присвячено розв'язанню задач:

- класифікації;
- семантичної сегментації.

Класифікація — задача визначення класу за характеристичним сигналом із задалегідь відомої множини.

На основі визначення [1] формально задачу класифікації поставимо таким чином:

нехай S_X — множина вхідних сигналів

$$S_X = [X_1, X_2, \dots, X_i, \dots, X_{N_{Total}}], \quad (1.1)$$

де X_i — i -й приклад вхідного сигналу,

N_{Total} — кількість сигналів.

В свою чергу кожний вхідний сигнал складається з компонентів:

$$X = [x_1, x_2, \dots, x_j, \dots, x_n], \quad (1.2)$$

де x_j — значення вхідного сигналу в j -ий момент часу або j -ого пікселю.

n — довжина вхідного сигналу.

Варто зауважити, що вхідний сигнал X може мати різну розмірність, тобто бути як 1D-вектором, так і 2D- або 3D-зображенням.

Також задано S_A — кінцеву множину ідентифікаторів класів

$$S_A = [A_1, A_2, \dots, A_i, \dots, A_{N_{Total}}], \quad (1.3)$$

де A_i — i -й ідентифікатор класу,

N_{Total} — кількість ідентифікаторів класів.

Кожен з класів A_i належить множині

Розмірність поточного вхідного вектору

$$A_i \in \{a_1, a_2, \dots, a_i, \dots, a_d\}, \quad (1.4)$$

де a_i — i -й ідентифікатор класу,

d — кількість класів.

Існує невідома залежність-відображення y^*_{cl} :

$$y^*_{cl}: Sx \rightarrow S_A, \quad (1.5)$$

де Sx — множина вхідних сигналів, що відповідає (1.1),

S_A — множина ідентифікаторів класів (1.3).

При постановці задачі відображення y^*_{cl} відоме тільки на об'єктах навчальної вибірки Sx^m :

$$Sx^m = (\bar{X}_1, \bar{A}_1), (\bar{X}_2, \bar{A}_2), \dots, (\bar{X}_i, \bar{A}_i), \dots, (\bar{X}_{NTrain}, \bar{A}_{NTrain}), \quad (1.6)$$

де \bar{X}_i — i -й вхідний сигнал навчальної вибірки,

\bar{A}_i — i -й ідентифікатор класу навчальної вибірки,

$NTrain$ — кількість прикладів навчальної вибірки.

Потрібно побудувати алгоритм α_{cl} , який дозволяє класифікувати довільний об'єкт:

$$\alpha_{cl}: Sx \rightarrow A, \quad (1.7)$$

де Sx — множина, що відповідає (1.1),

A — множина ідентифікаторів класів (1.4),

Розв'язання задачі класифікації схематично представлено на рисунку 1.1. Входом системи класифікації є діагностичний сигнал, виходом — розподіл ймовірностей (probability distributions), за яким визначається поточний клас.

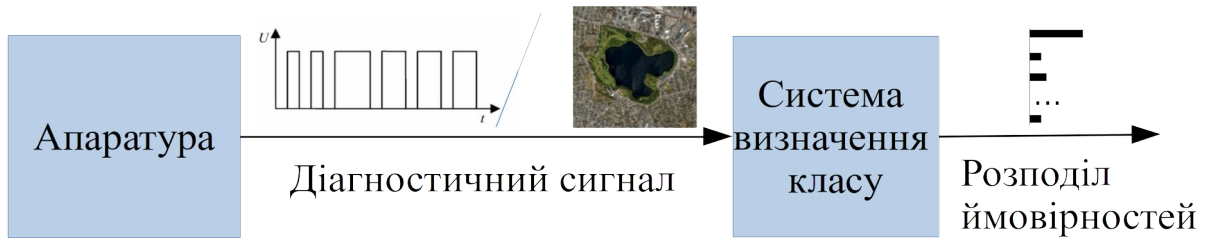


Рисунок 1.1. Схема розв'язання задачі класифікації за діагностичним сигналом

На рисунку 1.1 відображено те, що вхідний сигнал може бути заданий як часовий ряд або 2D- діагностичний знімок. В приведеній схемі мається на увазі, що система заздалегідь пройшла всі етапи навчання.

Задача класифікації є однією з найбільш розповсюджених для розв'язання в прикладних галузях. Приклади таких сучасних задач наведено в таблиці 1.1

Таблиця 1.1. Приклади задач класифікації

| № | Класифікація | Сигнал | Класи | Посилання |
|---|--------------------------------------|--|----------------------------------|-----------|
| Класифікація за часовим рядом | | | | |
| 1 | Визначення пульсарів | Частотно-часовий знімок радіотелескопу | Пульсар/не пульсар | [2], 2019 |
| 2 | Класифікація гравітаційних хвиль | Дані з датчика LIGO-обсерваторії | Сигнал/шум | [3], 2019 |
| Класифікація за діагностичними знімками | | | | |
| 3 | Класифікація зображень клітин людини | Знімки зображення клітин | Тип клітини: нормальна/проблемна | [4], 2019 |

Продовження таблиці 1.1

| | | | | |
|---|--|------------------|---|-----------|
| 4 | Визначення хвороби Альцгеймера на основі МРТ-знімків | МРТ-знімки мозку | Захворювання на ранніх стадіях / його відсутність | [5], 2019 |
| 5 | Визначення діабетическої ретинопатії | Знімок ока | Відсутність / стадія захворювання | [6], 2019 |

Специфікою даної роботи є умова проведення класифікації безпосередньо за сигналом, отриманим з апаратури без додаткової попередньої обробки (наприклад, для визначення частотних характеристик).

Семантична сегментація — розбиття сигналу на окремі регіони, кожен з яких відповідає тільки одному класу. Семантична сегментація працює з об'єктами одного класу як з одним цілим, на відміну від інстанс-сегментації, яка відрізняє різні об'єкти одного класу. Як вхідний сигнал розглядалося 2D-зображення, що відповідає знімку, переданому з діагностичної апаратури. Часто задачу семантичної сегментації розглядають як класифікацію кожного окремого пікселя (вокселя у випадку 3D-зображення).

За аналогією з задачею класифікації, формально задачу сегментації можна визначити таким чином:

нехай S_X — множина вхідних сигналів

$$S_X = [X_1, X_2, \dots, X_i, \dots, X_{N_{Total}}], \quad (1.8)$$

де X_i — i -й приклад вхідного сигналу;

N_{Total} — кількість сигналів.

В свою чергу кожний вхідний сигнал складається з компонентів:

$$\bar{X} = [x_1, x_2, \dots, x_j, \dots, x_n], \quad (1.9)$$

де x_i — значення вхідного сигналу в i -ому компоненті.

n — довжина вхідного сигналу.

Як і в задачі класифікації, вхідний сигнал може мати різну розмірність, тобто бути як і 1D-вектором, так і 2D- або 3D-зображенням.

Також задано A — кінцеву множину ідентифікаторів вихідних класів пікселів:

$$A = [A_1, A_2, \dots, A_i, \dots, A_d], \quad (1.10)$$

де a_i — i -й ідентифікатор класу,

d — кількість класів.

Існує невідома залежність-відображення y^*_{segm} :

$$y^*_{\text{segm}}: Sx \rightarrow Sx', \quad (1.11)$$

де Sx — множина, що відповідає (1.8),

Sx' — множина вихідних сигналів, кожен елемент якої x_i' є об'єктом з визначеного приналежністю кожного свого компоненту до одного з класів множини A (1.10). Тобто

$$\bar{X}' = [a_1, a_2, \dots, a_i, a_{\text{LenSegm}}], \quad (1.12).$$

a_i — значення i -ого елемента,

де LenSegm — розмірність вектору \bar{X} та \bar{X}' .

При постановці задачі відображення y^*_{segm} відоме тільки на об'єктах навчальної вибірки Sx^m :

$$Sx^m = (\bar{X}_1, \bar{X}'_1), (\bar{X}_2, \bar{X}'_2), \dots, (\bar{X}_i, \bar{X}'_i), \dots, (\bar{X}_{N\text{Train}}, \bar{X}'_{N\text{Train}}), \quad (1.13)$$

де X_i — i -ий вхідний сигнал навчальної вибірки,

X'_i — відображення i -ого вхідного сигналу навчальної вибірки, в якому кожен піксель позначено відповідним ідентифікатором з множини A (1.10),

$N\text{Train}$ — кількість прикладів навчальної вибірки.

Потрібно побудувати алгоритм α_{segm} , який дозволить провести семантичну сегментацію довільного об'єкту, тобто класифікувати кожний компонент сигналу X .

$$\alpha_{\text{segm}}: Sx \rightarrow Sx', \quad (1.11)$$

де Sx — множина, що відповідає (1.8),

Sx' — множина, кожен елемент якої є попиксельною класифікацією вхідного сигналу з множини Sx .

Розв'язання задачі сегментації схематично зображено на рисунку 1.2. Входом системи сегментації зазвичай є 2D- або 3D-зображення, виходом — сигнал такої самої розмірності, на якому виокремлені відповідні регіони.

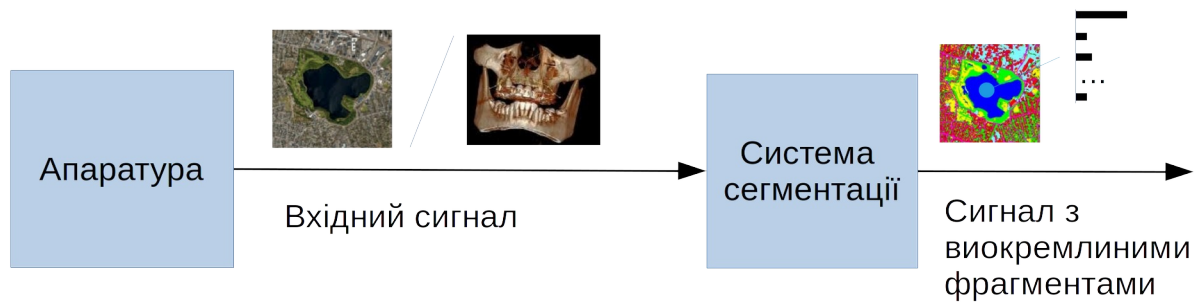


Рисунок 1.2. Схема розв'язання задачі сегментації за діагностичним сигналом

Приклади задач діагностування, які зводяться до наведеної задачі сегментації, наведено в таблиці 1.2.

Таблиця 1.2. Приклади задач сегментації

| Задача | Вхідний сигнал | Посилання |
|--|-------------------------|-----------|
| Сегментація сольових відкладень в надрах землі | Сейсмічний знімок зрізу | [7], 2018 |

Продовження таблиці 1.2

| | | |
|--|---|------------------------|
| Виокремлення автомобілів з видаленням фону | Фотографія | [8], 2017 |
| Сегментація будівель, водойм та транспортних засобів | Супутниковий знімок | [9], 2018 |
| Сегментація космічних об'єктів | Знімок з телескопу | [10], 2019 |
| Випадок 3D сигналу | | |
| Сегментація паталогій мозгу | 3D МРТ(магнітно-резонансна томографія)-знімок | [11], 2018; [12], 2018 |
| Сегментація органів людини | 3D КТ-знімок (комп'ютерна томографія) | [13], 2018 |
| Сегментація патологій легень | 3D МРТ/КТ знімок легень | [14], 2018 |

Основними підходами до розв'язання задач класифікації є:

- SVM [15],
- метод Віола-Джонса [16],
- Баєсівський класифікатор [17],
- алгоритм AdaBoost [18],
- нейронні мережі.

Основними підходами до розв'язання задач сегментації є [19]:

- ТурбоПікселі,
- методи рівня,
- ефективний метод на графах,
- метод К-середніх,

- зсув середнього і розвиток,
- метод водорозділу (Watershed),
- методи рівня,
- нейромережеві методи.

Завдяки поширенню та стрімкому розвитку нейронних мереж такий підхід є найбільш сучасним та точним. Для цього використовують спеціальні різновиди нейронних мереж — згорткові нейронні мережі. Однак і досі в цьому підході основною невирішеною проблемою є досягнення необхідної точності в загальному випадку.

1.2. Згорткові нейронні мережі

1.2.1. Концепція згорткових нейронних мереж

Згорткову нейронну мережу CNN (convolutional neural network) для розпізнавання зображень запропонували Y. Lecun, B. Boser в 1989 році в мережі LeNet [20]. Вона заснована на базових принципах зорової кори головного мозку. Головна її відмінність від попередніх розробок полягає в тому, що мережа не є повнозв'язною. Базовим блоком цієї мережі є шар згортки, який концептуально відображує одношаровий перцептрон з розмірністю рівною розмірності згортки.

В свою чергу, перцептрон вперше був запропонований в 1957 році [21]. Однак справжнього поширення в прикладних системах набув багатошаровий перцептрон MLP (multilayer perceptron), в якому реалізовано алгоритм зворотнього поширення помилки (backpropagation) [22]. В подальшому цей алгоритм був суттєво дороблений, наприклад, в роботі [23]. Проте MLP був неефективним для розпізнавання зображень, адже не міг адаптуватися до зсувів зображення та масштабування. Перцептрон являє собою повноз'єднанні шари (fully connected) нейронів.

В останніх шарах LeNet реалізовано повноз'єднанні шари MLP для класифікації. Вихід останнього згорткового шару 'витагується' в вектор, який безпосередньо під'єднаний до останніх повноз'єднаних шарів.

Схему обробки зображення на мережі LeNet зображено на рисунку 1.3.

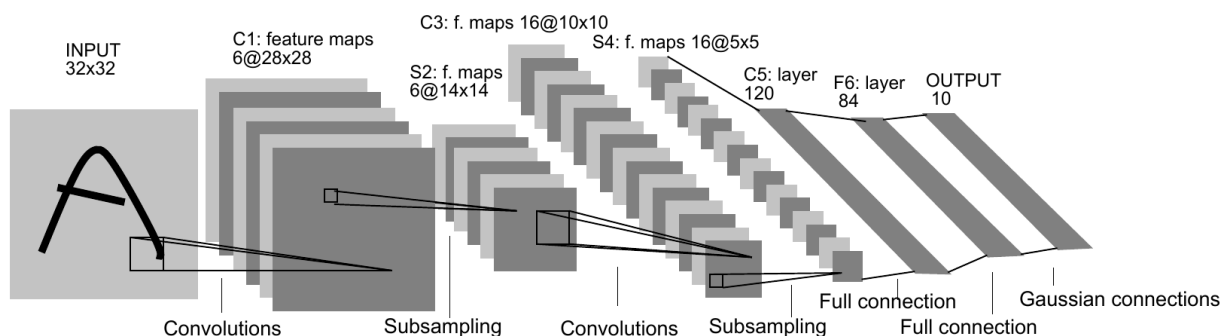


Рисунок 1.3. Мережа LeNet [20]

Завдяки тому, що в мережі CNN нейрони реагують на окремі області зображення, така мережа стає інваріантною до зсувів зображення.

1.2.2. Проблеми згорткових нейронних мереж та засоби їх вирішення

Свого практичного застосування мережі CNN набули в 2010-х роках. До цього основними проблемами розвитку таких нейронних мереж були:

- нестача обчислювальних потужностей;
- неможливість навчання глибоких моделей, що зумовлене затуханням градієнтів;
- перенавчання, що виникає у разі недостатньої складності архітектури нейронної мережі.

Нестача обчислювальних потужностей часткового розв'язана завдяки еволюції апаратного забезпечення, насамперед завдяки розвитку відеопроцесорів, які здатні обробляти дані в багатьох потоках.

Міжнародні змагання та конкурси на основі машинного навчання дозволили вирішити деякі проблеми нейронних мереж і, водночас, суттєво підвищити точність їх роботи.

В 2012 році А. Krizhevsky, І. Sutskever, G. E. Hinton представили мережу AlexNet [24], яка виграла конкурс ImageNet [25] з великим відривом від інших суперників, концепція роботи яких не базувалася на нейромережевих методах. Ця мережа стала першою нейронною мережею, яка виграла цей конкурс.

Мережа AlexNet була реалізована на фільтрах високих розмірностей, які мали велику кількість вільних параметрів. Також велика кількість параметрів була в останніх повно-з'єднаних шарах. Архітектуру мережі AlexNet зображено на рисунку 1.4. На рисунку відображено основні шари згорткових нейронних мереж:

- згортки,
- згортки з кроком (stride),
- шари максимальної агрегації (maxPooling)
- повно-з'єднані шари (dense),
- шари активації, наприклад ReLU.

Їх алгоритми та призначення визначені в другому розділі дисетраційної роботи.

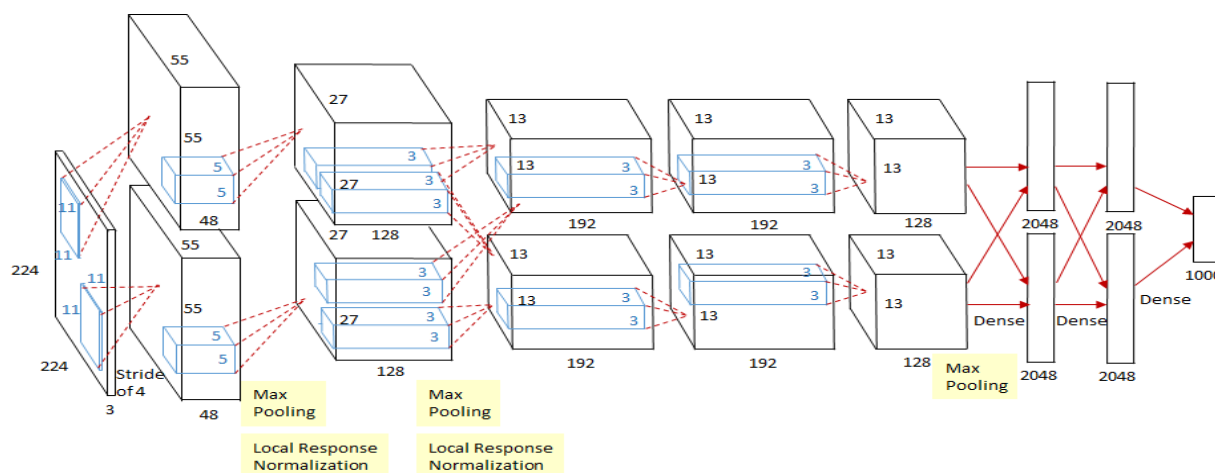


Рисунок 1.4. Мережа AlexNet [26]

На рисунку 1.4 цифрами позначено розмірності згорток та розмірності вхідного сигналу відповідного шару.

Після перемоги AlexNet нейронні мережі почали стрімко розвиватися. В 2014 році конкурс виграла мережа VGG [27] (К. Simonyan, А. Zisserman), яка стала вдосконаленням AlexNet. В цій мережі фільтри високих розмірностей замінили на серію фільтрів 3x3. Завдяки цьому вдалося зменшити кількість параметрів, а отже і збільшити швидкість навчання. При цьому рецептивне поле (receptive field) нейрона залишилося таким самим (рисунок 1.5).

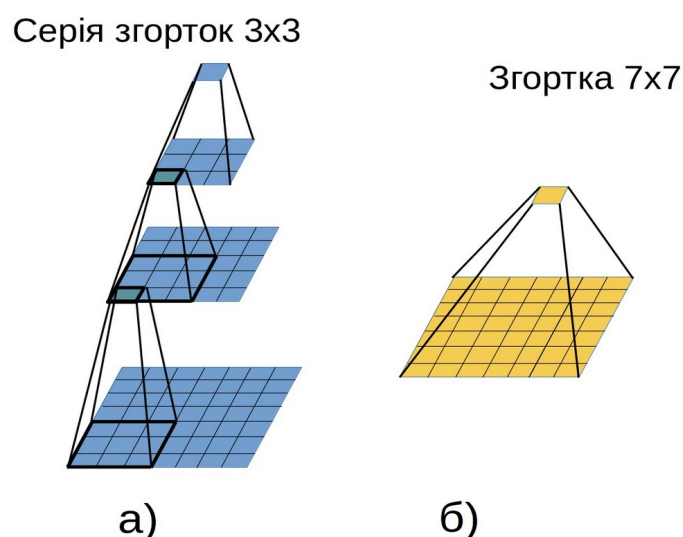


Рисунок 1.5. Рецептивне поле: а) трьох згорток 3x3; б) однієї згортки 7x7

Завдяки такій заміні вдалося зменшити кількість параметрів мережі. Для прикладу, згортка 7x7 потребує $7*7 = 49$ параметрів для одного вхідного каналу, в той час як серія 3-ох послідовних згорток 3x3, для яких рецептивне поле буде такої самої розмірності — $(3*3)*3 = 27$ параметрів. Також серія 3-ох згорток має 3 функції нелінійності, що робить модель більш розрізнявальною (discriminative).

Також в мережі VGG була застосована архітектура, яка стала традиційною: мережа має 5 блоків (рисунок 1.6), після кожного використовується шар пулінгу (або згортка з більшим кроком). Кожний наступний блок обробляє вхідний сигнал меншої розмірності, але збільшує кількість каналів. В подальшому будемо називати ці блоки, як Conv₁, Conv₂, Conv₃, Conv₄, Conv₅.

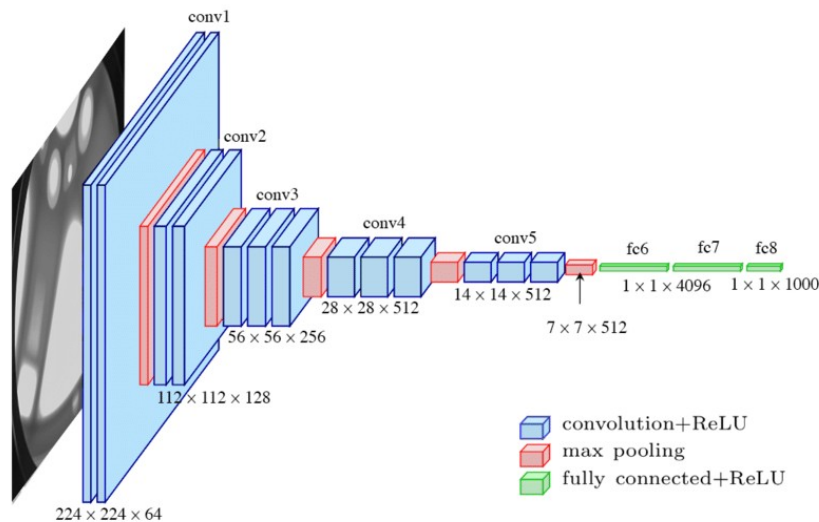


Рисунок 1.6. Архітектура мережі VGG [28]

Також в мережі VGG реалізована попередня ініціалізація вагів на основі роботи [29]. Згідно цієї роботи, вагам нейронної мережі призначаються не довільні ваги, а вибираються ваги з діапазону рівномірного розподілення, який залежить від вхідної та вихідної кількості нейронів. Така ініціалізація дозволяє призначити ваги так, щоб в процесі роботи не спостерігалось явища вибуху або затухання градієнтів через те, що нейрон отримує дуже велике вхідне значення або дуже мале.

Мережі VGG були представлені в декількох модифікаціях (VGG-11, VGG-13, VGG-16, VGG-19) в залежності від кількості шарів згорток мережі. Мережі з більшою кількістю фільтрів показували кращу точність на наборі даних (датасеті) ImageNet. Проте при подальшому нарощуванні шарів,

починаючи з деякої межі спостерігалось зворотнє явище: точність зменшувалася, а помилка зростала (рисунок 1.7).

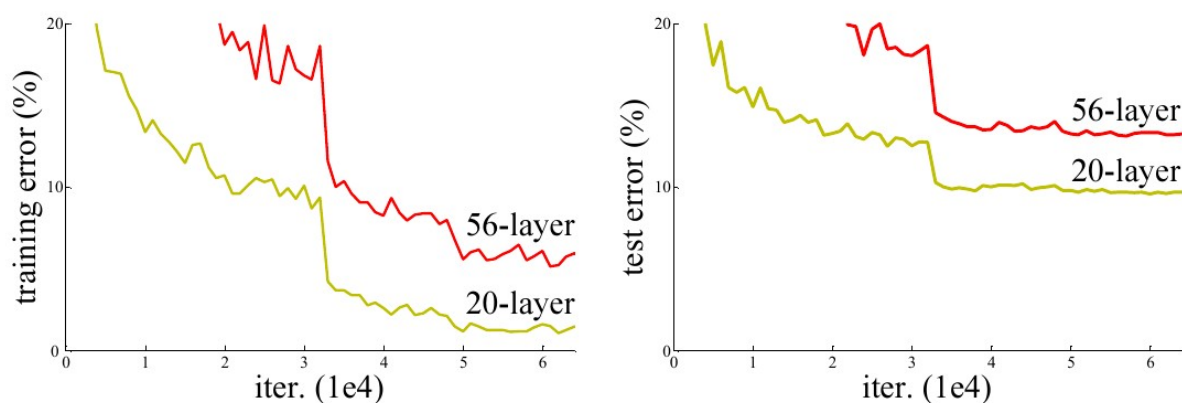


Рисунок 1.7. Помилка на навчальному та тестовому наборі для мереж різної глибини [30]

Це явище пов'язане з затуханням градієнтів (*vanishing gradients*). В процесі навчання глибоких мереж останні шари отримують велике значення градієнту, в той час як в перших шарах градієнт має мале значення амплітуди, отже ближчі до входу шари “майже не тренуються”.

Проблема **затухання градієнтів**, а отже і можливість навчання глибоких мереж, була вирішена в 2015 році в мережі ResNet [30] (К. He, X. Zhang, S. Ren, J. Sun). Це здійснено за допомогою додавання вхідного сигналу блоку до вихідного сигналу (*residual-connection*). Базовий блок представляє 2 послідовні згортки 3x3 (рисунок 1.8).

Також в роботі [30] запропоновано декілька модифікацій мереж, в залежності від кількості шарів: ResNet-{18, 34, 50, 101, 152}. Також автори змогли натренували мережу, яка складалася більше чим з 1000 шарів. Для мереж ResNet-50 і мереж з більшою кількістю шарів, базовий блок було змінено для економії обчислювальних ресурсів. На вході цього блоку було використано згортку з фільтром 1x1, яка зменшувала кількість вхідних каналів. На наступному етапі використано згортку 3x3, після неї — знову згортку 1x1, яка збільшує кількість каналів (рисунок 1.9). Такий метод

дозволив зменшити кількість параметрів мережі. Наприклад, мережа ResNet-34 має 3.6×10^9 параметрів, в той час як мережа ResNet-50 має 3.8×10^9 параметрів.

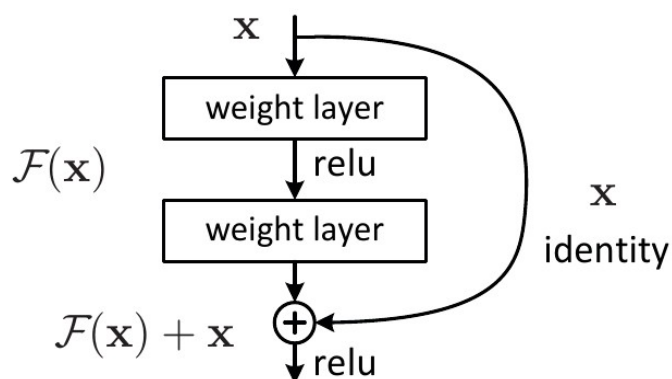


Рисунок 1.8. Базовий блок мережі ResNet [30]

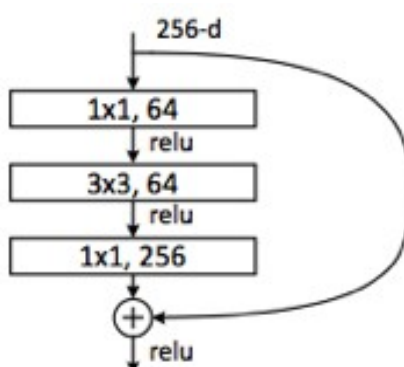


Рисунок 1.9. Блок Conv3 мережі ResNet-50, ResNet-101, ResNet-152 [30]

В мережі ResNet перший шар має згортку розмірності 7×7 , в якій використано зсув (stride) 2×2 , другий — шар активації, третій — шар max-pooling. Таким чином, розмірність вхідного сигналу знижується в 4 рази, що зменшує використання обчислювальних ресурсів. В мережі використано метод глобального пулінгу (global average pooling, GAP), через що останні вихідні шари не додають багато параметрів в мережу.

В роботі [31] представлені мережі ResNext, які були розвитком мереж ResNet. Блок мережі ResNext являє собою N незалежних гілок, кожна з яких є базовим блоком ResNet з меншою кількістю фільтрів згорток (рисунок 1.10.а) з об'єднанням (конкатенацією) результату. Близьким аналогом є згортка 3×3 , яка обробляє не весь вхідний сигнал, а тільки певну кількість вхідних каналів (групова згортка) (рисунок 1.10.б). Такий підхід зменшує кількість параметрів мережі, а також збільшує точність. Автори апробували ці підходи і довели, що вони мають однакову точність.

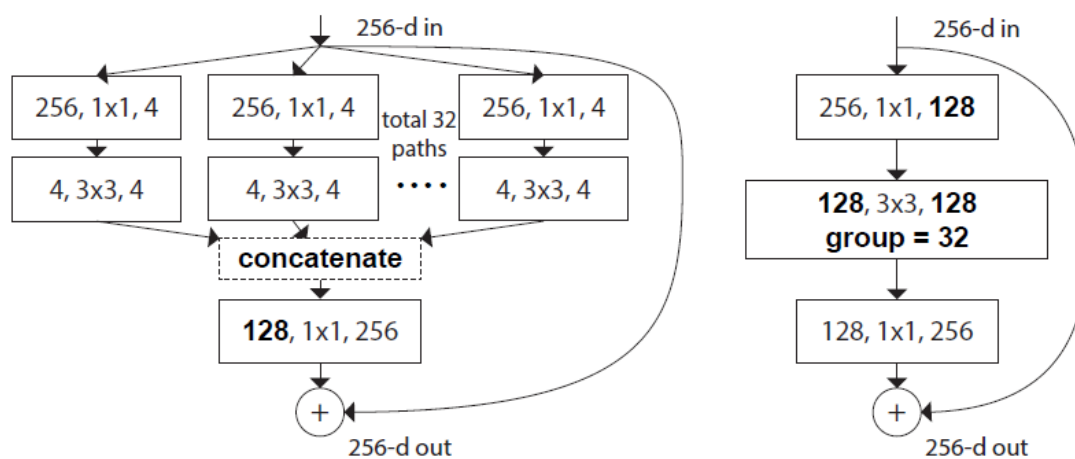


Рисунок 1.10. а) Блок мережі ResNext; б) Його еквівалент з використанням групової згортки [31]

В роботі [32] С. Szegedy, V. Vanhoucke, S. Ioffe та J. Shlens представили мережу Inception-v3. В цій мережі запропоновано використання Inception-блок, в якому використовують паралельно блоки згорток різної розмірності, після чого об'єднують результат (рисунок 1.11).

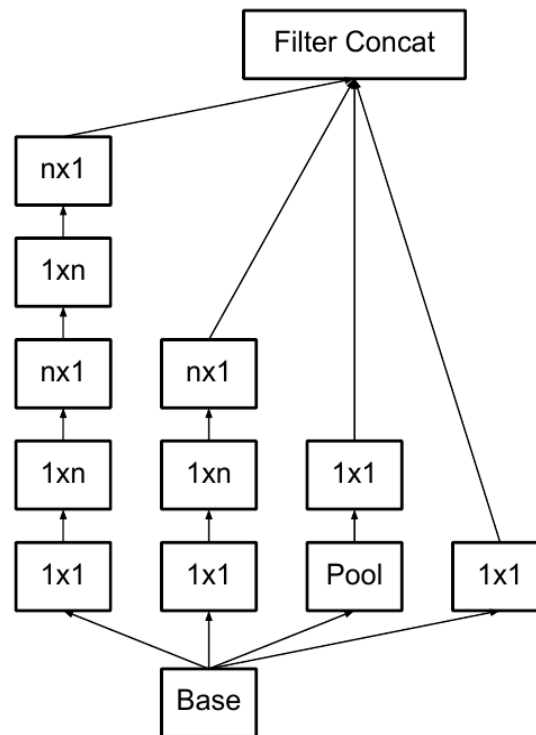


Рисунок 1.11. Блоки мережі Inception-v3 [32]

Також розвинено метод, який запропоновано в мережі VGG: замінено фільтри розмірності $n \times n$ на серію фільтрів $1 \times n$ та $n \times 1$. В цій мережі не використано методи, які були запропоновані в мережі ResNet. Боротьбу з затуханням градієнтів було здійснено іншим методом. Крім останнього повноз'єданого шару наприкінці обробки сигналу, використано ще один такий самий шар, який приєднано до одного з попередніх шарів згортки (рисунок 1.12).

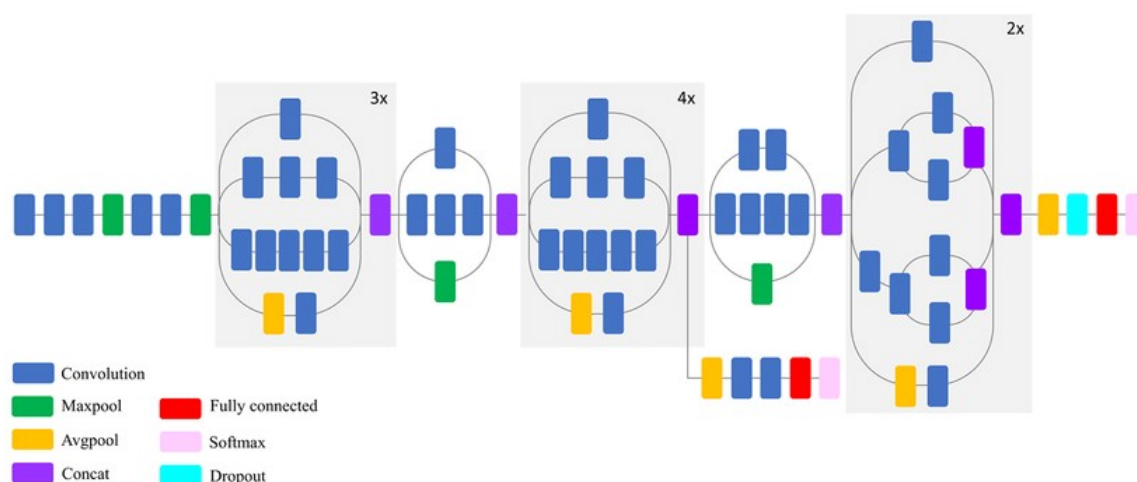


Рисунок 1.12. Архітектура мережі Inception-v3 [32]

Додатковий шар, використовувався тільки при навчанні. В мережі Inception-V4 [33] використали механізми residual-connection, тому необхідність в додаткових класифікаторах відпала.

Механізм residual-connection зарекомендував себе як найефективніший засіб боротьби з явищем затухання градієнту.

Для наступної проблеми перенавчання базовими техніками є:

- аугментація вхідних даних,
- виключення зв'язків DropOut [34],
- підвищення складності архітектури нейронної мережі.

Розв'язанню цієї проблеми також сприяє поява нових та доповнення існуючих наборів даних для навчання. Цьому, в свою чергу, сприяє розвиток мобільних пристроїв, який дозволив суттєво збільшити об'єми даних, насамперед зображень. Таким чином, постійно доповнюються доступні для навчання набори даних, а також з'являються нові. Розвиток обчислювальної техніки дозволив суттєво збільшити об'єми зберігання та отримання даних. Водночас, вдосконалюються спеціальні методи, такі як аугментація [35, 36] які штучно збільшують об'єми даних.

Механізм DropOut дозволяє рендомно (випадково) виключати нейрони з мережі, у зв'язку з цим нейрони не пристосовуються до 'заучування' вхідних даних.

Всі вище представлені засоби сприяють розв'язанню проблеми перенавчання.

1.2.3. Проблема підвищення точності згорткових нейронних мереж

Загальною проблемою при розв'язанні прикладних задач є досягнення необхідної точності класифікації та сегментації. Для задач діагностування це є надкритичним. Всі техніки, які застосовувалися в вище зазначених нейронних мережах, окрім розв'язання безпосередньо проблем перенавчання та затухання градієнтів, водночас так чи інакше були спрямовані на підвищення точності класифікації. Окрім цього, базовими способами безпосередньо **підвищення точності** є:

- механізми уваги,
- пакетна нормалізація,
- використання кращих функцій втрат та/або оптимізаторів,
- використання ансамблів моделей.

Використання механізмів уваги [37, 38] в традиційних архітектурах дозволило суттєво підвищити точність роботи мережі. Механізми уваги дозволили мережі більш точно 'звернути увагу' на ділянки вхідних сигналів, що відповідають за регіон зображення, який має найбільший інтерес для нейронної мережі.

Пакетна нормалізація (batch-normalization) [39] дозволила підвищити швидкість тренування та точність роботи нейронних мереж. Суть методу полягає в тому, що шари нейронних мереж будуть отримувати дані, які оброблені та мають нульове математичне очікування та одиничну дисперсію.

Однак розв'язання кожної прикладної задачі потребує досліджень з використання цих методів та їх комбінації, оскільки краще за критерієм

точності архітектурне рішення для однієї задачі, часто не надає достатньо точних результатів для іншої. З іншої сторони, в практичних задачах часто використовують архітектури, які вже вважаються стандартними, що може зменшувати точність. Таким чином, необхідно вдосконалювати як існуючі методи, так і розробляти нові, які, в подальшому, можна використовувати при пошуку оптимальних рішень для розв'язання поточної прикладної задачі.

1.3. Архітектури нейронних мереж призначених для семантичної сегментації

Паралельно з розвитком нейронних мереж для класифікації зображень, інтенсивно розвивався напрям сегментації зображень. В 2015 році V. Badrinarayanan, A. Kendall, R. Cipolla представили мережу SegNet [40]. Мережа складалася з двох частин — енкодеру та декодеру. Часто енкодер (ліву частину мережі) називають backbone. Саму вузьку частину мережі називають bottleneck (горло пляшки, вузьке місце). В мережі SegNet енкодер використано з мережі VGG без використання останніх повно-з'єднаних шарів. Декодер містить шари підвищення дискретизації та згорток. В останніх шарах використано згортку розмірністю 1x1.

Мережа U-Net [41] (O. Ronneberger, P. Fischer, T. Brox) є однією з найвідоміших мереж, призначених саме для сегментації. Мережа також складається з двох частин: енкодеру та декодеру, проте існують прямі з'єднання енкодеру та декодеру на різних шарах (skip connections), які копіюють вихід енкодеру у відповідний блок декодеру (рисунок 1.13).

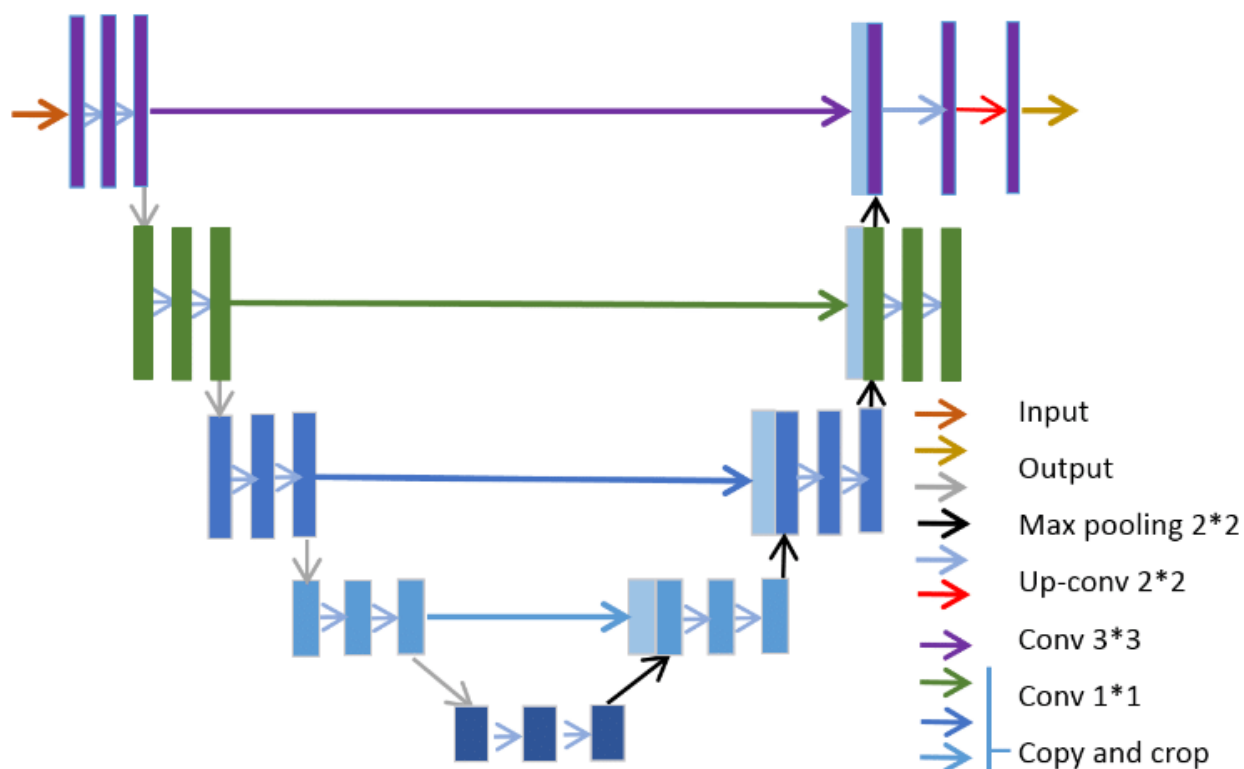


Рисунок 1.13. Архітектура мережі U-Net [41]

Мережа **FPN** (Feature Pyramid Network) [42] була запропонована T.Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, та S. Belongie в 2016 році. Мережа схожа за структурою на U-Net, проте кожний блок енкодера не копіюється, а передається через шар згортки (рисунок 1.14).

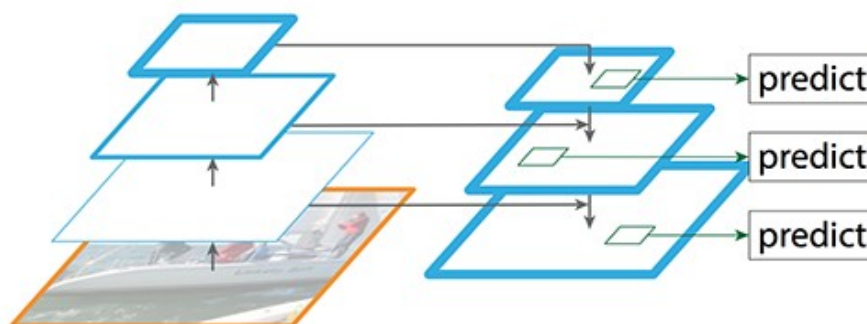


Рисунок 1.14. Архітектура мережі FPN [42]

Мережа U-Net робить передбачення наприкінці декодера, в той час як FPN робить передбачення на кожному блоці декодера. Після чого

здійснюється масштабування всіх передбачень до вихідного розміру з фінальною згорткою.

Концепція мережі PSPNet (Pyramid Scene Parsing Network) [43] полягає в використанні пулінгу з різним кроком страйду та ядер згортки з подальшим підвищенням розмірності зображення за допомогою білінійної інтерполяції (рисунок 1.15).

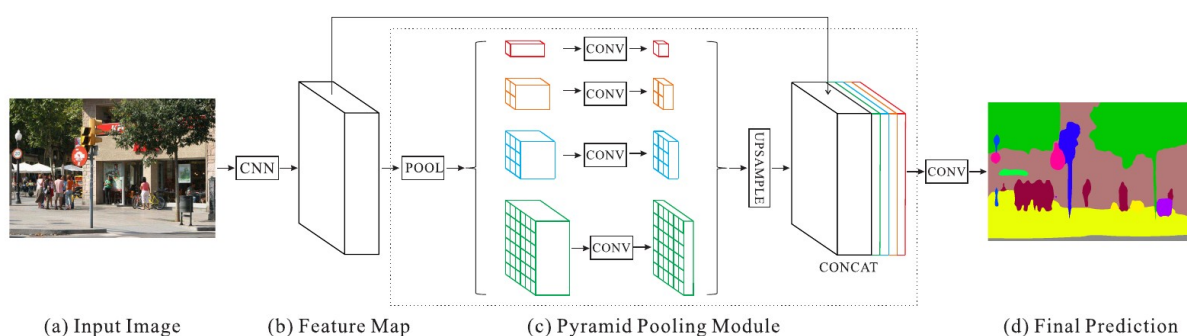


Рисунок 1.15. Концепція мережі PSPNet [43]

Часто нейронні мережі застосовуються в задачах розпізнавання режимів роботи обладнання, в яких можуть бути перехідні процеси, тому доцільно використовувати не тільки згорткові нейронні мережі, а також спеціальні засоби, такі як нейронні мережі Кохонена — карти самоорганізації SOM (self-organizing map).

1.3 Нейронні мережі Кохонена

D. J. Willshaw, C. Von Der Malsburg [44] описали тип нейронної мережі, яка заснована на біологічних признаках, а саме відображення сітчатки на зорову кору мозку. Мережа містить дві зв'язані двомірні решітки нейронів, одна проектується на іншу. Перша представляє вхідні нейрони, друга — вихідні. Автори взяли за основу біологічну властивість того, що сусідні нейрони з однієї частини кори головного мозку проектуються на сусідні нейрони в іншій частині. Дистанція між нейронами виражається в

термінах їх подібності по відношенню до деяких фізичних властивостей цих нейронів.

На основі цієї роботи Т. Kohonen запропонував різновид нейронних мереж — SOM (self-organizing map) [45]. Такі мережі за своєю спеціалізацією пристосовані для візуалізації даних і кластерного аналізу (якщо відома кількість кластерів). Перевагами SOM є стійкість до зашумлених даних, швидке навчання, невелика кількість прикладів навчальної вибірки. Великим недоліком цього типу нейронних мереж є надчутливість до початкового розподілу значень вагів.

В роботах [46-52] наведено базові способи вдосконалення SOM.

Кожен з дослідників пропонував вдосконалення за своїм набором критеріїв, тому необхідно систематизувати існуючі способи оптимізації, щоб в подальшому можна було обирати такі з них, що найкраще відповідають розв'язанню поточної задачі.

Основними критеріями оптимізації карт самоорганізації є:

- визначення чітких границь (чіткого околу центру кластеру);
- швидкість навчання.

Оптимізація за критерієм визначення чітких границь. Основним з підходів досягнення чітко визначених кластерів є створення SOM з динамічною структурою. В цьому випадку розмір “поверхні” SOM може змінюватися в процесі навчання забезпечуючи, наприклад, додатковий простір для нового кластеру. Це відбувається додаванням нових нод. Нодою є нейрон або група нейронів.

В [46] запропоновано дві моделі SOM. Перша відповідає базовій концепції Кохонена. На ній виконується неконтрольоване навчання і модель може бути використана для візуалізації даних, кластеризації і векторного квантування. Задачі розв'язуються за допомогою процесу контрольованого зростання розмірності SOM. Цей процес також включає в себе видалення блоків. Друга модель є результатом поєднання попередньої моделі з використанням радіальних базисних функцій (RBF).

Традиційно критеріями зупинки навчання SOM є відсутність помітних змін в вагах SOM на поточному кроці або досягнення заданої максимальної кількості кроків. В роботі [47] пропонується метод для вимірювання ступеня збереження топології. На основі генетичних алгоритмів в два етапи формується топологічна карта функції зміни поверхні SOM. Потім використовується спеціальний метод для інтерпретації SOM, реалізований генетичними алгоритмами на основі методу оцінювання кількості і місць розташування кластерів. Мережа краще формує кластери, які не є топологічно впорядкованими.

Автори [48] пропонують модифікацію карти самоорганізації — SOINN (Self-Organizing Incremental Neural Network). Мережа містить 2 шари, які навчаються один за одним. Перший шар використовується для визначення топологічної структури кластерів, другий — для визначення кількості кластерів і виявлення вузлів для них. В процесі навчання мережа отримує вхідний вектор і на першому шарі, створюється нейрон, який визначає клас вхідного вектора. Якщо вхідний вектор схожий на існуючий клас, тоді відповідні нейрони укріплюють свій зв'язок. У випадку, якщо вхідний вектор несхожий на існуючий клас, створений нейрон визначає новий клас. Перший шар є вхідним для другого. Схожі нейрони першого шару, які об'єднанні зв'язком, визначають один і той самий клас. Після закінчення навчання Ну шару, починає навчатися другий шар по подібному алгоритму. Карта “росте” в процесі навчання і при цьому менш чутлива до шуму вхідного сигналу.

В [49] запропоновано карту самоорганізації, яку назвали ESOINN (Enhanced Self-Organizing Incremental Neural Network). В ній усунуто такі недоліки SOINN, як:

- 1) невизначеність моменту зупинки навчання першого шару і початку навчання другого;
- 2) проблема злиття кластерів, які близько розташовані.

Мережа ESOINN одношарова, тому містить меншу кількість параметрів. Також, на відміну від SOINN, де нейрони-переможці завжди

з'єднувалися зв'язками, ESOINN вводить умови створення кластерів. Ці умови враховують взаємне розміщення класів, до яких належать нейрони-переможці. Додавання цих правил дозволяють алгоритму успішно розділяти близькі та частково перекриваючи один одного класи.

Оптимізація за критеріями швидкості навчання. В роботі [50] запропонована модифікація мережі Кохонена GSOM (Growing Self-Organizing Map). Модель GSOM вирішує проблему вибору оптимальної розмірності карти самоорганізації. При старті є набір векторів до стартових нод. При поданні нових вхідних векторів, якщо ваги відрізняються, відбувається додавання нових нод. GSOM застосовується для нелінійного зменшення розмірності, апроксимації, кластеризації. Мережа GSOM, як правило, дає краще представлення про геометрію даних, ніж SOM.

В [51] запропоновано модифікацію алгоритму навчання на великих наборах несхожих даних задля збільшення швидкості навчання. При навчанні обирається приклад, який обчислюється SOM. На карті визначається кластер-переможець, а в навчальній вибірці даних шукається приклад, який попадає в цей самий кластер. Цей цикл повторюється для всіх прикладів навчальної вибірки.

В [52] запропоновано два методи для прискорення навчання SOM. Перший називається “усунення часткової відстані” (“partial distance elimination”). Інший метод заснований на властивостях збереження топології карти Кохонена і статистичної кореляції вхідних векторів. Комп'ютерне моделювання показує, що для реальних даних, карта займає вдвічі менше обчислювальних ресурсів.

В даній роботі для апробації підвищення точності класифікації обрані спеціальні мережі SOM, GSOM, ESOINN (розділ 3).

Таким чином, в розділі було формалізовано задачі розпізнавання за діагностичним сигналом, обґрунтовано використання двох нейромережових підходів: згорткових нейронних мереж та карт самоорганізацій Кохонена,

визначено проблеми, які властиві цим мережам та напрямки подальших досліджень.

Висновки до розділу 1.

1. Виокремлено задачі діагностування, що потребують збільшення точності розв'язання на основі існуючих нейромережових методів. Наведено формалізацію постановки задач класифікації та семантичної сегментації.

2. Проведено аналіз розв'язання базових проблем навчання та функціонування згорткових нейронних мереж. Обрано для подальшого застосування механізми уваги, нормалізацію, виключення зв'язків.

3. Проведено аналіз архітектур та алгоритмічного забезпечення згорткових нейронних мереж. Обґрунтовано для подальшого вдосконалення архітектури ResNet, ResNext, U-Net, FPN, PSPNet.

4. Проведено аналіз модифікацій та вдосконалень нейронних мереж Кохонена. Виокремлено 2 критерії їх оптимізації. Для подальшого використання обрано модифіковані мережі SOM: GSOM, ESOINN.

Результати досліджень було апробовано на 1 міжнародній конференції з опублікуванням тез [53].

РОЗДІЛ 2

СЕМАНТИЧНА СЕГМЕНТАЦІЯ ТА КЛАСИФІКАЦІЯ НА МЕРЕЖАХ ГЛИБОКОГО НАВЧАННЯ

В розділі представлено розв'язання двох задач щодо підвищення точності семантичної сегментації та збільшення точності класифікації сигналів великої розмірності.

Дослідження цього розділу, щодо задачі семантичної сегментації, спрямовані на вдосконалення математичного та алгоритмічного забезпечення мереж глибокого навчання U-Net, FPN, DeepLab v3 та PSPNet для підвищення точності семантичної сегментації. Для цього вирішено такі задачі:

1. Визначити базові блоки згорткових нейронних мереж, їх архітектур та засоби оптимізації.
2. Провести аналіз алгоритмів виявлення контурів на зображенні.
3. Реалізувати оператор Собеля в додатковому блоці нейронної мережі.
4. Розробити модифікацію мереж, призначених для семантичної сегментації, на основі блоку Собеля
5. Модифікувати архітектуру мережі U-Net з додаванням блоку PSP та модифікацією структури блоку декодера.
6. Здійснити постановку тестових задач.
7. Провести обчислювальні експерименти з розв'язання задач семантичної сегментації для порівняльного аналізу результатів, отриманих на базових та модифікованих мережах.

Дослідження цього розділу, щодо задачі класифікації спрямовані на розробку способу модифікації архітектури згорткової нейронної мережі для підвищення точності класифікації сигналу великої розмірності.

Для досягнення поставленої мети необхідно вирішити такі задачі:

- Розробити додатковий блок згорткової нейромережі для збільшення рецептивного поля.
- Модифікувати базові архітектури згорткових нейронних мереж, пристосованих для розв'язання задач класифікації.
- Провести обчислювальні експерименти розв'язання задач класифікації за базовими та модифікованими архітектурами нейронних мереж.

2.1. Згорткові нейронні мережі

2.1.1. Базові блоки згорткових нейронних мереж

Основним блоком згорткової нейронної мережі є **шар згортки**. Згортка — процес додавання кожного елемента вхідного сигналу до сусідніх елементів, використовуючи зважене ядро. Під елементом сигналу мається на увазі значення в конкретний момент часу для часового ряду або значення пікселю у випадку 2D-зображення.

Ядра згортки зазвичай використовують розмірності 1D, 2D або 3D відповідно до розмірності вхідного сигналу. Ядро згортки (фільтр) застосовується до кожного елемента вхідного сигналу, результат записується в центральний елемент, тобто згортка 'зсовується' на певний крок. Оскільки для крайніх елементів немає сусідів, зазвичай використовують доповнення вхідного сигналу, як правило нулями.

В процесі навчання значення фільтрів налаштовуються. Формулу для обчислення вихідного значення після згортки конкретного елемента можна виразити як:

$$x_i^{\text{conv}} = \sum_{j=-k/2}^{k/2} x_{i-j} * W_{|w_j * 2|}, \quad (2.1)$$

де x_i — значення i -го елемента сигналу

x_i^{conv} — вихідне значення i -го елемента сигналу,

k — розмірність ядра згортки,

w_j — значення j -го елемента ядра згортки.

Також згортка може мати додаткову вагу — початкове зміщення (bias), входом якого є константа, яка дорівнює 1. В такому випадку формула (2.1) буде доповнена:

$$x_i^{conv} = \sum_{j=-k/2}^{k/2} x_{i-j} * w_{|w_j * 2|} + w_0 * b \quad (2.2)$$

де w_0 — вага зміщення (необов'язковий параметр),

b — початкове значення зміщення.

Приклад 1D вхідного сигналу та ядра згортки розмірністю 3 зображено на рисунку 2.1.

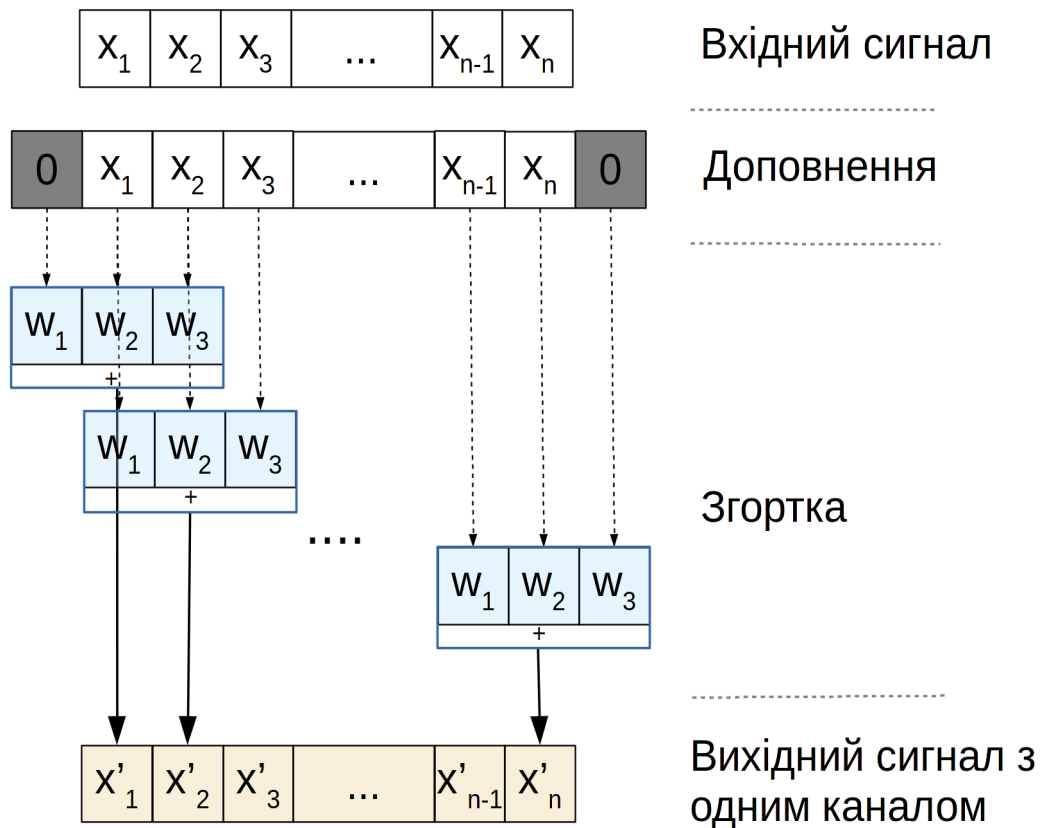


Рисунок 2.1. Приклад 1D згортки

Оскільки в CNN згорткові шари розташовані паралельно, кількість таких шарів рівна кількості ядер згортки, причому ці ядра відмінні між собою. Для кожного такого шару входом є поточний сигнал, виходом — сигнал з одним каналом. Таким чином, серія таких згорток надає серію каналів.

Для зниження розмірності і підвищення рецептивного поля (тобто ділянки вхідного сигналу, яка впливає на конкретний нейрон) наступного шару, в згорткових нейронних мережах використовується **шар агрегації** (пониження розмірності, **pooling**). Зазвичай використовують максимізаційне агрегування (max pooling), яке використовує максимальне значення з кожного з кластерів нейронів. В шарі pooling-у задається параметр кроку (stride), тобто кількість елементів, які будуть агрегуватися. В роботі прийнято стандартний крок, який дорівнює 2. Приклад шару пулінгу з кроком 2 зображено на рисунку 2.2.

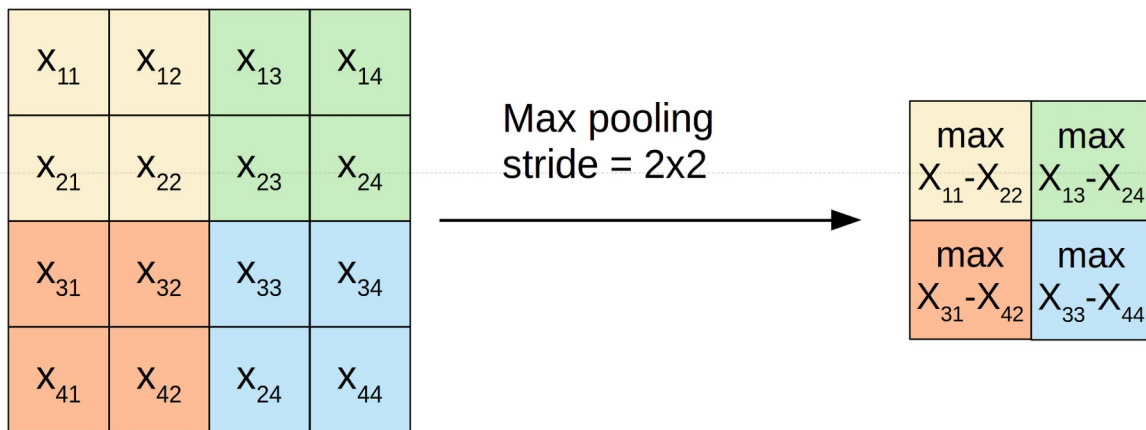


Рисунок 2.2. Приклад Max-Pooling з кроком 2 для 2D-сигналу

В останніх повно-з'єднаних шарах нейронних мереж зосереджено багато параметрів, їх кількість часто більша, за кількість параметрів всіх згорткових шарів. Це заважає швидко навчати мережі, мережі здатні до перенавчання і займають багато пам'яті.

Зниження кількості параметрів в повно-з'єднаних шарах запропоновано в роботі M. Lin, Q. Chen, S. Yan [54] на основі використання шару глобальної агрегації (**global average pooling, GAP**). Суть методу полягала в знаходженні середнього значення по каналу, замість того, щоб 'витягувати' весь вихід в єдиний вектор. Також в роботі використано єдиний повно-з'єднаний шар, замість серії повно-зв'язаних.

Схематично метод GAP зображено на рисунку 2.3. Завдяки тому, що вихід після GAP шару завжди сталий і не залежить від вхідного розміру, такий метод дозволив навчати та використовувати в робочому режимі нейронні мережі на зображеннях довільного розміру, без попереднього масштабування. Оскільки на більших зображеннях нейронна мережа повільно тренується, на практиці часто застосовується **метод послідовного навчання** [55]. В даній роботі було використано саме цей метод за алгоритмом:

1. Встановити коефіцієнт масштабування вхідного сигналу $\alpha < 1$.
2. Провести N епох тренування нейронної мережі з використанням коефіцієнту масштабування.
3. Якщо точності недостатньо, збільшити коефіцієнт масштабування α та повернутися до пункту 2.

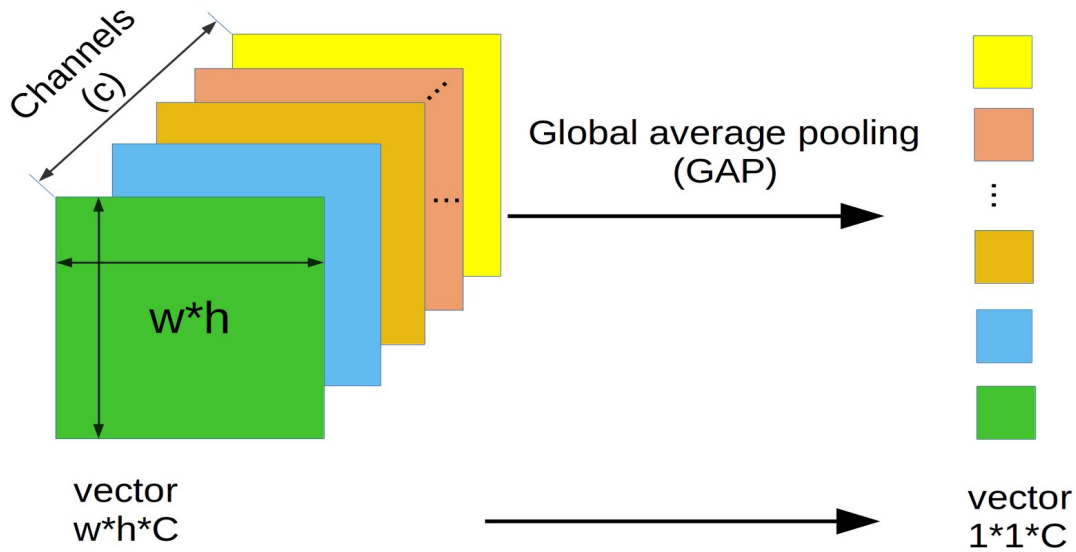


Рисунок 2.3. Приклад агрегування на основі середнього

Якщо згорткові нейронні мережі застосовують лише зазначені шари, то отримана мережа здатна здійснювати лише лінійні перетворення вхідного сигналу. Тому прийнято застосовувати додаткові блоки з **функцій активації**, які здійснюють нелінійне перетворення сигналу від шару до шару.

В сучасних згорткових нейронних мережах зазвичай використовують функцію активації “випрямляч” Rectified Linear Units (**ReLU**) [56] (R. Hahnloser, R. Sarpeshkar, M. Mahowald, R. Douglas, H. Seung) або його модифікації. Використання ReLU замість гіперболічного тангенсу або сигмоїди дозволило зменшити помилки та збільшити швидкість навчання і зменшити явище затухання градієнтів. Функція активації ReLU реалізує пороговий перехід в точці 0 та обчислюється за формулою:

$$f(x_i)_{\text{relu}} = \max(0, x_i), \quad (2.3)$$

де x_i — значення i -го елемента вхідного сигналу,

$f(x_i)_{\text{relu}}$ — значення вхідного сигналу після функції активації ReLU.

Це відбувається тому, що похідна (градієнт) від ReLU дорівнює $\{0, 1\}$, в той час як похідна гіперболічного тангенсу або сигмоїди змінюється в широких межах. Проте, якщо функція виходить на плато (тобто функція

знаходиться в насиченому стані), похідна в таких точках наближається до 0 (рисунок 2.4). Щоб ‘зсунути’ функцію з таких точок, потрібно потратити багато епох навчання, в той час як в ReLU таких точок не існує.

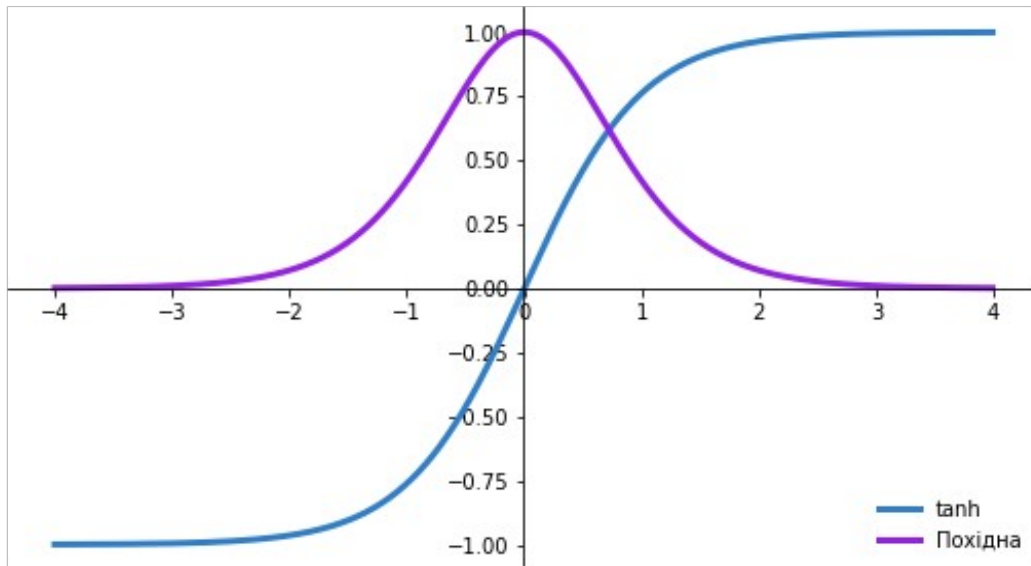


Рисунок 2.4. Графік функції гіперболічного тангенсу та його похідна

Однак використання функції ReLU має недолік: можливе ‘застрягання’ деяких ваг через те, що їх вихідне значення буде меншим 0, тобто після функції ReLU вихідне значення дорівнює 0, а отже її похідна — 0. В даній роботі використано функцію активації **PReLU** (Parametric Rectified Linear Unit) в блоках декодера U-Net, яка була запропонована в [57]. Завдяки цій функції активації усунено зазначений вище недолік ReLU.

Функція PReLU схожа на ReLU, але додає нахил лівої частини графіку ($x < 0$). Коефіцієнт нахилу налаштовується в процесі роботи (рисунок 2.5).

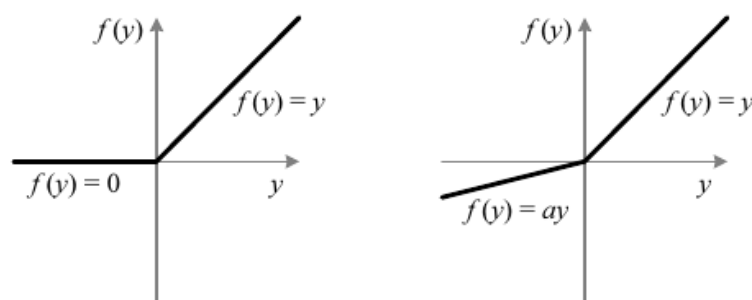


Рисунок 2.5. Функція активація ReLU та PReLU [57]

До базових шарів CNN відносять шари нормалізації різних типів. Найбільш розповсюдженим є **пакетна нормалізація (batch-normalization)**.

Шар нормалізації найчастіше використовується між шарами згортки та активації і обробляє дані так, щоб вони мали нульове математичне очікування та одиничну дисперсію. Тобто нормалізація зводить дані до проміжку $[-1; +1]$.

В запропонованій роботі застосовано нормалізацію, яка була запропонована в роботі [39] S. Ioffe, C. Szegedy, який дозволив пришвидшити навчання та збільшити точність. Оскільки метод працює для серії даних, які подаються на вхід (міні-батчу), тому він отримав назву пакетної нормалізації (batch-normalization).

2.1.2 Механізми оптимізації згорткових нейронних мереж

В роботі [37] J. Hu, L. Shen, S. Albanie, G. Sun та E. Wu запропонували використання **механізмів уваги** для розпізнавання зображень, ідеї якої пішли з розпізнавання тексту. Запропонований SE-блок (Squeeze-and-Excitation block) був інтегрований в інші сучасні мережі.

У блоці вхідний сигнал проходить через шар глобального пулінгу, після чого проходить 2 послідовних повноз'єднаних шари. На виході блоку використовується сигмоїдальна активація, що дозволяє визначати 'вагу' кожного каналу в межах $(0, 1)$. Після цього значення кожного з каналів помножується на вихідні значення другого повноз'єданого шару (рисунок 2.6). В процесі навчання мережа вчиться виокремлювати найбільш значимі канали. Оскільки розмірність сигналу після шару глобального пулінгу — це тільки кількість каналів, такий підхід не потребує великих додаткових ресурсів.

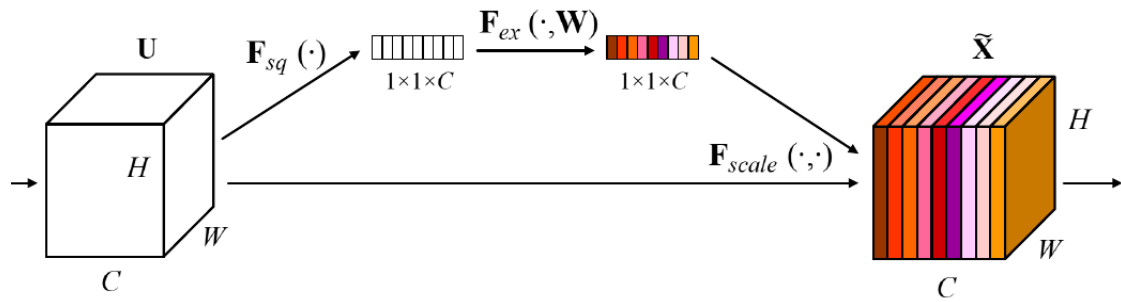


Рисунок 2.6. A Squeeze-and-Excitation block [37]

В роботі використано нейронні мережі, з даним механізмом уваги, а саме: SE_ResNext50.

Подальший розвиток механізмів уваги набув в роботі [38]. Крім того, що використовується механізм уваги по каналах, використано також механізм уваги і по регіонах зображення. Запропонований A. G. Roy, N. Navab, and S. Wachinger (scSE — Concurrent Spatial and Channel Squeeze & Excitation) механізм складається з двох блоків, які обчислюються паралельно, а результат сумується (рисунок 2.7). Верхній блок виконує згортку 1x1 та виконує множення вхідного сигналу на результат згортки, другий взятий з роботи [37]. В результаті мережа вчиться виокремлювати найбільш значимі об'єкти як по регіонах зображення, так і по каналах. В даній роботі гібридний механізм уваги застосовано в мережі U-Net у skip-з'єднаннях.

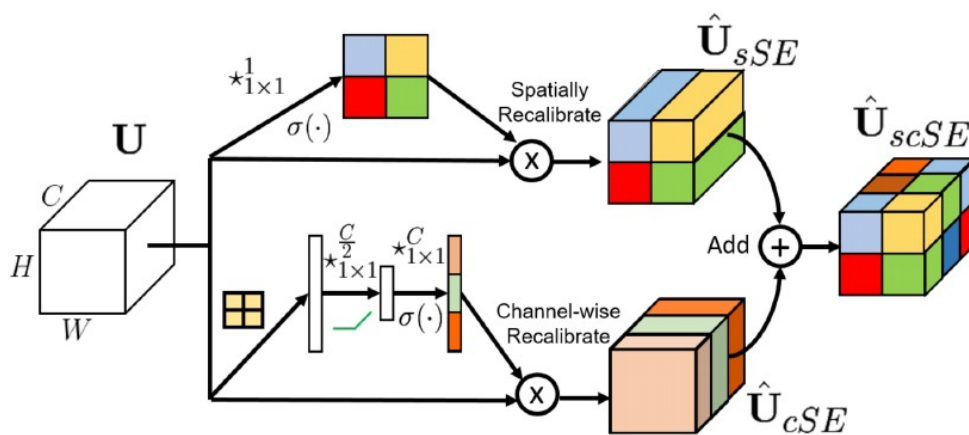


Рисунок 2.7. Блок scSE [38]

Окрім вищезазначених механізмів уваги в даній роботі використано заміну енкодера U-Net на класичну мережу.

Ця ідея запропонована в [58] і полягає в використанні заздалегідь апробованих архітектур. Крім цього, використання заздалегідь навчених на певних наборах даних мереж, зазвичай підвищує точність, оскільки backbone-мережа має властивість виокремлювати певні признаки.

2.2. Семантична сегментація

2.2.1. Підвищення точності семантичної сегментації

Підвищення якості сегментації можна здійснювати на рівні архітектури, завдяки модифікації існуючих блоків та/або створення блоків з новим функціоналом. При навчанні згорткових мереж, згорткові фільтри також виокремлюють границі. Однак, після проходження через багато шарів згортки на нижніх шарах енкодера границі об'єкту стають розмитими. В даній роботі використано припущення, що в прикладних задачах часто контур об'єкту можна знайти за зміною градієнту. Оскільки в задачах сегментації потрібно точно знати границі об'єкту, то додаткове виокремлення контурів має покращити роботу мережі.

Вхідними даними є вхідне зображення розмірності $c*w*h$, де c — кількість каналів (для RGB зображення $c=3$), w , h — ширина та висота. Оператор виділення контурів вертає карти признаков розмірності $w*h$, які подаються на декодер мереж.

Виявлення контурів на зображенні можливо за допомогою операторів Кенні, Робертса, Прюїтта, Собеля. В дослідженні [59] проведено аналіз ефективності застосування цих операторів. Кращими за точністю визнано використання операторів Кенні та Собеля. Однак оператор Собеля має такі

переваги як простота реалізації, яка впливає на швидкість обробки, та легка інтеграція зі згортковими нейронними мережами.

Для згортки вхідного зображення оператор Собеля використовує 2 матриці:

$$S_x = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}; S_y = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}. \quad (2.4)$$

Застосування матриць (2.4) дозволило ефективно виокремлюють контури для тих об'єктів, границі яких наближені до осі X (вісь X спрямована вправо) або осі Y. Додатково можна використовувати матриці згортки виду:

$$S_{d_1} = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{pmatrix}; S_{d_2} = \begin{pmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{pmatrix}. \quad (2.5)$$

Матриці (2.5) дозволяють ефективно виокремлювати контури, наближені до діагоналей.

Таким чином, можна визначити такі операції згортки:

$$\begin{aligned} G_x &= S_x * A, \\ G_y &= S_y * A, \\ G_{D_1} &= S_{d_1} * A, \\ G_{D_2} &= S_{d_2} * A, \end{aligned} \quad (2.6)$$

де G_x , G_y , G_{d1} , G_{d2} — результати згортки по осях та діагоналях,

S_x , S_y , S_{d1} , S_{d2} — матриці Собеля (2.4), (2.5),

A – вхідне зображення, що передається як матриця значень пікселів.

Результуюче зображення отримується об'єднанням результатів:

$$G = \frac{1}{2} \sqrt{G_y^2 + G_x^2 + G_{d_1}^2 + G_{d_2}^2}, \quad (2.7)$$

де G — фінальне зображення;

G_y , G_x , G_{d1} , G_{d2} — результати згортки по осях та діагоналях.

Результати застосування конкатенації всіх складових відображень оператору Собеля з безпосередньо результатом його обчислень (2.7), надає кращі результати ніж використання лише оператору Собеля.

Таким чином, після застосування всіх операцій отримується така сукупність матриць:

$$Out = [G, Gy, Gx, Gd_1, Gd_2]. \quad (2.8)$$

В подальшому реалізацію обчислень *Out* будемо називати **блоком Собеля**.

Як правило, входними зображеннями являються не одноканальні зображення (чорнобілі), а дво- і більше канальні (наприклад RGB-зображення). Оператор Собеля застосовується для кожного каналу незалежно. Наприклад, для RGB-зображення буде отримано 15 каналів, що буде відтворено сукупністю матриць *OutR*, *OutG*, *OutB*.

Модифікація архітектури мереж глибокого навчання

В даній роботі дослідження проводилися на мережах U-Net та FPN, DeepLab v3 та PSPNet. Елементарними шарами цих мереж є згортка, активація, конкатенація, пулінг, додавання, підвищення розмірності (UpSampling).

Для представлення модифікованої архітектури введемо поняття блоку, попередньо зазначивши, що розмірність зображення в загальному випадку визначається як:

$$Shape = W * H * C, \quad (2.9)$$

де *W* — висота зображення,

H — ширина зображення,

C — кількість каналів зображення.

Визначимо **блок** як групу елементарних шарів нейронної мережі, яка обробляє входні дані фіксованої розмірності лише по ширині (*W*) та висоті (*H*).

На рисунку 2.8 представлено блочну архітектуру модифікованої мережі U-Net, де застосовано оператор Собеля.

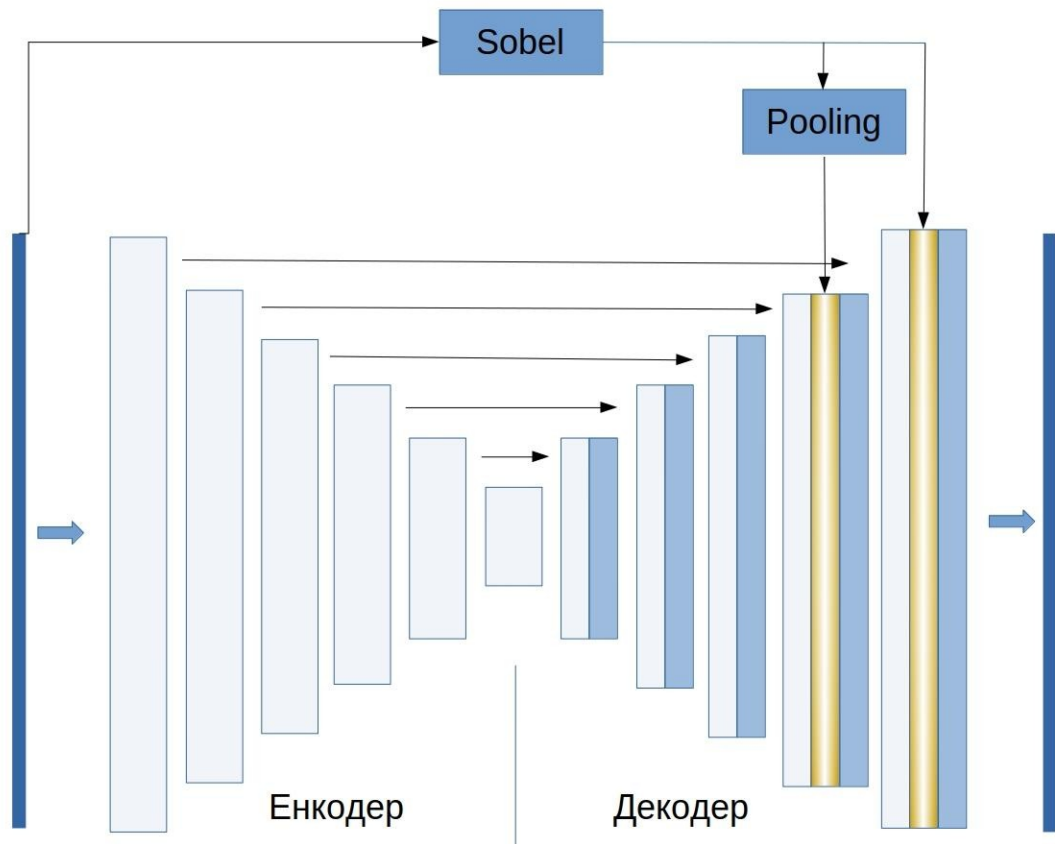


Рисунок 2.8. Модернізація мережі U-Net

На рисунку повтором синього блоку акцентовано копіювання даних з виходу блоку енкодера у відповідний блок декодера. Необроблене вхідне зображення подається на енкодер і в подальшому проходить стандартну обробку. В запропонованій архітектурі модифіковано лише декілька верхніх (останніх) блоків декодера. З рисунку 2.8 видно, що в цих блоках додано примусове визначення контурів (виокремлено жовтим кольором) і їх конкатенація з відповідними блоками енкодера. Дослідження показали, що кращі результати надає використання запропонованого способу на 2-3 останніх блоках.

Крім цього, було запропоновано модифікацію мережі FPN, яка також є мережею глибокого навчання, призначеною для семантичної сегментації, але на відміну від мережі U-Net має іншу структуру декодера.

Модифікацію мережі FPN зображено на рисунку 2.9. Основна відмінність полягає в тому, що сигнал після блоку Собеля проходить через шар згортки 3*3.

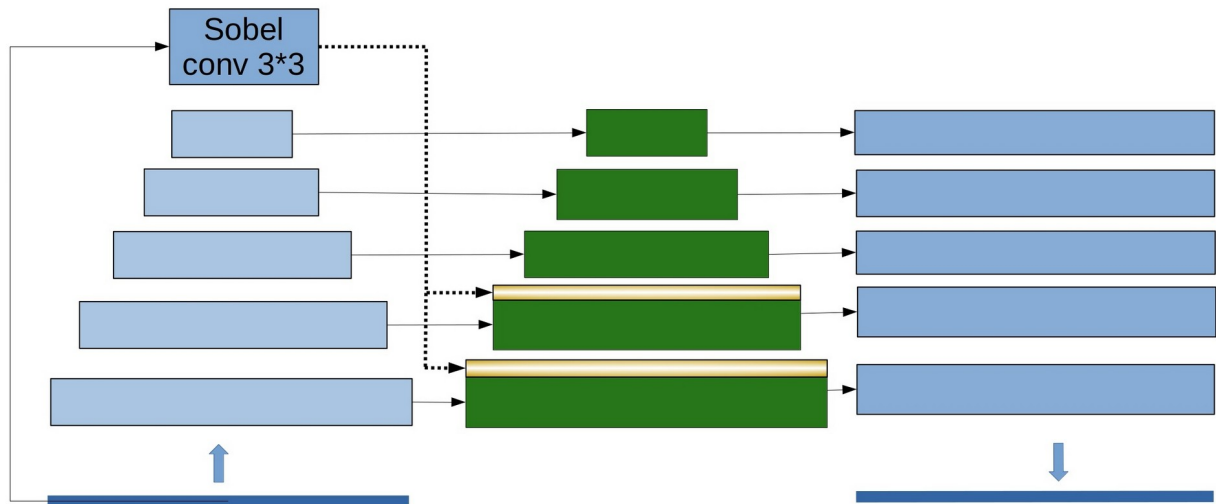


Рисунок 2.9. Модернізація мережі FPN

Оскільки розмірності шарів по висоті та ширині не співпадають, необхідно результат (2.8) понизити до відповідної розмірності. Блок декодера мережі має розмірність:

$$Shape = (W/k; H/k) \quad (2.10)$$

де W, H – відповідно ширина та висота вхідного зображення,
 k – номер блоку з кінця.

Шар пулінгу понижує розмірність виходу блоку Собеля в k разів.

Окрім розглянутих мереж U-Net та FPN, в дослідженнях також були задіяні інші згорткові мережі, призначені для сегментації: PSPNet та DeepLab v3. Оскільки ці мережі мають іншу структуру декодера, результат роботи блоку Собеля було запропоновано конкатенувати з виходом блоку енкодера (bottleneck) (рисунок 2.10). Додатково вихід після pooling-шару був переданий через шар згортки.

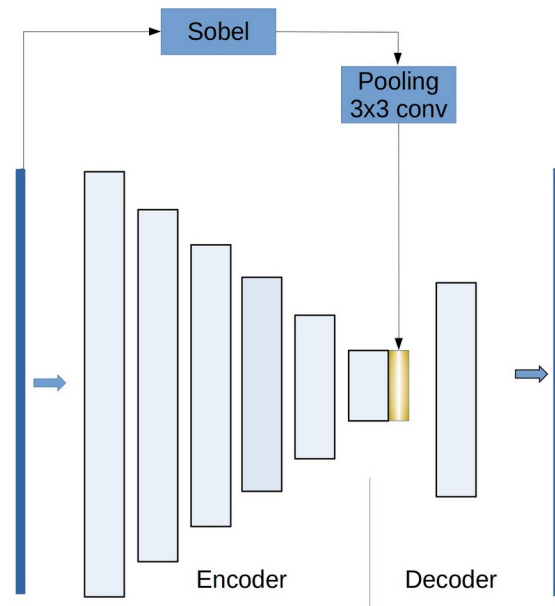


Рисунок 2.10. Реалізація блоку Собеля в мережах DeepLab v3 та PSPNet

Модифікацію мереж реалізовано мовою Python на фреймворку Keras з використанням бібліотеки `segmentation_models` [58].

2.2.2. Покращення блоку декодеру мережі U-Net

Класичний U-Net блок декодера представляє собою 2 згортки. В дисертаційній роботі було запропоновано реалізацію цього блоку на основі Resnet та Inception сегментів (рисунок 2.11).

В блоці реалізовано N inception-сегментів, де N — залежить від глибини розташування блоку. Паралельно із застосуванням Inception сегментів обчислюється Resnet сегмент. Результат обох сегментів поєднується (конкатенується) та подається на наступний шар. На рисунку 2.11 числами зазначені розміри фільтрів згортки. В таблиці 2.1 представлено залежність N від глибини та кількість шарів вихідного шару.

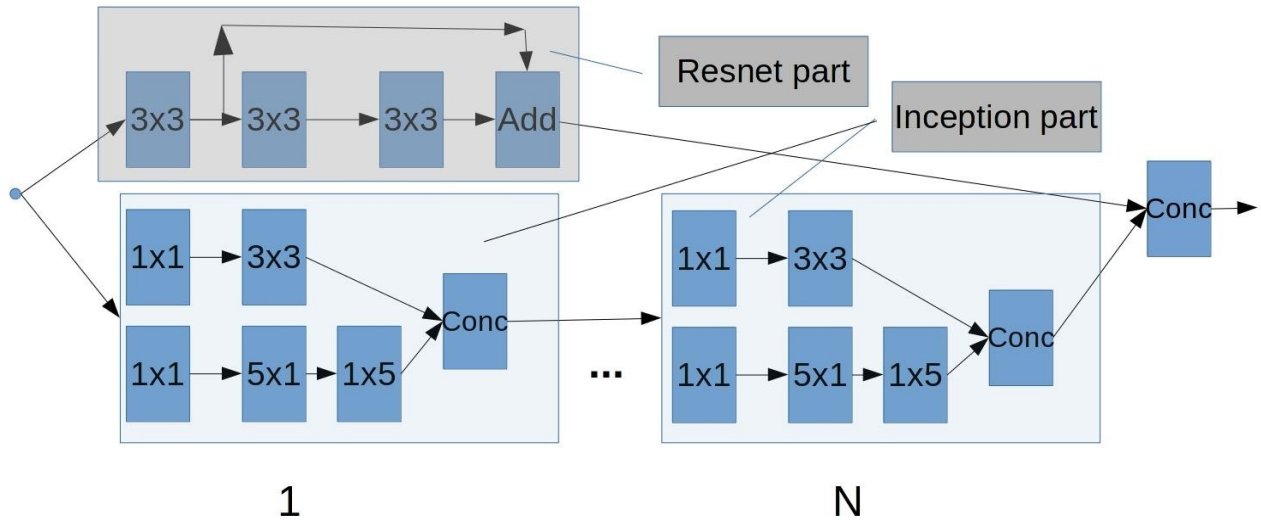


Рисунок 2.11. Блок декодера запропонованої архітектури

Таблиця 2.1. Залежність кількості Inception-сегментів від глибини розміщення блоку.

| Глибина розміщення блоку | Кількість Inception сегментів | Розмірність виходу блоку |
|--------------------------|-------------------------------|--------------------------|
| 1 | 1 | 128*128*64 |
| 2 | 2 | 64*64*128 |
| 3 | 3 | 32*32*256 |
| 4 | 5 | 16*16*512 |
| 5 | 7 | 8*8*768 |

Також в запропоновану мережу інтегровано PSP-блок. Блок PSP використовується після третього блоку енкодера.

Вхідними даними цього блоку є матриця розмірності $c*w*h$, де c — кількість каналів, w, h — ширина та висота. Блок PSP перетворює вхідний масив на масив зі стиснених карт признаков, до яких застосовуються згортки. Вихідні масиви після операцій згорток відновлюються до початкової розмірності та конкатенуються з вхідним сигналом.

Вихідний результат PSP поєднується з результатом U-Net. Після цього відбувається фінальна згортка фільтром 1×1 та сигмоїдальна активація. На рисунку 2.12 наведено модифіковану нейронну мережу U-Net архітектури з PSP блоком. Також в цій мережі блок декодера замінено на запропонований.

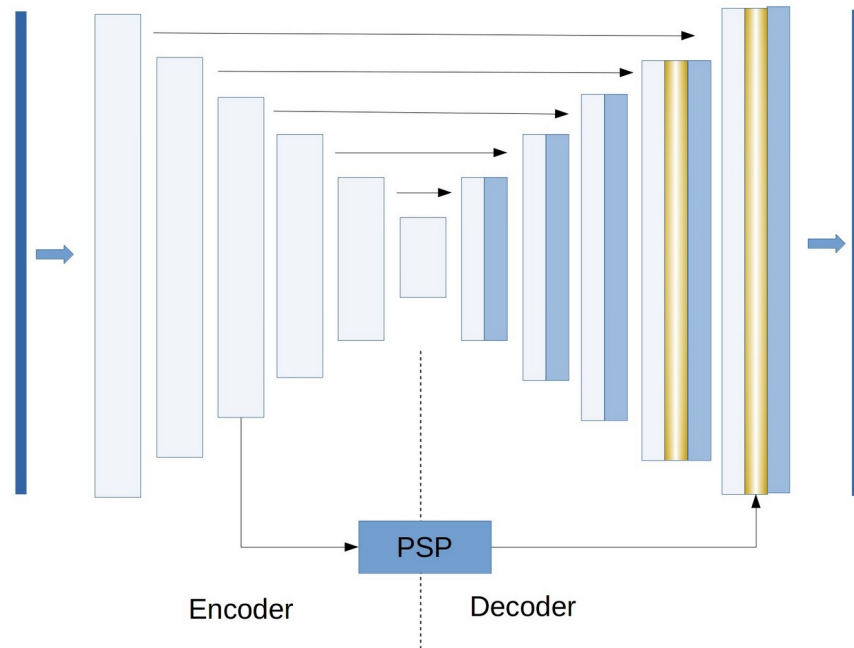


Рисунок 2.12. Модифікована мережа U-Net з блоком PSP

Модифікацію мереж, як і попередню модифікацію, реалізовано мовою Python на фреймворку Keras з використанням бібліотеки `segmentation_models` [60].

Оцінку якості роботи нейронних мереж можна отримати за допомогою метрик.

2.2.3. Метрики якості в задачах семантичної сегментації

Виділяють декілька способів оцінювання якості роботи алгоритму семантичної сегментації: об'єктивний та суб'єктивний [61]. Недолік суб'єктивних способів оцінювання полягає в тому, що його оцінює людина. Об'єктивні методи використовують не візуальні, а кількісні показники. Вони базуються на порівнянні вихідного результату алгоритму з еталоном.

На рисунку 2.13 представлено схему класифікації критеріїв оцінювання якості семантичної сегментації.

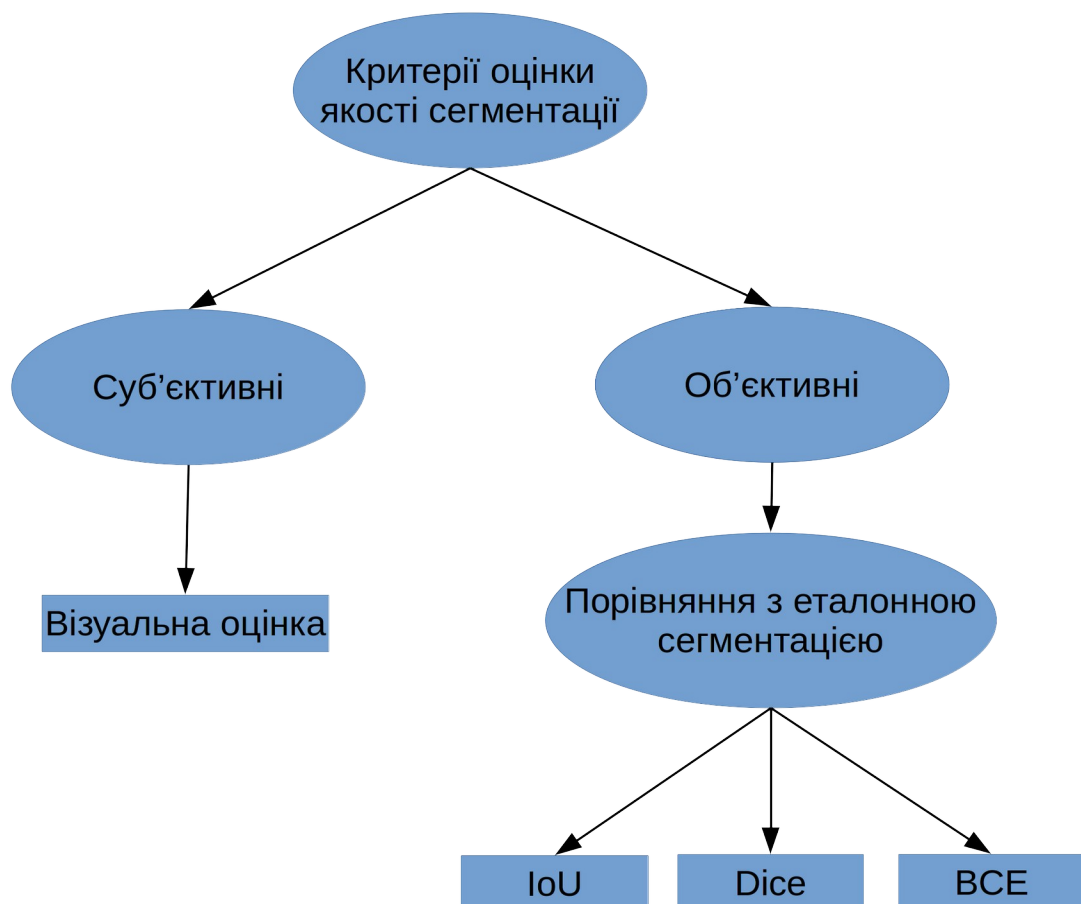


Рисунок 2.13. Класифікація критеріїв оцінювання якості семантичної сегментації.

В якості **об'єктивних метрик** для задач семантичної сегментації найчастіше застосовують IoU (Intersection-over-Union), коефіцієнт Dice та binary cross entropy (BCE) або categorical cross entropy — CCE у випадку мультикласової сегментації.

Метрика BCE (або CCE) показує, наскільки точно алгоритм провів класифікацію кожного пікселю. Метрики IoU та Dice враховують співпадіння цільових областей, тим самим надаючи більш коректну оцінку розв'язання задач семантичної сегментації.

Метрика IoU — це відношення площі перекриття передбаченої сегментації та істинної (еталонної) до області об'єднань цих областей:

$$Score_{IoU} = \frac{|A \cap B|}{|A \cup B|}, \quad (2.11)$$

де A — площа області сегментації, яка згенерована алгоритмом,

B — площа області сегментації на еталонному зображенні.

В комп'ютерних зображеннях під площею мається на увазі кількість пікселів, які знаходяться в даній області.

Метрику IoU можна відобразити графічно (рисунок 2.14).

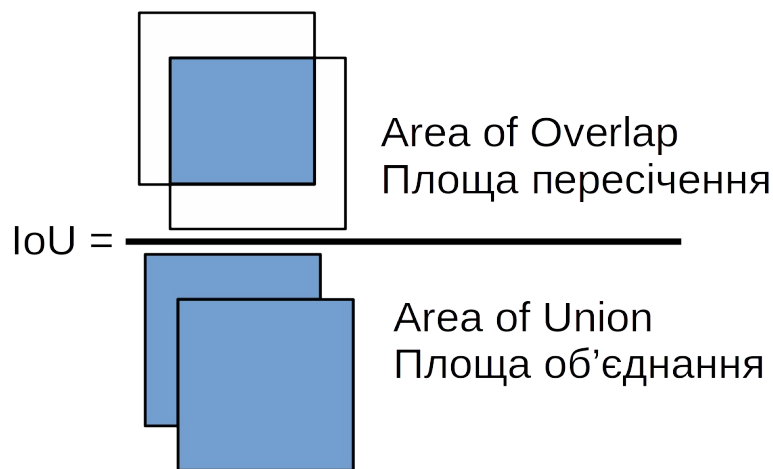


Рисунок 2.14. Графічний сенс метрики IoU

Метрика Dice обчислюється як відношення двох площ перекриття між істинними зображенням та еталонним до суми їх площ:

$$Score_{Dice} = \frac{2|A \cap B|}{|A+B|}, \quad (2.12)$$

A — область сегментації, яка згенерована алгоритмом,

B — область сегментації на еталонному зображенні.

Як і метрику IoU, метрику Dice можна відобразити графічно (рисунок 2.15).

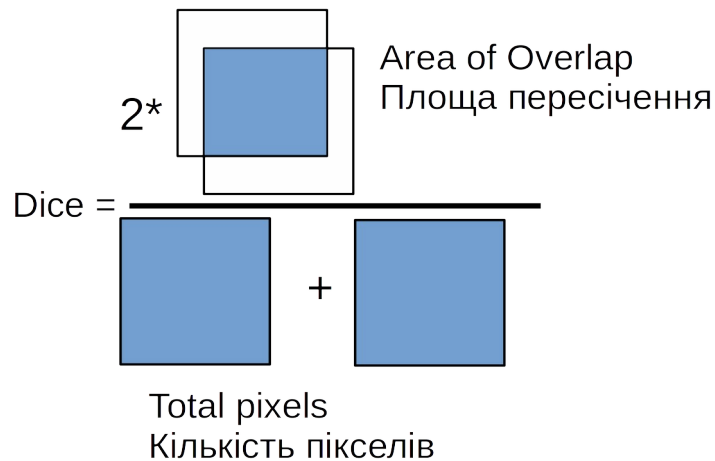


Рисунок 2.15. Графічний сенс метрики Dice

З рисунків 2.14 та 2.15 можна зрозуміло, що метрики Dice та IoU не залежать від масштабу зображення та мають результат в проміжку $[0, 1]$ (1 — якщо передбачення співпало з еталоном з точністю до пікселю).

Метрика Dice є диференційованою, тому при навчанні нейронних мереж для сегментації часто застосовують її як функцію втрат. Для того, щоб контролювати правильність класифікації кожного пікселя, додатково застосовують функцію втрат BCE. Тобто фінальна функція втрат є сумою BCE та Dice.

2.3. Обчислювальні експерименти з розв'язання задачі семантичної сегментації

2.3.1. Тестові задачі

Обчислювальні експерименти було проведено для розв'язання таких задач:

1. Salt identification Challenge [62] — задача ідентифікації сольових відкладень в надрах землі.

2. Cityscapes [63] — розпізнавання вуличних сцен заданих об'єктів зі стандартних класів.

Перша задача полягає в визначенні сольових проміжків в надрах землі, запропонована в міжнародному співтоваристві Kaggle. Це задача бінарної сегментації, тобто потрібно визначити, чи присутні відкладення солі на сейсмічних знімках. Датасет містить 4000 одноканальних зображень розмірністю 101x101 пікселів. Перед представленням нейронній мережі всі приклади були доповнені до розмірності 128x128. Датасет був поділений у пропорції 80:20 (80% - тренувальна вибірка, 20% - валідаційна).

Cityscape відноситься до класичних задач сегментації заданих об'єктів на фотографіях вуличних сцен. Набір даних, отриманий з ресурсу [64], містить 2975 зображень в тренувальній вибірці та 500 зображень в валідаційній. Сегментація здійснюється за 13 класами, таких як автомобіль, пішохід, дорожній знак та інші. Розмірність зображення – 256x256 пікселів – перед подачею до нейронної мережі була знижена до 128x128 пікселів.

Метрикою точності в усіх експериментах була Intersection over Union.

Функцією втрат для першої задачі було обрано суму фінцій бінарної крос-ентропії (binary cross entropy); для другої — категорійну крос-ентропію (categorical crossentropy) та Dice. Початковий коефіцієнт навчання – 0.002. Під час навчання використовувалося автоматичне пониження коефіцієнту швидкості навчання в 5 разів, якщо IoU на валідаційній вибірці не

знижувався протягом 5 епох. Також був використаний принцип ранньої зупинки, за яким навчання зупинялося, якщо мережа не покращувала показники IoU протягом 15 епох. Після завершення навчання в мережу завантажувалися ваги тієї епохи, на якій був досягнутий найбільший IoU на валідаційний вибірці.

2.3.2. Обчислювальні експерименти на мережах U-Net та FPN

В ході експериментів були протестовані базові та модифіковані запропонованим способом мережі U-Net, U-Net з увагою (UNet-Attention) та FPN. Увага добавлялася в Skip-з'єднання. В якості backbone-мережі обрано ResNet34, ResNet50 та SE-ResNext50. Для другої задачі ваги backbone-мережі були натреновані на ImageNet. Навчання проводилося на комп'ютері з відеокартою Nvidia GTX 1080Ti. Розмірність батчу при тренуванні була 64.

Для підвищення якості навчання було застосовано аугментацію. Аугментацію реалізовано як функцію спотворення, яка одночасно застосовувалася до вхідного зображення і до маски. З аугментацій при навчанні застосовувалися лише поворот відносно вертикальної осі (horizontal flip) з ймовірністю 0.5. Аугментація під час тесту (test time augmentation – TTA) при передбаченні нейронною мережею не застосовувалася.

Проведено такі серії обчислювальних експериментів:

- на базових моделях без модифікацій;
- на моделях з додаванням оператора Собеля до останнього блоку;
- на моделях з додаванням оператора Собеля до двох останніх блоків;
- на моделях з додаванням оператора Собеля до трьох останніх блоків.

Результати обчислювальних експериментів на задачі Salt Identification Challenge зведено в таблицю 2.2.

Таблиця 2.2. Результати обчислювальних експериментів на задачі Salt identification Challenge для модифікацій з блоком Собеля

| Архітектура | | IoU на валідаційній вибірці на моделі: | | | |
|--------------------|--------------|--|---------------------------------|----------------|----------------|
| Тип | Backbone | Базовий | З використанням блоку Собеля на | | |
| | | | 1 шарі | 2 шарах | 3 шарах |
| U-Net | ResNet34 | 0.80672 | 0.81241 | 0.81372 | 0.81268 |
| | ResNet50 | 0.76080 | 0.76287 | 0.76854 | 0.76901 |
| | SE-ResNext50 | 0.77962 | 0.78241 | 0.78400 | 0.78397 |
| U-Net Attention | ResNet34 | 0.80821 | 0.81429 | 0.81463 | 0.81487 |
| | ResNet50 | 0.76357 | 0.86520 | 0.86613 | 0.86579 |
| | SE-ResNext50 | 0.78547 | 0.78731 | 0.78812 | 0.78742 |
| FPN | ResNet34 | 0.79366 | 0.79922 | 0.80470 | 0.80240 |
| | ResNet50 | 0.75942 | 0.76157 | 0.76248 | 0.76173 |
| | SE-ResNext50 | 0.76439 | 0.76630 | 0.76721 | 0.76739 |

Аналогічно було проведено серію експериментів для задачі Cityscapes. Результати зведено в таблицю 2.3.

Окремо задача Salt identification Challenge була протестована з використанням кросс-валідації на платформі Kaggle. Датасет був поділений на 5 частин, 4 частини використовують для навчання, 1 для валідації (рисунок 2.16).

Таблиця 2.3. Результати обчислювальних експериментів на задачі Cityscapes для модифікацій з блоком Собеля

| Архітектура | | IoU на валідаційній вибірці на моделі: | | | |
|--------------------|--------------|--|---------------------------------|----------------|----------------|
| Тип | Backbone | Базовий | З використанням блоку Собеля на | | |
| | | | 1 шарі | 2 шарах | 3 шарах |
| U-Net | ResNet34 | 0.42502 | 0.43308 | 0.43603 | 0.43519 |
| | ResNet50 | 0.43106 | 0.43968 | 0.44119 | 0.44011 |
| | SE-ResNext50 | 0.43127 | 0.43983 | 0.44245 | 0.44210 |
| U-Net Attention | ResNet34 | 0.46703 | 0.47806 | 0.48040 | 0.47912 |
| | ResNet50 | 0.47142 | 0.47593 | 0.47684 | 0.47761 |
| | SE-ResNext50 | 0.47390 | 0.47481 | 0.47927 | 0.47917 |
| FPN | ResNet34 | 0.50421 | 0.51145 | 0.51474 | 0.51320 |
| | ResNet50 | 0.50944 | 0.51153 | 0.51287 | 0.51231 |
| | SE-ResNext50 | 0.51041 | 0.51289 | 0.51323 | 0.51376 |

Послідовно навчалися 5 моделей мереж U-Net архітектури за приведеною схемою. Після навчання, в робочому режимі, результат всіх мереж усереднювався, тобто визначалося середнє значення:

$$\text{Output} = \frac{1}{\text{FoldCnt}} \sum_{i=0}^{\text{FoldCnt}} \text{Output}_i, \quad (2.13)$$

де FoldCnt — кількість вибірок розбиття (дорівнює кількості мереж),
Output_i — результат передбачення i-ї мережі.

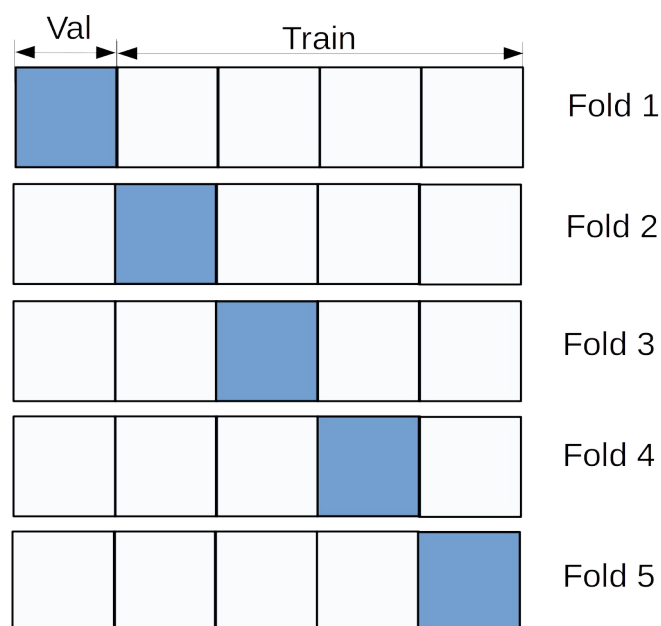


Рисунок 2.16. Схема кросс-валідації на 5 частин

Результати розв’язання задачі Salt identification Challenge, оцінені платформою Kaggle наведено в таблиці 2.4.

Таблиця 2.4. Результати розв’язання задачі Salt identification Challenge, оцінені платформою Kaggle для модифікацій з блоком Собеля

| Архітектура | | IoU, оцінений платфорою Kaggle | | | |
|--------------------|----------|--------------------------------|---------------------------------|---------------|---------------|
| Тип | Backbone | Базовий | З використанням блоку Собеля на | | |
| | | | 1 шарі | 2 шарах | 3 шарах |
| U-Net | ResNet34 | 0.8220 | 0.8222 | 0.8223 | 0.8223 |
| U-Net Attention | ResNet34 | 0.8225 | 0.8228 | 0.8230 | 0.8229 |
| FPN | ResNet34 | 0.8220 | 0.8222 | 0.8223 | 0.8222 |

З таблиць видно, що оператор Собеля у будь якому випадку покращує точність сегментації. Також очевидно, що при зміні архітектури необхідно визначати, яка кількість доданих блоків Собеля є оптимальною.

2.3.3. Обчислювальні експерименти на мережах PSPNet та DeepLab v3

Ці експерименти проводилися за тією самою схемою, визначеною в пункті 2.5.2. Їх виокремлено в додаткову серію завдяки тому, що архітектури PSPNet та DeepLab v3 відрізняються від U-Net та FPN архітектур і мають інший спосіб підключення блоку Собеля. Результати обчислювальних експериментів зведено до таблиці 2.5.

Таблиця 2.5. Результати розв'язання обчислювальних експериментів на задачах Salt identification Challenge та Cityscapes на мережах DeepLab v3 та PSPNet для модифікацій з блоком Собеля

| Архітектура | | IoU на валідаційній вибірці на моделі: | |
|--------------------------------------|--------------|--|------------------------------|
| Тип | Backbone | Базовий | З використанням блоку Собеля |
| Задача Salt identification Challenge | | | |
| DeepLab v3 | ResNet34 | 0.7904 | 0.7929 |
| | SE-ResNext50 | 0.7961 | 0.8003 |
| PSPNet | ResNet34 | 0.8090 | 0.8123 |
| | SE-ResNext50 | 0.8162 | 0.8192 |
| Задача Cityscapes | | | |
| DeepLab v3 | ResNet34 | 0.42373 | 0.42398 |
| | SE-ResNext50 | 0.42398 | 0.42442 |
| PSPNet | ResNet34 | 0.42368 | 0.42395 |
| | SE-ResNext50 | 0.42387 | 0.42414 |

Таким чином, експериментально доведено, що додавання блоку Собеля підвищує точність розв'язання задачі Salt identification Challenge:

- з 0.80821 до 0.81463 для мереж U-Net архітектури;
- з 0.79366 до 0.80470 для мереж FPN архітектури;
- з 0.7961 до 0.8003 для мереж DeepLab v3 архітектури;
- з 0.8162 до 0.8192 для мереж PSPNet архітектури.

Експериментально доведено, що додавання блоку Собеля підвищує точність розв'язання задачі Cityscapes:

- з 0.42502 до 0.43603 для мереж U-Net архітектури;
- з 0.50421 до 0.51474 для мереж FPN архітектури;
- з 0.42398 до 0.42442 для мереж DeepLab v3 архітектури;
- з 0.42387 до 0.42414 для мереж PSPNet архітектури.

2.3.4. Обчислювальні експерименти на мережах з модифікованим декодером

Окрім додавання блоку Собеля, в роботі була запропонована нова структура блоку декодера та додавання блоку PSP (розділ 2.3).

Тому для доведення ефективності цих модифікацій поставлено такі обчислювальні експерименти:

- 1) на базовій мережі U-Net архітектури;
- 2) на аналогічній мережі з додаванням блоку PSP;
- 3) на аналогічній мережі з модифікацією блоку декодера;
- 4) на аналогічній мережі з використанням обох модифікацій 2 і 3;
- 5) на аналогічній мережі з використанням модифікацій 2 і 3 та блоком Собеля.

Результати розв'язання задачі Salt identification Challenge для вище зазначених експериментів наведено в таблиці 2.6.

Таблиця 2.6. Результати розв'язання задачі Salt identification Challenge мережі U-Net

| № | Backbone | Наявність блоку PSP | Модифікація блоку декодеру | Наявність блоку Собеля на 2-ох шарах | Метрика IoU |
|----|--------------|---------------------|----------------------------|--------------------------------------|----------------|
| 1 | ResNet34 | | | | 0.80672 |
| 2 | | | ✓ | | 0.80811 |
| 3 | | ✓ | | | 0.80714 |
| 4 | | ✓ | ✓ | | 0.81017 |
| 5 | | ✓ | ✓ | ✓ | 0.81469 |
| 6 | ResNet50 | | | | 0.76080 |
| 7 | | | ✓ | | 0.76258 |
| 8 | | ✓ | | | 0.76307 |
| 9 | | ✓ | ✓ | | 0.76512 |
| 10 | | ✓ | ✓ | ✓ | 0.76916 |
| 11 | SE-ResNext50 | | | | 0.77962 |
| 12 | | | ✓ | | 0.78229 |
| 13 | | ✓ | | | 0.78276 |
| 14 | | ✓ | ✓ | | 0.78352 |
| 15 | | ✓ | ✓ | ✓ | 0.78512 |

Результати розв'язання задачі Cityscape для вище зазначених експериментів наведено в таблиці 2.7.

Таблиця 2.7. Результати розв'язання задачі Cityscape мережі U-Net.

| № | Backbone | Наявність блоку PSP | Модифікація блоку декодеру | Наявність блоку Собеля на 2-ох0 шарах | Метрика IoU |
|----|--------------|---------------------|----------------------------|---------------------------------------|----------------|
| 1 | ResNet34 | | | | 0.42502 |
| 2 | | | ✓ | | 0.42706 |
| 3 | | ✓ | | | 0.42750 |
| 4 | | ✓ | ✓ | | 0.42825 |
| 5 | | ✓ | ✓ | ✓ | 0.43917 |
| 6 | ResNet50 | | | | 0.43106 |
| 7 | | | ✓ | | 0.43364 |
| 8 | | ✓ | | | 0.43352 |
| 9 | | ✓ | ✓ | | 0.43408 |
| 10 | | ✓ | ✓ | ✓ | 0.44517 |
| 11 | SE-ResNext50 | | | | 0.43127 |
| 12 | | | ✓ | | 0.43320 |
| 13 | | ✓ | | | 0.43327 |
| 14 | | ✓ | ✓ | | 0.43401 |
| 15 | | ✓ | ✓ | ✓ | 0.44678 |

Позначкою ✓ позначено використання відповідної модифікації мережі U-Net. З таблиць видно, що найкращого результату можна досягнути з використанням всіх способів.

Експериментально доведено, що всі вказані модифікації підвищують точність розв'язання задач: Salt identification Challenge з 0.80672 до 0.81469; Cityscapes з 0.42502 до 0.43917.

2.4. Класифікація сигналів великої розмірності

Обробка сигналів, зокрема зображень в сучасних системах, як правило, здійснюється нейронними мережами. Кожна така задача має визначений математичний апарат, який враховує її специфіку. Класифікація зображень є розповсюдженою прикладною задачею, розв'язання якої необхідно в таких галузях, як медицина, астрономія, фізика, автоматизація керування та багатьох інших. Однак не завжди існуючі алгоритми забезпечують достатню точність в поточному випадку. Так, згорткові нейронні мережі (convolutional neural network, CNN), що відмінно зарекомендували себе при аналізі зображень, не завжди забезпечують достатню точність при класифікації зображень великої розмірності.

Достатньої точності розв'язання таких задач можна досягти використанням мереж глибокого навчання з великою кількістю шарів. Однак при роботі такої мережі (в inference-режимі) необхідно застосувати значні комп'ютерні ресурси, які не завжди доступні кінцевому користувачу. Тому оптимізація архітектури нейронної мережі за критеріями точності класифікації при зменшенні кількості шарів є актуальною задачею та має практичну значущість.

2.4.1. Згорткові нейронні мережі, пристосовані для розв'язання задач класифікації.

Сучасною базовою архітектурою мереж з великою кількістю шарів вважається ResNet [30] та її модифікації. Особливість цієї мережі полягає в

skip-з'єднаннях, які дозволили навчати глибокі мережі. Найпопулярнішими модифікаціями архітектури ResNet є:

- ResNext [31], суть якої полягала у використанні групових згорткок;
- SeResNet [37], в якій використовувалися механізми уваги;
- SeResNext[37], в якій синтезовано перші 2 підходи.

Концепції цих архітектур дозволяють підвищити ефективність за рахунок додавання шарів. На відміну від цих мереж EfficientNet [65] дозволяє масштабування як в глибину (тобто за рахунок збільшення кількості шарів), так і в ширину (збільшенням кількості вихідних каналів).

Зазначені мережі мають багато модифікацій, в залежності від кількості шарів, наприклад, ResNet18, EfficientNet-b0. Ці модифікації добре себе зарекомендували як мережі з відносно невеликою кількістю шарів з, як наслідок, достатньо швидким навчанням.

Мережа WaveNet [66] була розроблена для генерації звукових сигналів. Її концепція полягає в застосуванні проріджених згорткок (dilated-convolution). Така мережа може застосовуватися для багатьох інших задач, в тому числі, задачі класифікації, за умови заміни останніх шарів такими, що відповідають поточному класу задачі.

Проблема полягає в тому, що реалізації таких мереж, як ResNet18, EfficientNet-b0, WaveNet з кількістю шарів, які задовольняють вимогам щодо обмежених обчислювальних ресурсів, не надають достатню точність класифікації на сигналах великої розмірності. В загальному випадку при збільшенні розмірності сигналу, який подається на нейронну мережу, точність класифікації зростає. Однак для згорткових нейронних мереж з обмеженою кількістю шарів, починаючи з деякого критичного значення розмірності, точність зменшується. Погіршення точності розпізнавання пов'язане з тим, що рецептивне поле нейронів останнього шару не 'покриває' всього вхідного сигналу. Тому для вирішення цієї проблеми необхідно модифікувати архітектуру згорткової нейронної мережі таким чином, щоб надати можливість використання нейронних мереж з відносно невеликою

кількістю шарів, забезпечуючи при цьому достатню точність класифікації на сигналах великої розмірності.

2.4.2. Блок агрегації сигналів різної розмірності

В основу концепції покладено ідею збільшення рецептивного поля нейрона. Звичайно цього досягають за рахунок збільшення кількості шарів нейронної мережі, пониження розмірності вхідного сигналу або використання проріджених згорток. В даній роботі запропоновано реалізувати в мережі глибокого навчання додатковий блок. Вхідним сигналом цього блоку є карта признаков з нейронної мережі. Функцією цього блоку є пониження розмірності вхідного сигналу з подальшим застосуванням згортки. Оскільки після цих операцій вхідна та вихідна розмірність не співпадають, в блоці необхідно передбачити відновлення розмірності до початкових розмірів. Вихідний сигнал додається до вхідного сигналу, таким чином кількість каналів не змінюється.

На рисунку 1 зображено структуру додаткового блоку, який названо блоком агрегації сигналів різної розмірності — AggrBlock.

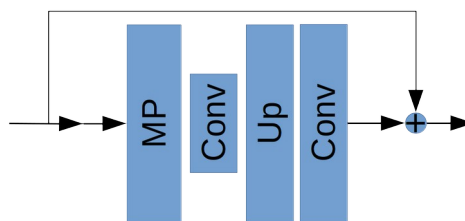


Рисунок 2.17. Структура блоку агрегації сигналів різної розмірності (AggrBlock)

Запропонований блок містить такі компоненти:

- MP — шар MaxPooling, який понижує розмірність сигналу вдвічі.

В проведених дослідженнях було протестовано варіант заміни цього шару згорткою з кроком (stride) 2, але такий варіант показав гірші результати.

- Conv — шари згорток відповідно до типу задачі 1D або 2D з розміром ядра 3, а також шари нормалізації та активації.

- Up — шар підвищення поточної розмірності сигналу до початкової;

- оператор додавання вхідного сигналу до вихідного.

Для того, щоб, по перше, зберегти можливість використання попередньо навчених мереж, та, по друге, не збільшувати кількість параметрів мережі, результат блоку не конкатенується з вхідним сигналом, а додається до нього. Тобто можна зафіксувати ваги всієї мережі крім шару класифікатора та блоків агрегації і провести її навчання.

2.4.3. Архітектура модифікованої мережі

Запропоновано використовувати блоки агрегації сигналів різної розмірності AggrBlock між існуючими стандартними блоками згорткової мережі. Під стандартним блоком вважається сукупність шарів обробки сигналу однієї розмірності. Блоки агрегації суттєво збільшують рецептивне поле нейронів останніх шарів для сигналів великої розмірності, а отже і можливості мережі.

Мережі типу ResNet можна схематично зобразити як вхідний шар і 4 стандартні блоки, які містять певну кількість шарів згорток, активації та нормалізації. На рисунку 2.18 представлено каркасну схему мережі ResNet з запропонованим блоком агрегації.

Кожен наступний блок обробляє вхідний сигнал меншої розмірності, що схематично зображено на рисунку 2.18. Глобальний пулінг (GAP) зменшує розмірність сигналу до 1D шляхом агрегування. Останній компонент мережі є повноз'єднаним шаром, який здійснює безпосередньо класифікацію.

Аналогічну схему мають мережі архітектури EfficientNet.

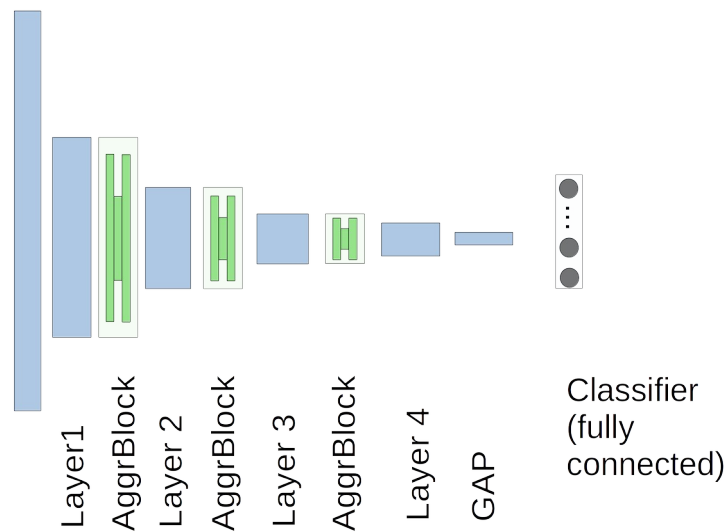


Рисунок 2.18. Каркасна схема мережі ResNet

2.5. Обчислювальні експерименти з розв'язання задачі класифікації сигналів великої розмірності

Обчислювальні експерименти проводилися для двох задач класифікації: сигналів - числових рядів (1D-сигнал) та зображень (2D-сигнал).

2.5.1. Мережі для проведення експериментів

Метою обчислювальних експериментів був порівняльний аналіз результатів класифікації для 8 мереж, перші 4 з яких мають архітектури, що не містять запропонованого блоку, а останні 4 — такі самі архітектури (з позначкою ++) з доданим блоком агрегації.

- 1) ResNet18;
- 2) EfficientNet-b0;
- 3) WaveNet;
- 4) WaveNetGRU;
- 5) ResNet18++;

- 6) EfficientNet-b0++;
- 7) WaveNet++;
- 8) WaveNetGRU++.

Каркасну схему мережі типу ResNet наведено на рисунку 2.18.

Базову реалізацію WaveNet запозичено з платформи Kaggle [67]. Оскільки тестовою задачею є аналіз послідовності сигналів, цю архітектуру доповнено рекурентними шарами, а саме Bidirectional Gated Recurrent Unit (BiGRU). Такий блок обробляє вхідний сигнал паралельно з основною архітектурою. Вихід блоку конкатенується з виходом основної архітектури, після чого подається на кінцевий повноз'єднаний шар. Таку мережу позначено WaveNetGRU.

На основі архітектур WaveNet і WaveNetGRU розроблено мережі WaveNet++ та WaveNetGRU++, які містять запропонований блок агрегації.

Схему WaveNetGRU++ представлено на рисунку 2.19.

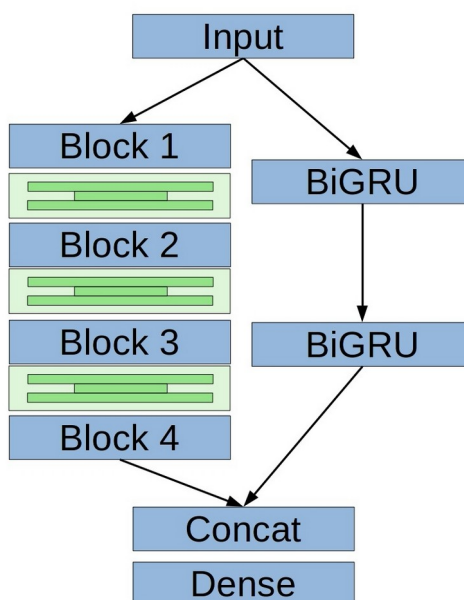


Рисунок 2.19. Архітектура мережу WaveNetGRU++

Мережа WaveNet++ формується аналогічно. Різниця полягає у відсутності BiGRU шарів.

Для обчислювальних експериментів на 1D-сигналах в якості базової архітектури було обрано WaveNet, для 2D-сигналів (зображень) — ResNet18, EfficientNet-b0, як найпростіших модифікацій мереж ResNet та EfficientNet.

2.5.2. Задача класифікації 1D-сигналів

Для цього типу класифікації тестовою задачею обрано задачу визначення кількості відкритих іонних каналів на основі електрофізіологічних сигналів [68]. Задача надана на платформі Kaggle Ліверпульським університетом.

Представлено приклади сигналу, кожен з яких складається з 500.000 дискретних значень. Тренувальний датасет містить 10 прикладів даних, тестовий — 4. Кількість вихідних класів — 11. Кількість відкритих іонних каналів визначалася для кожного дискретного значення вхідного сигналу.

Оскільки вхідний сигнал має велику розмірність, для навчання він був поділений на проміжки по 4.000 дискретних значень.

Метрикою оцінювання результатів класифікації було визначено масго F1 score, яка оцінювалася як

$$F_1 = 2 * \frac{precision * recall}{precision + recall} , \quad (2.14)$$

де *precision* — точність, яка визначається співвідношенням кількості правильно спрогнозованих прикладів, які належать даному класу, серед всіх прикладів, які класифікатор відніс до цього класу.

Recall — повнота, тобто доля правильно спрогнозованих класифікатором прикладів відносно всіх прикладів.

В свою чергу *precision* обчислюється за формулою:

$$precision = TP / (TP + FP), \quad (2.15)$$

де TP — True positive,

FP — False positive.

Recall обчислюється за формулою:

$$\text{recall} = \text{TP} / (\text{TP} + \text{FN}), \quad (2.16)$$

де FN — False negative.

Було проведено початкову серію обчислювальних експериментів для визначення оптимальної функції втрат. Оскільки метрика точності змагання подібна до метрики Dice, до тестування були додані метрики, які характерні для задач сегментації. Було проведено експерименти та проаналізовано результати розв'язання поточної задачі класифікації за такими функціями втрат:

- CCE (categorical cross entropy);
- Focal Loss [69];
- TverskyLoss [70];
- Dice loss [71].

Результати аналізувалися як за кожною функцією окремо, так і за їх сукупностями. В результаті, було обрано гібридну функцію втрат:

$$\text{loss} = \text{CCE} + \text{FocalLoss} + \text{TverskiyLoss}, \quad (2.17)$$

де CCE — categorical cross entropy loss;

FocalLoss — функція втрат для важких прикладів [8];

TverskyLoss — функція втрат для несбалансованих прикладів [9].

Навчання проводилося з використанням крос-валідації з розбиттям датасету на 5 частин. Для збільшення точності фінального результату - на відміну від тренування - в робочому режимі мережа передбачувала приклади по 4.000, 8.000, 16.000 дискретних значень вхідного сигналу. Після передбачування результат узагальнювався.

Запропоновані мережі брали участь в змаганні Kaggle. Результати їх участі наведено в таблиці 2.8. Local Cross-validation Score — результат, отриманий на валідаційних частинах вибірок, Kaggle public Score — результат 30% тестової вибірки, який оголошувався негайно, Kaggle private

score — результат 70% тестової вибірки, який оголошувався після завершення змагання і за яким проходило фінальне оцінювання результатів.

Таблиця 2.8. Результати розв'язання задачі 'Identify the number of channels open at each time point'.

| # | Neural Network | Macro F1 score | | |
|---|----------------|------------------------------|---------------------|----------------------|
| | | Local Cross-validation Score | Kaggle public Score | Kaggle private score |
| 1 | WaveNet | 0.93866 | 0.94254 | 0.94105 |
| 2 | WaveNetGRU | 0.93921 | 0.94447 | 0.94512 |
| 3 | WaveNet++ | 0.94085 | 0.94281 | 0.94406 |
| 4 | WaveNetGRU++ | 0.94225 | 0.94481 | 0.94562 |

За результатами змагань з визначення кількості іонних каналів модель WaveNetGRU++ отримала 20 місце серед 2618 учасників, дозволила отримати віртуальну срібну медаль і увійти в топ 1% учасників.

2.5.2. Задача класифікації 2D-сигналів

Тестовою задачею обрана задача класифікації за рисунком (скетчем) користувачів сервісу Google Quick Draw. Тестові дані були зібрані за допомогою цього сервісу, де користувачам пропонували намалювати нескладні об'єкти. Задача надана на платформі Kaggle компанією Google [72]. Вхідні дані представляють собою траєкторію руху миші, які перетворюються у зображення. Тренувальний набір даних містить 112.163 приклади. Задля збільшення швидкості тестування, кількість прикладів була зменшена в 10 разів, тобто до 11.216.

Тестування проводилося на зображеннях розмірності 768x768 пікселів.

Функцією втрат нейронної мережі було обрано Categorical Cross Entropy.

Метрикою оцінювання результатів класифікації платформою Kaggle було визначено Mean Average Precision @ 3 (MAP@3). Функція отримує 3 найбільш ймовірні класи, які передбачила нейронна мережа. Якщо цільовий клас міститься на першому місці — виходом метрики буде 1, на другому — 1/2, на третьому — 1/3, в іншому випадку — 0.

Навчання проводилося з розбиттям набору даних на тренувальний та валідаційний у пропорції 80%:20%.

Результати тестування для мереж ResNet18 та EfficientNet-b0 наведено в таблиці 2.9.

Таблиця 2.9. Результати розв'язання задачі 'Quick, Draw! Doodle Recognition Challenge'.

| № | Neural Network | MAP@3 score |
|---|-------------------|---------------|
| 1 | ResNet18 | 0.9112 |
| 2 | ResNet18++ | 0.9141 |
| 3 | EfficientNet-b0 | 0.9167 |
| 4 | EfficientNet-b0++ | 0.9174 |

Таким чином, запропонований спосіб модифікації архітектур згорткових нейронних мереж дозволив підвищити точність класифікації як для 1D сигналу, так і для 2D сигналу.

Висновки до розділу 2

1. Представлено базові блоки та механізми оптимізації згорткових нейронних мереж.
2. Проведено аналіз існуючих алгоритмів виявлення контурів на зображенні. Обрано оператор Собеля.
3. Запропоновано спосіб реалізації додаткового блоку на основі оператору Собеля для мереж глибокого навчання для примусового виявлення контурів на зображенні.
4. Набуло подальшого розвитку архітектурне рішення для мереж FPN, PSPNet, DeepLab v3, U-Net за рахунок вбудови запропонованого блоку Собеля для підвищення точності семантичної сегментації.
5. Набуло подальшого розвитку архітектурне рішення мереж U-Net архітектури за рахунок їх інтеграції з блоком PSP для підвищення точності семантичної сегментації.
6. Представлено тестові задачі для доведення коректності запропонованих способів.
7. Проведено обчислювальні експерименти з доведенням ефективності запропонованих методів. Експериментально доведено:
 - додавання блоку Собеля підвищує точність з 0.80821 до 0.81463 для задачі Salt identification Challenge та з 0.42502 до 0.43603 для задачі Cityscape.
 - додавання блоку PSP з модифікацією блоку декодера підвищує точність з 0.80672 до 0.81017 та з 0.42502 до 0.42825 для задачі Cityscape.
 - комбінація цих методів підвищує точність до 0.81469 для задачі Salt identification Challenge та до 0.43917 для задачі Cityscape.
8. Запропоновано блок агрегації сигналів різної розмірності, який забезпечує достатнє збільшення рецептивного поля для поточного сигналу.
9. На основі запропонованого блоку створено модифікації базових архітектур ResNet18, EfficientNet-b0, WaveNet, а також WaveNetGRU – архітектури WaveNet з використанням рекурентних шарів.

10. На основі обчислювальних експериментів доведено ефективність запропонованого способу модифікації архітектури згорткової нейронної мережі для підвищення точності класифікації сигналу великої розмірності за умови обмежених обчислювальних ресурсів для мереж ResNet18, EfficientNet-b0, WaveNet. У випадку 1D сигналу на мережі WaveNet точність за метрикою масго F1 Score підвищено з 0.94406 до 0.94562; у випадку 2D-сигналу (зображення) точність за метрикою MAP@3 — з 0.9112 до 0.914.

За результатами досліджень опубліковано 2 статті [73,74] одна з яких в міжнародному періодичному виданні, а також тези доповіді на міжнародній конференції [75].

РОЗДІЛ 3

КЛАСИФІКАЦІЯ СИГНАЛУ З ВТРАТАМИ НА КАРТАХ САМООРГАНІЗАЦІЇ

Дослідження цього розділу, є збільшення точності розпізнавання вхідного сигналу з втратами на основі нейромережі SOM.

Для досягнення цієї мети було вирішено такі задачі:

- розроблено спосіб класифікації за характеристичним сигналом з випадковими втратами на основі карт самоорганізації;
- обрано існуючі та розробити власні програмні реалізації SOM за базовим алгоритмом та алгоритмом, доповненим запропонованим способом;
- проведено верифікацію запропонованих програмних реалізацій з існуючими аналогами на основі обчислювальних експериментів;
- проведено обчислювальні експерименти з розпізнавання неповних вхідних сигналів.

В розділі проведено аналіз способів оптимізації алгоритмів функціонування нейронних мереж Кохонена – карт самоорганізації (Self-organizing map – SOM) за швидкістю навчання та часткою коректної кластеризації. Визначено ефективну оптимізацію карт самоорганізації за другим критерієм – Enhanced Self-Organizing Incremental Neural Network (ESOINN) . Визначено, що у випадку неповного вхідного сигналу, тобто сигналу з втратами в невідомі моменти часу, частка коректної кластеризації неприпустимо низька на будь-яких алгоритмах SOM, як базових, так і оптимізованих (розділ 3.1).

Неповний сигнал представлено як вхідний вектор нейронної мережі, значення якого подані єдиним масивом тобто без урахування відповідності

моментів втрат поточним значенням і без можливості визначення цих моментів. Запропоновано та програмно реалізовано спосіб визначення відповідності неповного вхідного вектора до вхідного шару нейронів для підвищення частки коректного розпізнавання. Спосіб засновано на пошуку мінімальної відстані між поточним вхідним вектором та вектором-ваг кожного з нейронів. Для зменшення часу роботи алгоритму запропоновано оперувати не окремими значеннями вхідного сигналу, а їх неподільними частинами та відповідними групами вхідних нейронів (розділ 3.2). Запропонований спосіб реалізовано для SOM та ESOINN. Для доведення ефективності реалізації базового алгоритму SOM проведено його верифікацію з існуючими аналогами інших розробників.

Розроблено схему імітації втрат вхідних даних та згенеровано тестові вибірки для обчислювальних експериментів на неповних сигналах.

На основі експериментів доведено ефективність запропонованого способу для класифікації за неповним вхідним сигналом на основі карт самоорганізації як для реалізацій базового алгоритму SOM, так і для ESOINN

3.1. Постановка проблеми

Однією з базових задач в системах контролю об'єкту за вхідним сигналом їх характеристик є задача класифікації. Якщо сигнал представляється як часовий ряд доцільне використання нейронних мереж для забезпечення високого рівня коректності розпізнавання.

При втраті частини сигналу або деяких його значень у невідомі моменти часу неможливо застосовувати різновиди мереж, розмір вхідного шару яких відповідає розміру вхідного вектору. Однак можна використовувати мережі Кохонена, а саме карти самоорганізації (Self-organizing map – SOM). Концептуальною перевагою SOM, в порівнянні з іншими нейронними мережами, є можливість навчання на малій кількості

прикладів навчальної вибірки, що є суттєвим для систем автоматичного контролю, особливо для передаварійних режимів роботи обладнання.

Нейронні мережі SOM знайшли широке використання для розв'язання сучасних прикладних задач класифікації за сигналом характеристик в різноманітних галузях, наприклад:

- діагностування якості яблук та насіння кукурудзи на основі динамічного спеклу (dynamic speckle) [76]. Вхідною інформацією SOM є результати дескрипторів стану насіння;

- класифікація астрономічних об'єктів. Вхідними даними SOM є дані космічного телескопу Gaia [77];

- створення виборок локальних галактик після зіркоутворення [78]. Основним джерелом інформації є база даних Sloan Digital Sky Survey (SDSS) – проект дослідження зображень і спектрів красного зміщення зірок і галактик;

- порівняння ансамблів білків [79]. Вхідною інформацією є результати комп'ютерного моделювання ансамблів білків.

В цих прикладах результат представляється візуалізацією SOM у вигляді розфарбованої топографічної карти.

Однак при кластеризації на мережі SOM сигналу з втратами також значно зменшується частка коректного відношення до відповідного класу, що унеможлиблює використання цієї мережі при розв'язанні практичних задач, наприклад, пошуку аномалій (anomaly detection). Тому вдосконалення алгоритмів функціонування SOM для розпізнавання сигналу з втратами є актуальною задачею та має практичне значення.

3.1.2. Проблеми розпізнавання за сигналом з втратами

Нейронні мережі Кохонена, зокрема SOM, базуються на концепції самоорганізації [45]. Такі мережі за своєю спеціалізацією пристосовані для візуалізації даних і кластерного аналізу. Перевагами SOM є стійкість до

зашумлених даних, швидке навчання, невелика кількість прикладів навчальної вибірки.

Нейронна мережа Кохонена представляє з себе одношарову мережу (рисунок 3.1).

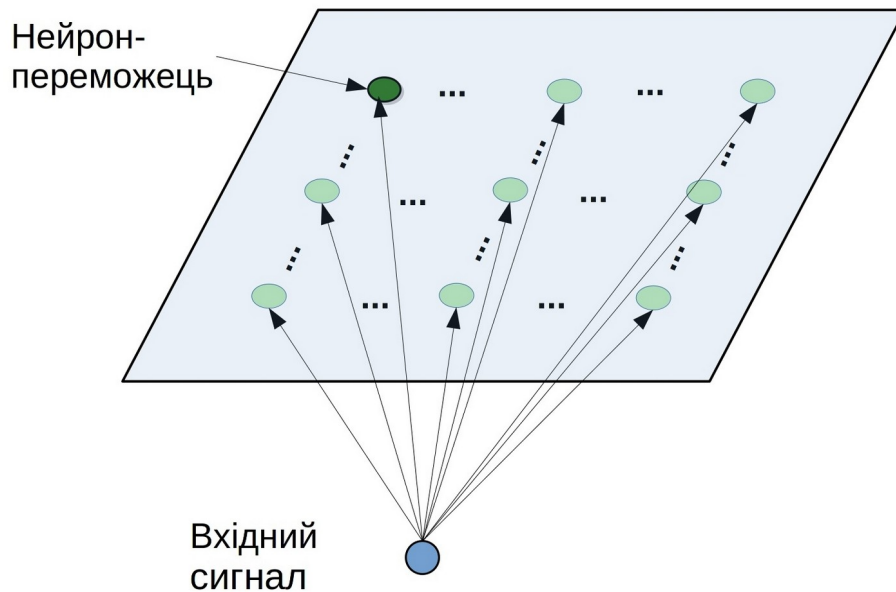


Рисунок 3.1. Нейронна мережа Кохонена

Для спрощення зображення вхідний сигнал представлено однозначним вектором, в реальних задачах цей вектор має N значень. Кожен нейрон реагує на весь вхідний сигнал.

В процесі роботи нейронних мереж Кохонена визначається єдиний нейрон-переможець, ваги якого найбільш схожі на вхідний сигнал.

Великим недоліком нейронних мереж Кохонена є надчутливість до початкового розподілу значень вагів. Вирішення цієї проблеми запропоновано в роботах [47, 51, 80-82]. В роботі [80] запропоновано ефективну схему ініціалізації SOM, яка полягає в тому, що приклади навчальної вибірки, різниця між векторами яких суттєва, спеціально ініціалізуються в різних частинах карти самоорганізації. Результати

моделювання довели, що початкова ініціалізація суттєво прискорює навчання. В [81] наведено приклад попередньої ініціалізації вершин кластерів SOM до основного навчання прикладами, для яких заздалегідь відома приналежність до певного класу. Це дозволило скоротити час навчання та примусово розділити кластери SOM.

Крім вдосконалення за початковою ініціалізацією, оптимізацію алгоритмів SOM здійснюють за такими критеріями: пришвидшення навчання та збільшення частки коректної кластеризації в робочому режимі мережі. Відповідно до цих критеріїв, можна умовно виокремити два підходи до оптимізації алгоритмів SOM.

Перший підхід містить методи, спрямовані на пришвидшення навчання.

Критеріями зупинки традиційного навчання SOM є відсутність помітних змін в вагах нейронної мережі на поточному кроці або досягнення заданої максимальної кількості кроків. В роботі [47] запропоновано новий критерій зупинки навчання – ступінь збереження топології. Оптимізована мережа краще формує кластери, які не є топологічно впорядкованими.

В роботі [51] запропоновано модифікацію алгоритму навчання на великих наборах несхожих даних для прискорення часу навчання. В роботі [81] вдосконалено алгоритм навчання на розріджених даних зі зменшенням потреб в обчислювальних ресурсах.

Використання R і k-d дерев (R-tree, k-d tree) для індексації даних з метою прискорення процесу навчання на багатовимірних наборах даних запропоновано в [82].

Загальним недоліком різновидів нейронних мереж, представлених в роботах [47, 51, 80-82], є відносно низька частка коректного розпізнавання в робочому режимі. Тому другий підхід до оптимізації алгоритмів SOM спрямовано на підвищення частки коректного розпізнавання за рахунок чіткого визначення меж кластерів.

Цей підхід полягає в створенні SOM зі змінюваною структурою, забезпечуючи, наприклад, додатковий простір для нових кластерів. Це реалізується додаванням нових фрагментів решітки SOM, які в публікаціях [46, 48-50, 84, 85] називають нодою, юнітом або кластером. Нодою є нейрон або група нейронів, які створюють новий цілий кластер або його “порцію”, що додається в процесі навчання.

В роботі [46] представлено дві моделі SOM. Перша відповідає базовій концепції Кохонена, за якою виконується навчання без вчителя. Однак задачі розв’язуються за допомогою процесу контрольованого зростання розмірності SOM і видалення нодів. Друга модель є результатом поєднання попередньої з використанням радіальних базисних функцій.

В роботі [84] наведено модель карти самоорганізації, яка додає нову ноду, якщо ваги суттєво відрізняються від наявних в мережі.

В роботі [50] описано модифікацію мережі Кохонена Growing Self-Organizing Map (GSOM), яка вирішує проблему вибору оптимальної розмірності карти самоорганізації. Мережа GSOM застосовується для задач нелінійного зменшення розмірності, апроксимації та кластеризації.

На відміну від традиційної мережі SOM, мережа GSOM не має фіксованої структури. Ця мережа стартує з 4 початкових нод. Нові ноди можуть рости на граничних (кінцевих) вузлах і приєднуватись до основної мережі (рисунки 3.2.б). Тобто мережа може динамічно розвиватися (‘зростати’) у будь-якому напрямку назовні, залежно від нових доданих вузлів (рисунки 3.2.а).

В роботі [48] представлено модифікацію карти самоорганізації – Self-Organizing Incremental Neural Network (SOINN). Мережа містить два шари, які навчаються один за одним. Перший шар використовується для визначення топологічної структури кластерів, другий – для визначення кількості кластерів і виявлення вузлів для них. Перший шар є вхідним для другого. Схожі нейрони першого шару, які об’єднані зв’язком, визначають один і той самий клас. Після закінчення навчання першого шару починає навчатися

другий шар за подібним алгоритмом. Карта “зростає” в процесі навчання, якщо відстань між вхідним вектором і наявними в мережі більше певного порогу (тобто вхідний сигнал ‘не схожий’ на вже існуючий клас), і при цьому менш чутлива до шуму вхідного сигналу.

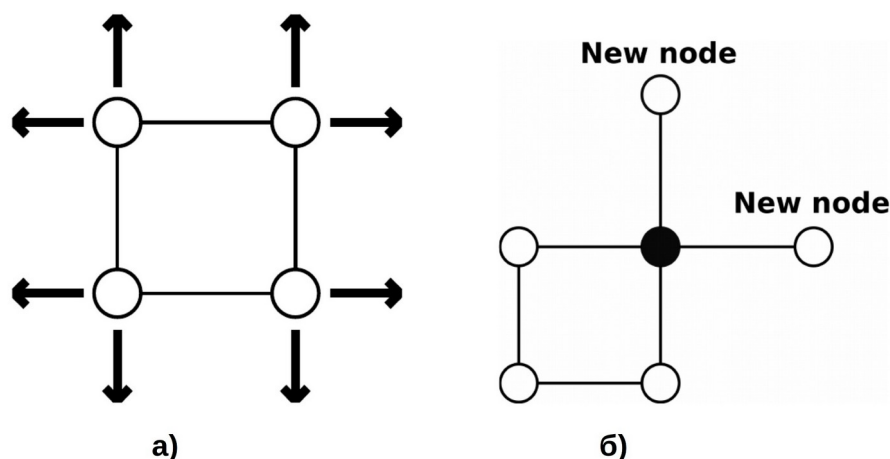


Рисунок 3.2. Початкова структура GSOM [50]:

а) можливості розширення мережі на початку ; б) створення нових вузлів на граничних нодах

В роботі [49] запропоновано карту самоорганізації, яку назвали Enhanced Self-Organizing Incremental Neural Network (ESOINN). В ній усунуто такі недоліки SOINN, як:

- 1) невизначеність моменту зупинки навчання першого шару і початку навчання другого;
- 2) проблема злиття кластерів, які близько розташовані.

В цій роботі запропоновано ефективний алгоритм, який ефективно розділяє близькі та частково перекриваючі один одного класи (рисунок 3.3).

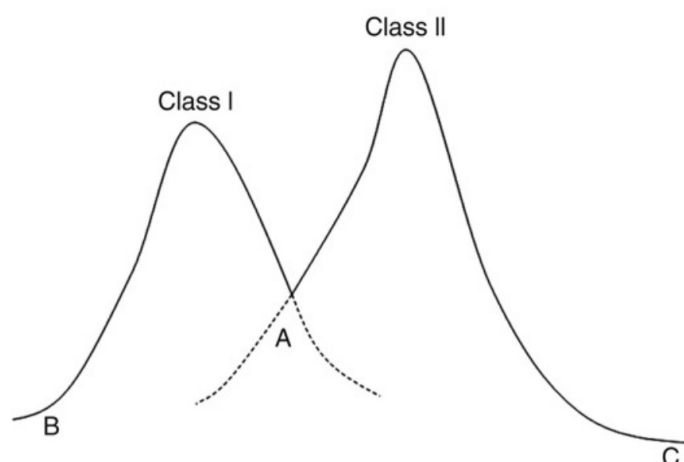


Рисунок 3.3. Розподіл ймовірності з перекриттям [49]

Таким чином, ESOINN є оптимальною за точністю розпізнавання SOM, що підтверджується зокрема результатами випробовувань на наборах даних MNIST, наведене в статті [85].

Однак і SOM, і ESOINN не надають достатню частку коректної кластеризації сигналу зі втратами. Проблема виникає тому, що в SOM нейрон-переможець обирається за мінімальною відстанню між поточним вхідним вектором і вектором ваг даного нейрона. Якщо подається повний сигнал, то розмірність поточного вхідного сигналу дорівнює розмірності вхідного шару карти самоорганізації. Таким чином, кожен i -й нейрон вхідного шару відповідає значенню i -го елемента вхідного вектора. Якщо вхідний сигнал неповний (частина сигналу втрачена), тобто розмірність вхідного вектора не дорівнює розмірності вхідного шару мережі, значення відповідних індексів вхідного сигналу і вхідного шару нейронів не співпадають, помилка зростає і сигнал кластеризується неправильно. Розв'язати проблему має додаткове попереднє визначення відповідності вхідного сигналу до нейронів вхідного шару. Тому необхідно провести дослідження та вдосконалити алгоритми функціонування SOM для досягнення коректної класифікації у випадку сигналу з втратами.

3.2. Класифікація за характеристичним сигналом з випадковими втратами на основі SOM

3.2.1. Постановка задачі для нейромережевого комплексу

Задача розпізнавання вхідного сигналу в загальному випадку полягає в визначенні відповідності вхідного сигналу

$$X_{som} = [X_1, X_2, \dots, X_n] \quad (3.1)$$

де x_i – значення вхідного сигналу в i -й момент часу; до класу a_k , який належить множині:

$$A = \{a_1, a_2, \dots, a_d\}, \quad (3.2)$$

де d — кількість класів.

Складність класифікації полягає в тому, що заздалегідь невідомо, в які моменти часу t_j були втрачені відповідні виміри сигналу x_j . Тобто на вхід SOM подається єдиний масив даних, в якому немає пропусків.

SOM розв'язує тільки задачу кластеризації, тобто визначає кластер SOM, який відповідає поточному вхідному сигналу. Після цього необхідно розв'язати задачу класифікації, тобто визначити відповідність даного кластеру класу a_k .

Для розв'язання задачі класифікації можна використати такі підходи:

- метод найближчого сусіда;
- метод опорних векторів (support vector machine, SVM) [15];
- метод потенціальних функцій [86];
- додаткова нейронна мережа.

Метод найближчого сусіда полягає в тому, що об'єкт відносять до того класу, відстань векторами яких найменша.

Метод потенціальних функцій базується на припущенні що образи, які належать різним класам, утворюють в навчальній вибірці множини, які не

перетинаються [77]. Навчальна вибірка “розповсюджує потенціал”, і об’єкт, який потрібно класифікувати, відносять до того класу, що відповідає найбільшому потенціалу.

Метод опорних векторів (SVM) полягає в тому, що будується гіперплощина, яка має розділяти навчальні виборки.

Проте дані підходи мають спільний недолік: вони дуже залежні від вибросів, тому якість класифікації може дуже змінюватися в залежності від поточної задачі.

Найбільш розповсюдженим і точним підходом є використання додаткових нейронних мереж, насамперед, багат шарового перцептрону MLP (Multilayer perceptron). На рисунку 3.4 наведено схематичне зображення комплексу SOM та MLP.

В робочому режимі навчена SOM отримує на вхід сигнал X_{som} . Рішення задачі кластеризації полягає у визначенні найбільш відповідного кластеру SOM. Це має відображення на матриці всіх виходів нейронів решітки W_{som} . Для розв’язання задачі класифікації на MLP всі строки матриці W_{som} записуються послідовно і створюють вхідний вектор MLP. Таким чином розмірність вхідного шару MLP залежить від розмірності матриці SOM. Розмірність вихідного шару MLP дорівнює кількості класів розпізнавання прикладної задачі. Кожен нейрон відповідає певному класу a_k з множини A (3.2). Вихідний вектор MLP Y відображає множину ймовірностей відповідності вхідного вектору до кожного з класів. Результатом розв’язання задачі класифікації є номер нейрона, що визначає розпізнаний клас.

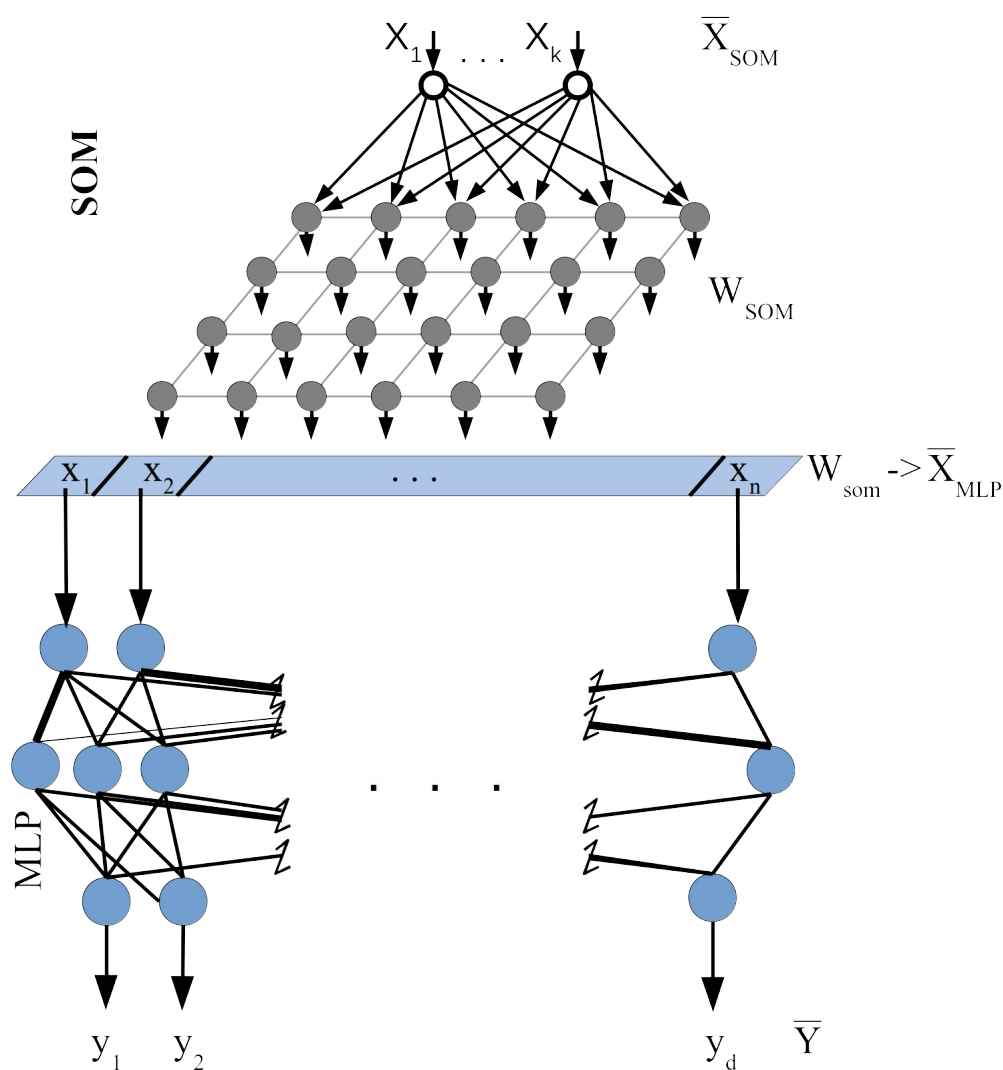


Рисунок 3.4. Розв'язання задачі класифікації на нейронних мережах SOM та MLP

Однак за умови втрати частини характеристичного сигналу, вхідний вектор не відповідає вхідному шару мережі SOM і задача кластеризації розв'язується невірно. Для того, щоб цього уникнути, необхідно здійснити попереднє визначення відповідності вхідного сигналу до нейронів вхідного шару.

3.2.2. Визначення відповідності компонентів вхідного вектору нейронам вхідного шару SOM

Проблема коректного розпізнавання за неповним сигналом може бути вирішена на основі попереднього визначення відповідності нейронів вхідного шару нейронної мережі елементам вхідного сигналу. Необхідно встановити відповідності між значенням x_i вхідного сигналу та нейроном n_j вхідного шару нейронної мережі.

Сигнал представляється як вхідний вектор нейронної мережі, значення якого подаються єдиним масивом, тобто без урахування відповідності моментів втрат поточним значенням і без можливості визначення цих моментів. На рис. 3.5 схематично показано таку відповідність для повного (рис. 3.5, а) та неповного (рис. 3.5, б) вхідних векторів. Якщо на вхід нейронної мережі подається сигнал з втратами, тобто нейронів вхідного шару більше ніж значень X_{som} . Таким чином будуть задіяні тільки перші k нейронів вхідного шару, в результаті чого результат розпізнавання буде некоректний.

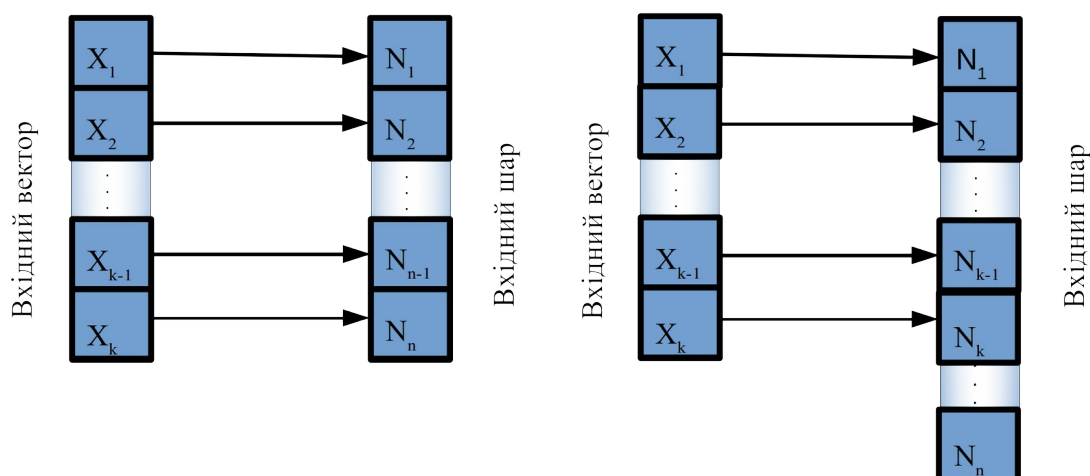


Рисунок 3.5. Відповідність вхідного вектора вхідному шару:

- а) випадок повного сигналу ($k=n$);
- б) випадок сигналу з втратами ($k < n$)

Вирішення даної проблеми відбувається визначенням оптимального для поточного сигналу співставлення нейронів вхідного шару вхідному вектору SOM за таким алгоритмом, вхідними даними якого є сигнал з втратами довжиною k , де $k < n$:

1. Встановлення початкової комбінації C_1 співставлення x_i до n_j для всіх k нейронів вхідного сигналу за правилом

$$x_i = n_i. \quad (3.3)$$

2. Подання X_{som} на вхід SOM.

3. Визначення нейрона-переможця за алгоритмом SOM.

4. Перевірка завершення алгоритму:

оскільки кількість задіяних нейронів в SOM дорівнює кількості нейронів вхідного шару, обчислюється поточна помилка нейрона-переможця (без врахування пропущених значень) як Евклідова відстань між вектором його ваг та вхідним сигналом:

$$D = |X_{som} - W_j|^2, \quad (3.4)$$

де X_{som} – поточний вхідний вектор SOM;

W_j – вектор ваг j -го нейрона-переможця.

Алгоритм завершується в таких випадках:

– якщо поточна помилка менша, ніж задане мінімальне значення помилки δ

$$D < \delta, \quad (3.5)$$

повертається номер поточного нейрону переможця;

– якщо перебрано всі комбінації співставлення компонентів вхідного вектору та нейронів вхідного шару без виконання умови (3.5), повертається результат з найменшою помилкою D .

5. Змінення поточної комбінації співставлення C_p на C_{p+1} :

відбувається як один крок перестановки зв'язку x_i до n_j . Для першої ітерації зсув відбувається для $i=k$. З кожною подальшою ітерацією i зменшується на 1.

6. Повернення до п. 2.

В результаті виконання алгоритму співставлення компонентів вхідного сигналу нейронам вхідного шару визначається оптимальна комбінація співставлення. Після цього виконується кластеризація на SOM.

Недоліком даного способу є те, що його обчислювальна складність є факторіальною і залежить від кількості втрачених значень вхідного вектору. Якщо в циклі на ранніх етапах не досягнуто виконання умови (3.5), то виконується повний перебір відповідності зв'язків компонентів вхідного вектора та нейронів вхідного шару для визначення комбінації з мінімальною похибкою.

Для вирішення проблеми стрімкого зростання обчислювальної складності в роботі запропоновано оперувати не окремими значеннями вхідного сигналу, а їх групами. Пошук відповідності між вхідним шаром і вхідним вектором встановлюється не для кожного значення вхідного вектора, а для групи. Групою нейронів вважається неподільний фрагмент вхідного сигналу з g значень x^g . Цей фрагмент є неподільним з точки зору того, що всі зв'язки та елементи групи відповідають нейронам вхідного шару за принципом:

$$x_i^g = N_j + const, \quad (3.6)$$

тобто всі зв'язки в межах групи на кожній ітерації однаково зміщуються на одне значення.

Оптимальний розмір групи g визначається наявними обчислювальними ресурсами і може визначатися експериментально як час, потрачений на класифікацію на тестових прикладах.

Чим менша розмірність групи значень вхідного вектора, тим більша обчислювальна складність. З іншого боку, чим більша кількість компонентів вхідного вектора – тим більшої точності можна досягти.

В проведених обчислювальних експериментах на тестовій задачі визначено, що коректне розпізнавання зберігається при втраті до 20 % вхідного сигналу. Таким чином, при повному вхідному вектору в 100 значень порядок складності алгоритму 10^{20} , а при використанні груп нейронів з початковою розмірністю 7 зменшується в 10^{18} раз.

3.3. Етапи розв'язання задачі класифікації за неповним сигналом

1. Реалізація програмного комплексу SOM та MLP за архітектурою, наведеною на рис. 3.4.

2. Навчання автономної SOM на навчальній виборці з прикладами повного сигналу.

3. Навчання в нейромережевому комплексі MLP за виходом SOM на тій самій навчальній виборці.

4. Визначення відповідності компонентів поточного вхідного сигналу нейронам вхідного шару SOM за запропонованим способом.

5. Класифікація на нейромережевому комплексі SOM – MLP, де SOM приведена у відповідність до поточного неповного сигналу.

Перші три етапи є типовими для класифікації і проводяться один раз. Після цього нейромережевий комплекс готовий класифікувати за повним сигналом. Четвертий етап надає можливість класифікувати неповний сигнал, що здійснюється на п'ятому етапі за алгоритмами функціонування SOM та MLP. Таким чином, класифікація кожного поточного сигналу з втратами у невідомі моменти часу полягає у виконанні двох останніх етапів.

3.4. Математична модель формування прикладів повних сигналів навчальної вибірки

Тестовий приклад є задачею розпізнавання кривих другого порядку за подібними фрагментами – верхніми частинами кола, еліпса і параболи [92].

Вхідним сигналом X_{som} є перелік дискретних значень функції, що визначає відповідну криву.

В роботі було:

- згенеровано навчальну вибірку на основі кривих другого порядку;
- проведено обчислювальні експерименти з моделюванням різних рівнів зашумленості сигналу.

Шум моделювався як нормальне розподілення:

$$\varepsilon = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3.7)$$

де параметр $\sigma=0.4$.

Якщо випадкове значення виходить за межі діапазону, то вважаємо що шум дорівнює нулю.

3.5. Реалізації нейронних мереж SOM

Для доведення ефективності запропонованого способу визначення відповідності вхідного вектору до нейронів вхідного шару розроблено такі різновиди нейронних мереж:

- 1) карта самоорганізації SOMbase, реалізована на базовому алгоритмі SOM;
- 2) карта самоорганізації SOMmod – на базовому алгоритмі SOM, доповненому запропонованим способом;
- 3) карта самоорганізації ESOINNmod – на алгоритмі ESOINN, доповненому запропонованим способом;

4) багатошаровий перцептрон MLPbase (MLP – Multilayer perceptron) на алгоритмі багатошарового перцептрона.

Перші дві версії SOM реалізовано на одних і тих самих структурах даних та базових алгоритмах для коректного порівняння базового алгоритму SOM з алгоритмом, доповненим запропонованим способом.

Для порівняння результатів розпізнавання на оптимізованих SOM обрано ESOINN з ресурсу [78], на основі якої в мережі ESOINNmod реалізовано запропонований спосіб.

Для тестування досліджуваних мереж Кохонена використовувався розроблений MLPbase, за винятком мереж ENCOG [79] та NeuroPH [80], для яких був використаний MLP з їх бібліотек. Це пов'язано з тим, що SOM розв'язує тільки задачу кластеризації, однак після цього за визначеними кластерами SOM потрібно розв'язати задачу класифікації, тобто визначити, який кластер відповідає якому класу прикладної задачі.

Для верифікації розробленої програмної реалізації SOM за базовим алгоритмом проведено обчислювальні експерименти розпізнавання повного сигналу у порівнянні з існуючими аналогами інших розробників SOM: Encog, NeuroPH. Таким чином доводиться коректність власної реалізації базового алгоритму SOMbase для його подальшого вдосконалення запропонованим способом.

Обчислювальні експерименти з розпізнавання неповних вхідних сигналів проводилися на:

1) мережах SOM без запропонованого способу: SOMbase, NeuroPH, Encog, SOINN, ESOINN, GSOM;

2) мережах SOM з запропонованим способом: SOMMod, ESOINNmod.

Після проведення обчислювальних експериментів порівнювалися частки коректного розпізнавання усіма способами.

Навчання нейронних мереж проводилося за навчальною вибіркою, яка містить 100 прикладів (34 приклади кола, 33 еліпса, 33 параболи). Кожний приклад відтворює повний сигнал, який складається зі 100 значень,

обчислених за формулами відповідних кривих. Навчалися такі карти самоорганізації: NeuraPH, ENCOG, GSOM, SOINN, SOMbase, ESOINN, SOMmod, ESOINNmod, а також багат шаровий перцепрон MLP. Початкова швидкість навчання для SOM дорівнювала 0.1. Швидкість навчання всіх SOM через перші 200 кроків примусово зменшувалась в 5 разів, а в подальшому зменшувалась ще в 5 разів кожні 50 кроків. Для навчання SOM знадобилося 400 епох.

Початкова швидкість навчання багат шарового перцепрона дорівнювала 0.01 і примусово зменшувалась в 5 разів, якщо на протязі 5 епох не спостерігалось змін в точності класифікації тестової виборки. В подальшому, якщо на протязі 15 кроків не спостерігалось змін похибки, навчання зупинялося.

Для кінцевого розв'язання задачі класифікації матриця виходів всіх нейронів кожної SOM подавалася як послідовність рядків на MLP, який визначав приналежність до відповідного класу. Для мереж Encog, NeuraPH використовувався MLP з їх бібліотек, в решті випадків – MLP власної розробки.

3.6. Верифікація реалізацій базових алгоритмів SOM та MLP

Перша серія обчислювальних експериментів проводилася для верифікації мережі SOM, розробленої за базовим алгоритмом, з існуючими аналогами, які поділяються на мережі без оптимізації: Encog [87], NeuraPH [88], та з оптимізацією SOINN [89], ESOINN [90], GSOM [91].

Тестова вибірка, за якою проводилося порівняння, містила повні сигнали вхідного вектора \bar{X} без втрат, які згенеровані за формулами відповідних кривих з урахуванням шуму.

Оскільки вибірка даних є збалансованою, коректність класифікації можна визначати за метрикою точності (accuracy):

$$Accuracy = \frac{P}{N}, \quad (3.8)$$

де P – кількість класів, які класифікатор вірно розпізнав; N – розмір навчальної вибірки.

Метою експерименту було визначення частки коректного розпізнавання за повним сигналом кожною нейронною мережею в однакових умовах. В усіх подальших експериментах частка коректного розпізнавання визначалась таким чином:

$$Cor = Accuracy * 100\%, \quad (3.9)$$

де Accuracy – метрика точності.

В таблиці 3.1 представлено результати розпізнавання всіх реалізацій SOM, які брали участь в поточному та подальших обчисленнях. Для коректного порівняльного аналізу також зазначено наявність або відсутність оптимізації поточної SOM.

Таблиця 3. 1. Результати порівняння реалізацій SOM за повним сигналом

| № п/п | Назва реалізації SOM | Наявність оптимізації SOM | Частка коректного розпізнавання, % |
|-------|----------------------|---------------------------|------------------------------------|
| 1 | SOMbase | – | 80 |
| 2 | NeuroPH | – | 79 |
| 3 | Encog | – | 81 |
| 4 | SOINN | + | 87 |
| 5 | ESOINN | + | 90 |
| 6 | GSOM | + | 86 |

Таким чином, доведено ефективність власної реалізації базового алгоритму (таблиця 3.1, п. 1–3) та підтверджено, що оптимізовані різновиди SOM на повному сигналі дають кращий результат в порівнянні з нейромережами, реалізованими за базовим алгоритмом (таблиця 3.1, п. 4–6).

3.7. Обчислювальні експерименти з класифікації за сигналом з випадковими втратами

Друга серія обчислювальних експериментів була призначена для визначення ефективності функціонування SOM за умови втрати сигналу.

Оскільки на ефективність розпізнавання впливає порядок втрат та часові проміжки їх виникнення, поставлено декілька експериментів, які відтворювали різні випадки моментів втрат. З цієї точки зору, задачі розпізнавання було розподілено на такі класи:

1. Втрата єдиної частини сигналу.
2. Втрати значень сигналу в невідомі моменти часу.

Відповідно до цих класів, було згенеровано 4 тестові виборки (перші 3 для першого класу задач та остання – для другого):

1. Втрата послідовних кінцевих значень.
2. Втрата початкових значень.
3. Втрата послідовних значень на заданому проміжку від t_{21} по t_{79} з випадковим початком.
4. Втрата значень сигналу на всьому діапазоні t_1 по t_{100} у випадкові моменти часу.

Для кожної задачі було згенеровано 100 прикладів неповного сигналу. Всі нейронні мережі, які використовувалися в експериментах, були навчені на прикладах повного сигналу. При випробовуваннях в робочому режимі на вхід SOM подавався сигнал з втратами.

В таблиці 3.2 наведено результати тестування за неповним сигналом. Курсивом виокремлено назви нейронних мереж, розробниками яких реалізовано не базовий, а оптимізований алгоритм SOM. В останніх двох стовпцях представлено результати розпізнавання з використанням запропонованого способу базовій реалізації алгоритму – SOMmod та однієї з найкращих оптимізацій за критерієм частки коректного розпізнавання – ESOINNmod.

Таблиця 3.2. Результати розпізнавання за неповним вектором

| № | Умови задачі | | Частка коректного розпізнавання в реалізаціях нейронних мереж, % | | | | | | | |
|---|---|-----------------------------|--|-------|------|-------|---------|-----------|---------------------------|-----------|
| | | | без запропонованого способу | | | | | | 3 запропонованим способом | |
| | Діапазон втрат в моменти τ_i , де $i=$ | Кількість втрачених значень | NeuraPH | ENCOG | GSOM | SOINN | SOMbase | ESOINN | SOMmod | ESOINNmod |
| Втрата єдиного фрагменту вхідного сигналу | | | | | | | | | | |
| 1 | 80–100 | 5 | 84 | 84 | 85 | 85 | 84 | 93 | 85 | 93 |
| | | 10 | 82 | 82 | 86 | 86 | 82 | 91 | 83 | 91 |
| | | 20 | 79 | 81 | 85 | 87 | 82 | 90 | 82 | 90 |
| 2 | 1–20 | 5 | 32 | 32 | 38 | 38 | 32 | 21 | 73 | 77 |
| | | 10 | 19 | 20 | 35 | 35 | 20 | 17 | 70 | 73 |
| | | 20 | 16 | 16 | 17 | 14 | 16 | 14 | 66 | 69 |
| 3 | 1–100 | 5 | 48 | 48 | 56 | 58 | 48 | 62 | 77 | 81 |
| | | 10 | 41 | 41 | 54 | 49 | 41 | 42 | 74 | 77 |
| | | 20 | 32 | 33 | 35 | 35 | 33 | 38 | 68 | 69 |
| Втрати вхідного сигналу в невідомі моменти часу | | | | | | | | | | |
| 4 | 1–100 | 5 | 68 | 69 | 69 | 71 | 68 | 74 | 74 | 79 |
| | | 10 | 54 | 54 | 58 | 60 | 54 | 56 | 61 | 66 |
| | | 20 | 37 | 37 | 39 | 39 | 37 | 43 | 69 | 71 |

З точки зору моментів втрат сигналу, найгірше розпізнаються сигнали, в яких втрачено початкові значення.

Результати розпізнавання такого сигналу на всіх оптимізованих та неоптимізованих SOM без реалізацій запропонованого способу були неприпустимо низькі та складала:

- при втраті 5 % сигналу – від 21 % до 38 %;
- при втраті 10 % сигналу – від 19 % до 38 %;
- при втраті 20 % сигналу – від 14 % до 16 %.

Всі мережі найбільш стійкі до втрат кінцевих значень. Приблизно однакові результати показали мережі з запропонованим способом і без нього; точність складала від 79 % до 95 %.

У випадку втрат сигналу на всьому діапазоні з 1-ого по 100-е значення, результат гірший. Однак експерименти показали, що запропонований спосіб значно підвищує частку коректного розпізнавання, особливо при втраті великої кількості значень.

Результати класифікації за неповним сигналом (таблиця 3.2) свідчать про те, що запропонований метод значно збільшує частку коректної кластеризації при неповних вхідних даних. Це стосується як мережі, реалізованої за базовим алгоритмом SOM, так і для оптимізованої мережі, чий алгоритм доповнено запропонованим способом.

3.8. Аналіз результатів обчислювальних експериментів

Запропоновано спосіб визначення відповідності неповного вхідного вектору до вхідного шару нейронів на основі різниці векторів вхідного сигналу і ваг нейронів вхідного шару. Згідно з цим способом, необхідно провести попереднє визначення комбінації зв'язків між компонентами вхідного вектору та нейронами вхідного шару.

Така попередня підготовка подачі сигналу на нейронну мережу зменшує помилку класифікації. Це відбувається тому, що в SOM примусово активується потрібний кластер. Для цього визначається комбінація відповідності компонентів неповного вхідного вектору нейронам вхідного

шару SOM. Таким чином мінімізується відстань між поточним вхідним вектором та вектором ваг активного нейрона за формулою (3.4), і, відповідно, помилка класифікації.

Запропонований спосіб представлено у вигляді алгоритму, кожен етап якого має необхідні формули та схему для розрахунків. Спосіб дозволяє розв'язати проблему розпізнавання діагностичного сигналу з втратами в невідомі моменти часу.

Для доведення цього способу було запропоновано відповідну тестову задачу. Необхідно було створити задачу класифікації, в якій можна моделювати приклади сигналів у вигляді часових рядів та змінювати моменти втрат їх значень. Для ускладнення постановки задачі класифікації була поставлена умова, що сигнали мають бути схожі за описом настільки, що при імітації шумових значень частково накладаються один на інший. Розв'язанням цього завдання стала математична модель генерації навчальних прикладів за аналітичними рівняннями подібних фрагментів кривих другого порядку. Таким чином створено математичну модель імітації зашумлених діагностичних сигналів, за якою можна генерувати навчальну та тестові вибірки для розв'язання складної задачі класифікації.

Для проведення подальших експериментів була розроблена власна реалізація SOM за базовим алгоритмом – SOMbase. Для доведення її ефективності, була проведена верифікація з аналогами інших розробників (таблиця 3.1).

Для обчислювальних експериментів з доведення ефективності запропонованого способу були обрані існуючі програмні реалізації SOM, створені за базовим алгоритмом (NeuraPH, Encog) та з трьома різновидами оптимізації інших розробників: GSOM, SOINN, ESOINN. Дві реалізації SOM – власна SOMbase та одна з найефективніших за часткою коректності розпізнавання ESOINN – були доповнені запропонованим способом. Інші реалізації залишились без змін.

Результати випробувань (таблиця 3.2) довели, що запропонований спосіб значно покращує коректність кластеризації, однак моменти втрат суттєво впливають на його ефективність. При найменш ефективному застосуванні показники залишаються такими самими або незначно кращими (на 1–3 %). В найкращих випадках частка коректного розпізнавання підвищується з 14 % до 69 % на доповненій запропонованим способом нейронній мережі ESOINN.

Таким чином експериментально доведено ефективність запропонованого способу класифікації за неповним сигналом.

Практична значущість роботи полягає в можливості використання запропонованого способу для діагностування стану технічних об'єктів за часовим рядом їх характеристик. Особливістю є те, що немає необхідності в перетвореннях початкового сигналу та екстракції додаткових характеристик, наприклад частотних. Це затребувано в багатьох прикладних задачах контролю поточних характеристик в реальному часі з можливими втратами сигналу через зовнішні фактори.

Недоліком способу є зростання обчислювальної складності, тобто часу класифікації. Для усунення цього недоліку запропоновано використання груп, що дозволило знизити час класифікації тестової вибірки в 10^{18} раз.

Серед обмежень даного способу є ймовірність неправильної класифікації сигналу з викидами. Саме розпізнавання таких сигналів та підвищення стійкості SOM є перспективою подальших досліджень. Це необхідно, зокрема, для розв'язання задач пошуку аномалій.

Висновки до розділу 3

1. Запропоновано метод класифікації за характеристичним сигналом з втратами у випадкові моменти часу на основі карт самоорганізації Кохонена за рахунок визначення відповідності неповного вхідного вектору вхідному шару нейронів для підвищення частки коректного розпізнавання.

2. Проведено аналіз способів оптимізації алгоритму SOM. Обрано існуючі іреалізації SOM та їх оптимізації для проведення обчислювальних експериментів: ENCOG, NeuroPH, SOINN, GSOM, ESOINN. Визначено ефективну оптимізацію SOM – ESOINN. Для обчислювальних експериментів: розроблено програмні реалізації SOM за базовим алгоритмом та алгоритмом, доповненим запропонованим способом; доповнено програмну реалізацію ESOINN запропонованим способом.

3. Для доведення ефективності реалізації базового алгоритму SOM проведено його верифікацію з існуючими аналогами: NeuroPH, Encog. Доведено, що частка коректної класифікації власної розробки (80 %) відповідає результатам такого самого алгоритму інших розробників (78 % та 81 % відповідно).

4. На основі обчислювальних експериментів доведено ефективність запропонованого способу для класифікації за неповним вхідним сигналом на основі карт самоорганізації як для реалізацій базового алгоритму SOM, так і найефективнішої оптимізації ESOINN. В залежності від моментів втрат частка коректного розпізнавання на доповненій запропонованим способом нейронній мережі ESOINN в гіршому випадку незначно покращується на 1–3 %; в найкращому випадку підвищується з 14 % до 69 %.

За результатами досліджень опубліковано 1 стаття [92] та тези доповідей на міжнародних конференціях [93, 94].

РОЗДІЛ 4

СЕРЕДОВИЩЕ МОДЕЛЮВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧІ КЛАСТЕРИЗАЦІЇ

Метою досліджень, представлених в розділі 4, є розробка середовища моделювання нейронних мереж Кохонена для порівняльного аналізу існуючих програмних реалізацій SOM при розв'язанні задачі кластеризації та доведення коректності нових модифікованих алгоритмів.

Для досягнення цієї мети вирішено такі задачі:

- 1) визначити критерії моделювання в середовищі;
- 2) розробити архітектуру середовища моделювання SOM, інтерфейси та інший допоміжний функціонал, зокрема бази даних та відповідну програмну реалізацію;
- 3) провести обчислювальні експерименти в запропонованому середовищі моделювання.

В розділі обґрунтовано обрання існуючих програмних реалізацій карт самоорганізації SOM (self-organizing map) та відповідних їм багат шарових перцептронів MLP (multilayer perceptron) для розв'язання саме задачі класифікації. Всі обрані програмні реалізації є вільнодоступними та розповсюджуються з відкритою ліцензією.

Визначено параметри SOM та MLP, на які може впливати експериментатор. Обрано критерії порівняння реалізацій SOM.

В роботі наведено архітектуру середовища моделювання і представлено функціонал його компонентів.

Для демонстрації застосування запропонованого середовища приведено розв'язання класичних тестових задач машинного навчання. Це

дозволило провести коректне порівняння результатів обчислювальних експериментів, а також ефективності програмних реалізацій як за базовим, так і за оптимізованим алгоритмами.

4.1. Програмні реалізації SOM

Задача кластеризації розв'язується в багатьох прикладних галузях. Досягнути її швидкого та достатньо точного розв'язання можна за допомогою спеціальних нейронних мереж – карт самоорганізації Кохонена. Цей тип нейронних мереж постійно вдосконалюється як на рівні алгоритмів, так і на рівні програмного забезпечення. Тому доцільно створювати спеціальні програмні інструменти, що дозволяють в однакових умовах провести навчання та безпосередньо обчислювальні експерименти з розв'язання задачі кластеризації для коректного порівняльного аналізу отриманих результатів. Другим завданням такого середовища є визначення оптимального програмного інструментарію розв'язання прикладних задач технічної діагностики, наприклад, пошуку аномалій, класифікації сигналу з втратами.

Безперервний автоматичний аналіз сигналів необхідний для штатного функціонування технічних систем. Для його реалізації широко використовуються спеціальні засоби розпізнавання, насамперед, нейронні мережі. Для розв'язання в реальному часі задач кластеризації сигналу, наприклад, при розпізнаванні передаварійних режимів роботи обладнання, доцільно використовувати нейронні мережі Кохонена — SOM (Self-organizing map). Ці мережі дозволяють розпізнавати вхідний сигнал змінного розміру і проводити навчання на обмеженій вибірці. Розв'язання кожної практичної задачі потребує пошуків архітектури нейронної мережі SOM та налаштування її параметрів, тому створення фреймворка для порівняння якості кластеризації актуальне та має практичну значущість.

Основні вдосконалення SOM можна умовно поділити на оптимізацію за критеріями швидкості навчання та точності роботи. Оскільки в робочому

режимі контролю обладнання використовується навчена мережа, подальші дослідження були спрямовані на підвищення долі коректної кластеризації SOM. Найвідомішими сучасними оптимізаціями за критерієм точності є алгоритми: GSOM (Growing Self-Organizing Maps) [50], SOINN (Self-Organizing Incremental Neural Network) [48], ESOINN (Enhanced self-organizing incremental neural network) [49].

Програмні реалізації мереж вищезазначених алгоритмів є вільнодоступними та розповсюджуються за відкритою ліцензією. Вони представлені на ресурсі Github [95]. Всі ці реалізації розраховані на кластеризацію неперервного сигналу за дискретними значеннями через сталі проміжки часу. Однак в технічній діагностиці є задачі, в яких неможливо отримати такий сигнал, наприклад, при розв'язанні задач пошуку аномалій або розпізнаванні сигналу з втратами. Для подібних задач необхідно розробити нові алгоритми, доповнені відносно базового алгоритму SOM, і, відповідно, провести їх валідацію. Приклад такої розробки наведено в статті [92] для задачі класифікації за характеристичним сигналом з випадковими втратами на основі карт самоорганізації.

Апробування нових розробок SOM потребує створення умов для їх навчання і проведення коректного порівняльного аналізу результатів тестування з існуючими програмними реалізаціями. Крім цього для розв'язання кожної прикладної задачі доцільно проводити моделювання за параметрами їх алгоритмів для обрання оптимального в кожному поточному випадку. Для розробки вдосконалених алгоритмів SOM та оптимізацій розв'язання прикладних задач на основі існуючих розробок необхідно створити середовище моделювання карт самоорганізації Кохонена з вбудованими апробованими реалізаціями алгоритмів SOM і можливістю підключення нових розробок.

4.2. Постановка задачі моделювання для нейронної мережі SOM

При створенні середовища необхідно врахувати, що SOM виконує тільки задачу кластеризації. Для кінцевого визначення належності сигналу до певного класу необхідно додатково розв'язувати задачу розділення класів. Найчастіше для цього застосовують додаткову нейронну мережу — багатошаровий перцептрон — MLP (multilayer perceptron).

Таким чином задача розпізнавання розв'язується поетапно. Спочатку SOM навчається за поданими характеристиками. Після цього двовірна матриця значень решітки W_{som} перетворюється в одноірний вектор-стовпчик \bar{X}_{mlp} , який далі подається на вхід MLP. Навчання MLP проводиться за прикладами, отриманими на задалегідь навченій SOM. На рисунку 4.1 зображено етапи розв'язання задачі класифікації з визначенням вхідної і вихідної інформації.

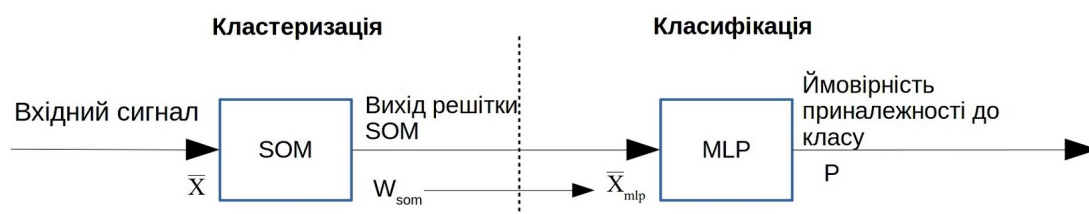


Рисунок 4.1. Етапи навчання комплексу нейромереж

Середовище моделювання SOM може бути застосоване у двох випадках:

- 1) для обрання оптимальної реалізації карти самоорганізації при розв'язанні поточної прикладної задачі;
- 2) для верифікації та коректного порівняння нових розробок SOM з існуючими.

Тому при проектуванні середовища моделювання було проведено аналіз існуючих реалізацій SOM, як з традиційними так і з оптимізованими алгоритмами функціонування.

Середовище повинно мати можливість використовувати існуючі та зручно додавати нові розробки нейромереж Кохонена та інших типів.

Для вбудови нових мереж передбачено обгортки — wrappers, які взаємодіють з усіма мережами, в тому числі і з тими реалізаціями, які імплементовано на інших мовах програмування.

В середовище вбудовано такі різновиди SOM: GSOM [91], SOINN [89], ESOINN [90], SOMbase, SOMmod [92], Encog [87], NeuroPH [88]. Мережі Encog, NeuroPH є апробованими реалізаціями базового алгоритму SOM. Мережі GSOM, SOINN, ESOINN є одними з найефективніших алгоритмів оптимізованих карт самоорганізації. Мережі SOMbase та SOMmod є прикладами доданих мереж з відповідно базовим та оптимізованим алгоритмом. Ці мережі використано для розв'язання задачі розпізнавання вхідного сигналу з втратами.

В таблиці 4.1 наведено назви всіх SOM, які включені в систему моделювання, та відповідні їм MLP.

Для середовища було розроблено власний багатошаровий перцептрон — MLPbase. Однак завдяки різним програмним реалізаціям SOM, цю мережу не можна використовувати на всіх картах самоорганізації. Тому для реалізацій Encog та NeuroPH використано MLP з їхніх бібліотек.

Таблиця 4.1. Відповідність нейромереж SOM та MLP в середовищі

| № | Мережа SOM | Мережа MLP |
|---|------------|-------------------|
| 1 | GSOM | MLPbase |
| 2 | SOINN | MLPbase |
| 3 | ESOINN | MLPbase |
| 4 | Encog | Мережа із Encog |
| 5 | NeuroPH | Мережа із NeuroPH |
| 6 | SOMbase | MLPbase |
| 7 | SOMmod | MLPbase |

Для навчання комплексу передбачено задання таких параметрів:

- 1) коефіцієнт швидкості навчання (learning rate) для SOM;
- 2) коефіцієнт швидкості навчання (learning rate) для MLP;
- 3) кількість епох до зменшення коефіцієнту швидкості навчання для SOM;
- 4) коефіцієнт зменшення швидкості навчання для SOM;
- 5) величина зменшення коефіцієнта навчання MLP при виході його на плато (Learning Rate Scheduler).

Перші два параметри впливають на швидкість навчання. Третій та четвертий — потрібні для більш точного навчання SOM, п'ятий — для коректного навчання MLP. Для визначення третього та четвертого параметру необхідно проводити експерименти з дослідження помилки на тестовій вибірці при повному циклі навчання.

Ефективність реалізації SOM для розв'язання поточної задачі визначається за такими критеріями:

- час навчання SOM (T_{SOM});
- частка коректної класифікації на комплексі SOM та MLP (D);
- час роботи на певному наборі даних (T).

4.3. Архітектура середовища

Основна проблема полягає в тому, що потрібно поєднувати існуючі реалізації нейронних мереж на різних мовах програмування. Тому в архітектурі використано модулі — обгортки (wrappers) для кожної завантаженої SOM.

На рисунку 4.2 відображено архітектуру середовища. На рисунку виокремлено курсивом всі компоненти, які реалізовано в даному проєкті. Реалізації нейронних мереж реалізованих на інших мовах програмування, виокремлено сірим кольором. Середовище реалізовано на мові C++ з

використанням бібліотеки Qt для взаємодії з базами даних та JNI для взаємодії з мовою Java.

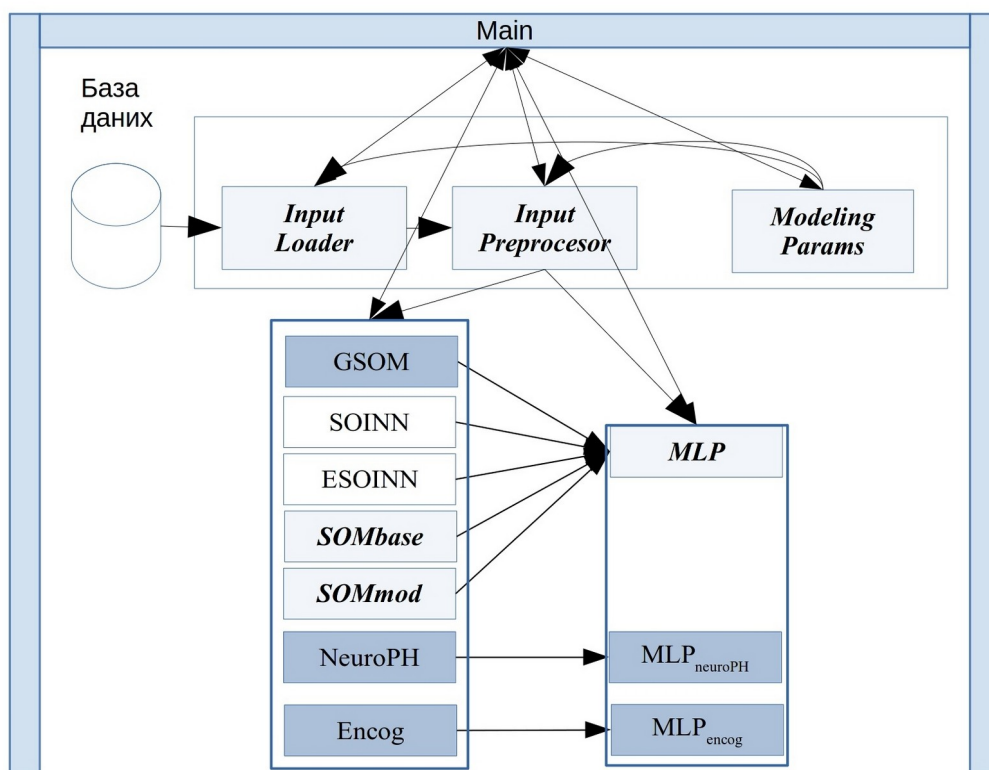


Рисунок 4.2. Архітектура середовища моделювання карт самоорганізації

Середовище реалізовано у вигляді застосунку, який складається з таких основних модулів:

- база даних — містить тестові та навчальні вибірки;
- модуль InputLoader — завантажує навчальну вибірку з бази даних або файлової системи. Модуль після завантаження передає всю вибірку в модуль InputPreprocessor;
- модуль InputPreprocessor — обробляє навчальну вибірку. Модуль дозволяє проводити експерименти з втрат, спотворення сигналу, внесення аномалій в сигнал а також формує постановки задачі класифікації та кластеризації в робочому режимі. При навчанні приклади подаються безпосередньо на обрану реалізацію SOM;

- модуль NNComplex — реалізує інтерфейси над різними реалізаціями нейромереж (SOM та MLP). Для мереж GSOM, NeuroPH, Encog зроблено обгортки для взаємодії додатку з реалізаціями цих мов;

- модуль ModelingParams - містить параметри навчання, такі як швидкість навчання MLP та SOM та інші;

- модуль main — реалізує основний функціонал, який безпосередньо взаємодіє з усіма модулями, проводить навчання та збір статистики.

Кожна з реалізацій SOM має відповідну їй реалізацію MLP, яка використовується автоматично.

4.4. Обчислювальні експерименти

Було проведено обчислювальні експерименти для трьох поставлених задач класифікації:

- за характеристичними ознаками,
- за зображенням,
- за вхідним сигналом, заданим як часовий ряд.

Відповідно було розроблено графічний інтерфейс користувача (Graphical user interface, GUI) для розв'язання класичних наборів даних та моделювання сигналу з втратами.

4.4.1. Графічний інтерфейс користувача для розв'язання задач на класичних наборах даних

Стартовий графічний інтерфейс користувача зображено на рисунку 4.3.

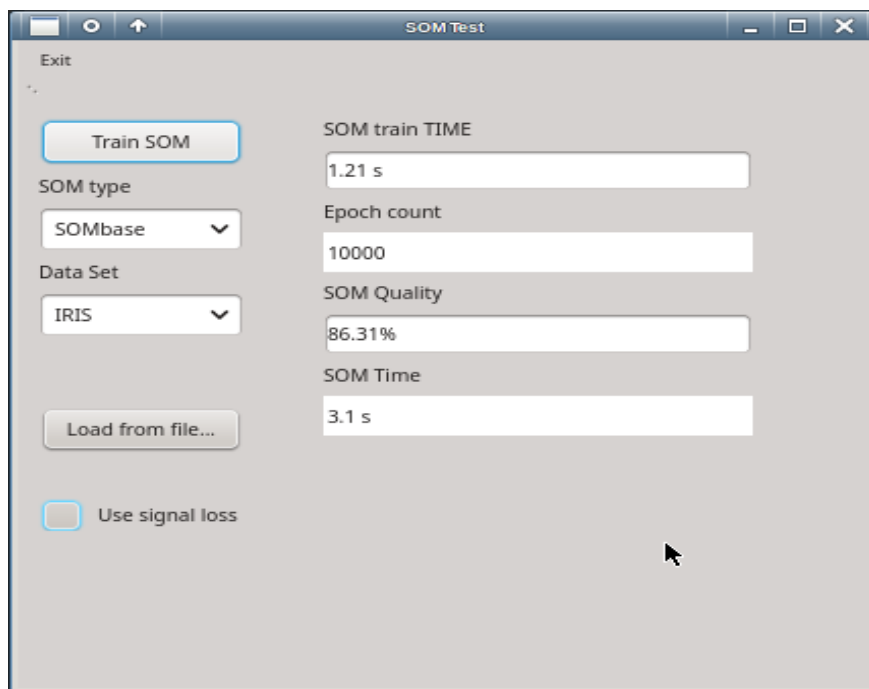


Рисунок 4.3. Графічний інтерфейс користувача

Користувач має змогу вибрати мережу із вище описаних (рисунок 4.4).

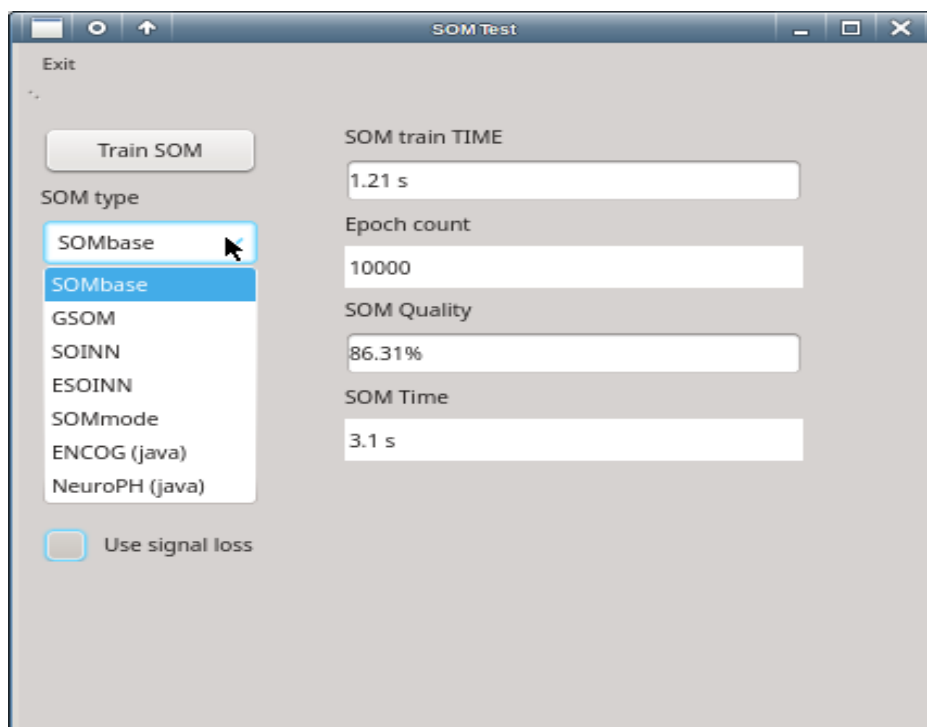


Рисунок 4.4. Варіанти вибору нейронних мереж

Крім цього, користувач може вибрати набір даних для із вже наявних в системі задач, а саме іриси Фішера та MNIST (рисунок 4.5).

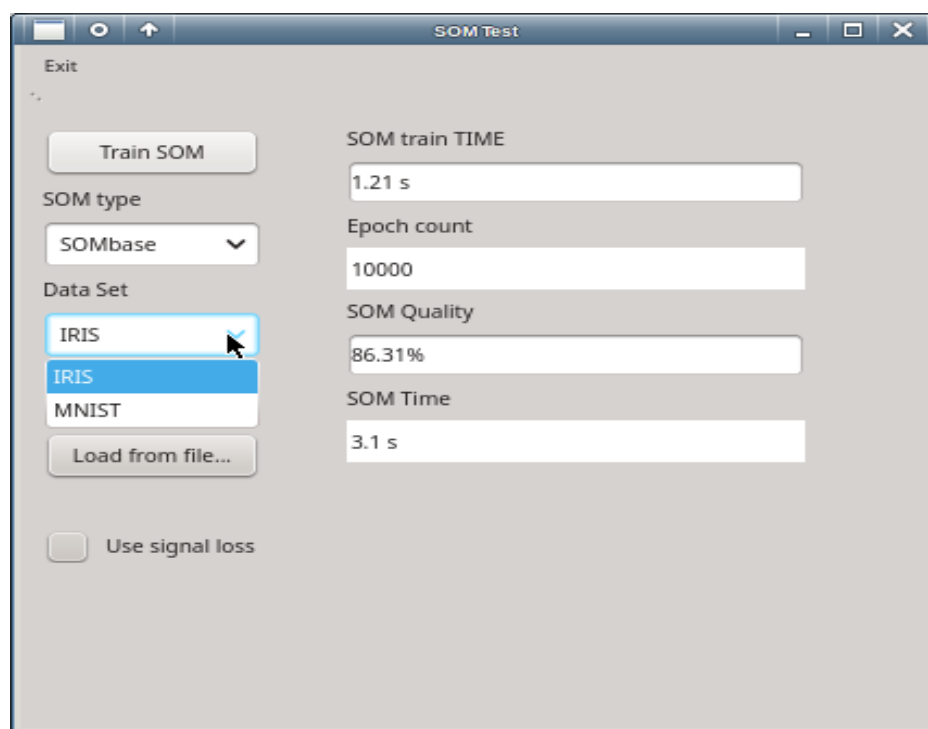


Рисунок 4.5. Варіанти вибору наявних наборів даних

Такий інтерфейс дозволяє завантажувати класичні набори даних для наявних мереж та їх подальшого тестування.

4.4.2. Обчислювальні експерименти на класичних наборах даних

Апробацію розробленого середовища проведено на таких задачах:

- класифікація ірисів Фішера [96];
- розпізнавання рукописних цифр за базою даних MNIST [97].

Ці задачі є класичними тестами для нейронних мереж.

Іриси Фішера — це задача класифікації ірисів трьох видів по чотирьом характеристикам:

- довжина зовнішньої частки оцвітини;
- ширина зовнішньої частки оцвітини;
- довжина внутрішньої частки оцвітини;
- ширина внутрішньої частки оцвітини.

Набір даних з класифікації ірисів містить 150 прикладів

MNIST — база даних рукописних цифр розмірністю 28x28 пікселів.

Цей набір даних містить 70000 прикладів.

В даних обчислювальних експериментах ставилася задача порівняння ефективності реалізації базових та оптимізованих SOM.

Для розв'язання було послідовно навчено всі нейронні мережі за такою послідовністю:

1. Завантаження вибірки з файлу або бази даних;

Завантажується з розподіленням на тренувальну та валідаційну вибірки.

2. Навчання SOM.

SOM навчається з початковим кроком навчання. Через певну кількість епох проводиться зменшення кроку навчання та повторення цього циклу до кінця навчання SOM.

3. Навчання MLP.

Після завершення навчання SOM, відбувається навчання MOP. Після кожної епохи відбувається перевірка точності роботи MLP. У випадку, якщо точність на валідаційній вибірці не зростає 5 епох поспіль, зменшується коефіцієнт швидкості навчання та процес продовжується з новими параметрами. Якщо точність на валідаційній вибірці не зростає 12 епох — навчання MLP завершується.

В таблиці 4.2 зведено параметри навчального комплексу, які застосовувалися у задачах.

Таблиця 4.2. Параметри навчання комплексу

| Параметри навчання комплексу | Набори даних | |
|--|--------------|-------|
| | Іриси Фішера | MNIST |
| Коефіцієнт швидкості навчання SOM | 0.2 | 0.2 |
| Коефіцієнт швидкості навчання MLP | 0.2 | 0.2 |
| Кількість епох до зменшення коефіцієнту швидкості навчання для SOM2 | 50 | 30 |
| Коефіцієнт зменшення швидкості навчання для SOM | 5 | 5 |
| Коефіцієнт зменшення коефіцієнта навчання MLP при виході його на плато | 5 | 5 |

Для оцінювання результатів експериментів було обрано такі критерії:

- частка коректної класифікації;
- час навчання;
- час обробки набору даних.

Останні два критерії визначають ефективність розв'язання задачі.

Результати експериментів зведено до таблиці 4.3.

Таблиця 4.3. Результати розв'язання класичних задач машинного навчання

| Реалізація SOM | GSOM | SOINN | ESOINN | SOMbase | SOMmode | ENCOG | NeuroPH |
|--------------------------------------|------|-------|--------|---------|---------|-------|---------|
| Критерій оцінювання | | | | | | | |
| Класифікація ірисів Фішера | | | | | | | |
| Час навчання SOM (T_{SOM}), сек. | 3,2 | 3,1 | 3,1 | 1,2 | 1,2 | 1,3 | 1,3 |
| Точність класифікації (D), % | 92 | 91 | 92 | 86 | 86 | 86 | 86 |

Продовження таблиці 4.3.

| | | | | | | | |
|--------------------------------------|------|------|------|------|------|------|------|
| Час роботи (T), сек. | 5,5 | 6,1 | 5,4 | 3,1 | 3,1 | 3,3 | 3,3 |
| MNIST | | | | | | | |
| Час навчання SOM (T_{SOM}), сек. | 211 | 197 | 195 | 173 | 169 | 175 | 178 |
| Точність класифікації (D), % | 95.4 | 95.6 | 96.1 | 90.1 | 90.1 | 90.2 | 90.1 |
| Час роботи (T), сек. | 270 | 220 | 210 | 140 | 140 | 152 | 154 |

Оскільки іриси Фішера мають обмежений набір даних, представлення всіх прикладів цього набору здійснювалось 10000 разів.

Результати експериментів за класичними тестовими задачами підтверджують те, що модифіковані мережі, такі як і, ESOINN, GSOM є ефективнішими за реалізацію базового алгоритму функціонування.

4.4.3. Графічний інтерфейс для моделювання втрат вхідного сигналу

Застосунок дозволяє завантажувати власний набір даних. Для цього потрібно натиснути завантаження з файлу та вибрати директорію (рисунок 4.5).

Рисунок 4.5. Процес вибору директорії з набором даних

Директорія повинна містити файли (рисунок 4.6):

- train.txt — вхідні вектори для навчання,
- train_labels.txt — вихідні значення,
- val.txt — валідаційні вхідні вектори,
- val_labels.txt — валідаційні вихідні значення,

- test.txt — опціональні тестові вхідні вектори,
- test_labels.txt — опціональні вихідні значення,
- info.txt — параметри набору даних, в якому розписані розмірності даних, параметри виходу.

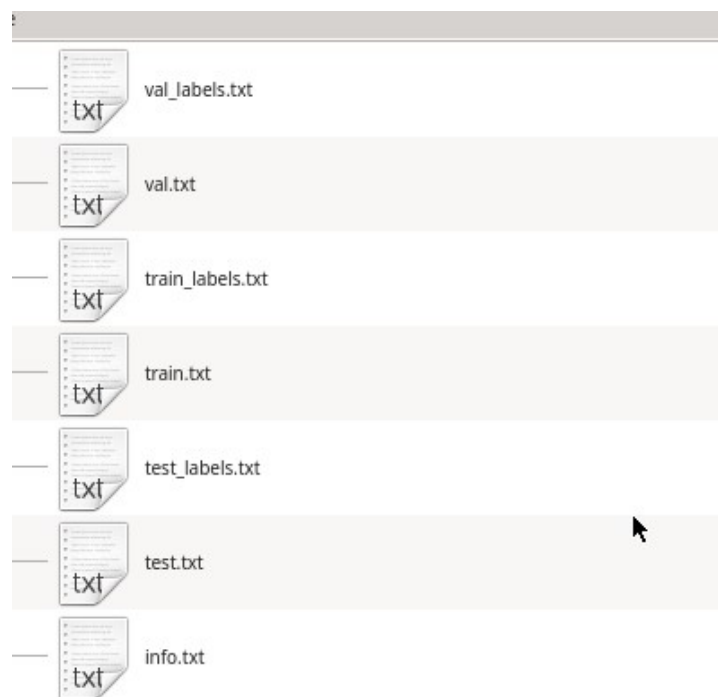


Рисунок 4.6. Необхідні файли для власного набору даних

Для моделювання втрат користувачу потрібно вибрати опцію моделювання втрат. У формі, що з'явиться (рисунок 4.7), користувач зможе ввести наступні дані:

- кількість втрачених значень,
- початковий індекс втрат,
- кінцевий індекс втрат.

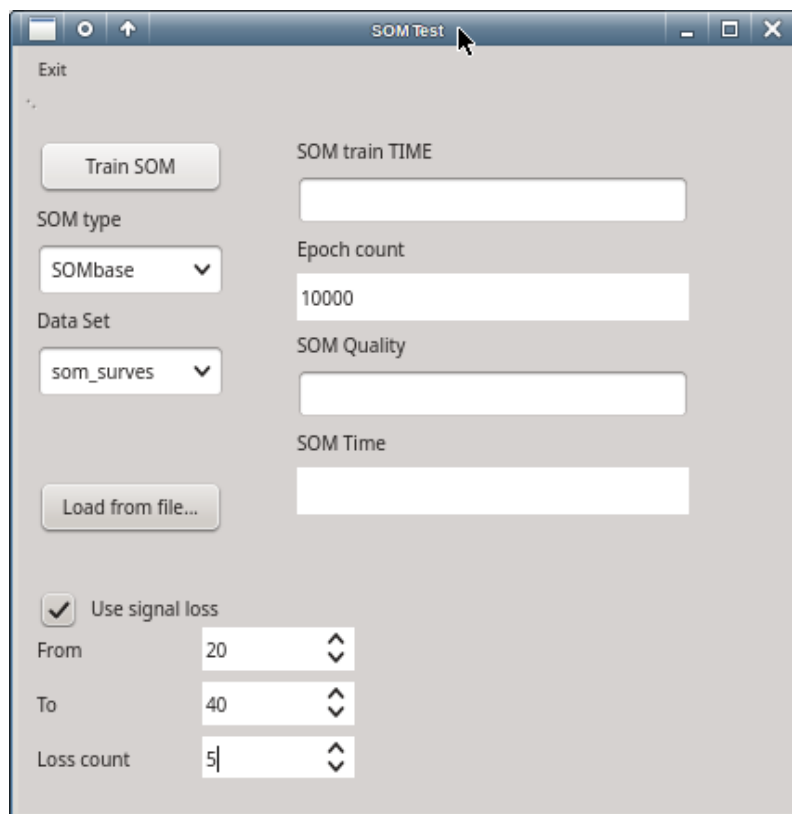


Рисунок 4.7. Параметри моделювання втрат сигналу

4.4.4. Обчислювальні експерименти з моделювання вхідного сигналу з втратами

Було перевірено втрати сигналу для задачі класифікації за часовим рядом [92]. Задача полягає в класифікації типу кривої другого порядку (коло, еліпс, парабола) за векторами, які відображають близькі за значеннями фрагменти кривих.

Було протестовано випадкові втрати на всьому діапазоні значень у випадкові моменти часу. Результати зведено в таблицю 4.4.

Таблиця 4.4. Результати розпізнавання за неповним вектором

| Реалізація SOM | | Частка коректного розізнавання | | | | | | |
|--------------------------------|----|--------------------------------|-------|-----------|-----------|-----------|-------|---------|
| | | GSOM | SOINN | ESOINN | SOMbase | SOMmode | ENCOG | NeuroPH |
| Кількість втрачених значень | 5 | 69 | 71 | 74 | 74 | 74 | 69 | 68 |
| | 10 | 58 | 60 | 56 | 61 | 61 | 54 | 54 |
| | 20 | 39 | 39 | 43 | 69 | 69 | 37 | 37 |

Розроблене програмне середовище дозволяє, по перше, проводити обчислювальні експерименти розв'язання поточної задач на відомих програмних реалізаціях SOM (GSOM, SOINN, ESOINN, NeuroPH, Encog), які поширюються за вільною ліцензією, по друге — доводити ефективність власних розробок карт самоорганізації у порівнянні з апробованими реалізаціями.

Висновки до розділу 4

1. Запропоновано концепцію порівняльного аналізу алгоритмів і відповідних реалізацій карт самоорганізації Кохонена для доведення коректності нових модифікованих алгоритмів SOM.
2. Представлено програмну реалізацію середовища моделювання нейронних мереж для розв'язання задачі кластеризації.
3. Наведено приклад обчислювальних експериментів з доведенням ефективності відомих реалізацій за базовим та модифікованими алгоритмами.

За результатами досліджень опубліковано стаття [98] та тези доповіді на міжнародній конференції [99].

РОЗДІЛ 5

КЛАСИФІКАЦІЯ ТА СЕГМЕНТАЦІЯ НА ЗГОРТКОВИХ МЕРЕЖАХ З ВИКОРИСТАННЯМ КАРТ САМООРГАНІЗАЦІЇ

Метою досліджень, представлених в розділі, є інтеграція згорткової нейронної мережі з картою самоорганізацією Кохонена в єдиному нейромережевому комплексі для підвищення точності. Для цього вирішено такі задачі:

- розробити концепцію інтеграції SOM в мережі глибокого навчання;
- розробити архітектуру гібридної згорткової мережі з інтегрованою картою самоорганізації Кохонена;
- провести порівняльний аналіз результатів розв'язання тестових задач на згортковій нейронній мережі та мережі, модифікованій заданим способом.

Класифікація об'єктів і процесів є розповсюдженою прикладною задачею. Її розв'язання, насамперед, потрібно в системах автоматичного діагностування, наприклад, для визначення стану роботи обладнання за діагностичним сигналом або виявлення патологій за медичними знімками. З розвитком згорткових нейронних мереж відкрились нові перспективи розв'язання таких задач. Однак не для всіх випадків діагностування достатньо точності класифікації, яку можна досягти на цих мережах. Це стосується, наприклад, своєчасного діагностування початку перехідних процесів. Водночас інший різновид нейронної мережі – карти самоорганізації Кохонена – мають концептуальну властивість для навчання за невизначеною множиною класів, тобто дають можливість розв'язувати таку задачу. Тому підвищення точності класифікації на основі вбудування мереж Кохонена в

архітектури згорткових мереж є актуальною задачею і має практичну значущість.

В розділі проведено аналіз засобів підвищення точності згорткових нейронних мереж та розв'язання проблем, які при цьому виникають. Також наведені способи підвищення частки коректної кластеризації на мережах Кохонена за рахунок 'зростання' її решітки при визначенні нових класів в процесі навчання. Саме ця властивість дає можливість розпізнавання перехідних процесів. Визначено, що існуючі рішення поєднання мереж Кохонена та згорткових мереж спрямовані на підвищення ефективності тільки карт самоорганізації, тому для підвищення точності класифікації та семантичної сегментації згортковими мережами виникла необхідність розробки нової архітектури. В розділі наведено її опис. Оскільки вихідною інформацією мереж Кохонена є матриця вагових значень нейронів решітки, для обробки діагностичних знімків необхідно було пов'язати її з представленням зображень. В роботі запропоновано концепцію вбудованого блоку асоціативного масиву на основі мереж Кохонена.

За запропонованим способом розроблено програмну реалізацію гібридної нейронної мережі. Наведено постановку та результати обчислювальних експериментів. Експериментально доведено ефективність запропонованого способу.

5.1. Проблема інтеграції згорткових нейронних мереж з картами самоорганізації Кохонена

Класифікація на основі нейромережових методів була однією з перших задач штучного інтелекту. З розвитком мереж глибокого навчання відкрилися нові перспективи для класифікації процесів за діагностичними сигналами та знімками. Такі задачі властиві, наприклад, технічному, медичному, сейсмічному діагностуванню. При їх розв'язанні надважливою є точність. Проблема полягає в тому, що на згорткових нейронних мережах необхідна

точність класифікації може бути досягнута не для всіх задач діагностування. Наприклад, такі мережі не завжди здатні виявляти перехідні стадії процесів, що діагностуються. Для розв'язання таких задач існує інший різновид нейронних мереж — карти самоорганізації Кохонена (Self organizing map – SOM). Однак SOM призначені для розв'язання задачі кластеризації і потребують використання додаткових засобів для подальшої класифікації поточного прикладу. Використання SOM як блоку згорткової мережі розширює клас задач технічної діагностики і підвищує точність класифікації. Тому підвищення точності класифікації за рахунок використання карт самоорганізації Кохонена як частини згорткової мережі є актуальною задачею та має практичну значущість.

Мережі глибокого навчання насамперед призначені для задач класифікації, детекції, сегментації. Сучасні дослідження в галузі глибокого навчання спрямовані на підвищення точності та/або скорочення часу роботи.

Результати сучасних досліджень з машинного навчання прийнято апробувати на спеціальних конкурсах, найвідомішим з яких є ImageNet — змагання з класифікації об'єктів на зображеннях. Перший суттєвий результат з підвищення точності класифікації було отримано мережею AlexNet [24], яка виграла конкурс ImageNet LSVRC-2012. Ця мережа була модифікацією LeNet [20] і мала декілька основних особливостей:

- функцію активації ReLU, яка дозволило збільшити швидкість навчання завдяки зменшенню затухання градієнту;
- випадкове виключення групи зв'язків між нейронами (DropOut), що частково вирішило проблему перенавчання.

Конкурс ImageNet LSVRC-2014 виграла мережа VGG [27], яка була модифікацією AlexNet. В мережі VGG згорткові фільтри високої розмірності були замінені на серію фільтрів 3*3. При такому підході рецептивне поле окремого нейрона не зменшується, проте зменшується кількість параметрів мережі, завдяки чому мережу легше тренувати.

В 2015 році була представлена мережа Inception (GoogleNet) від компанії Google [32]. При реалізації мережі була розвинена ідея, закладена в VGG — заміна фільтрів $n \times n$ послідовністю фільтрів $n \times 1$ та $1 \times n$. Також на основі Inception були розроблені мережі Inception V2 та Inception V3.

Втім зазначеним мережам була властива проблема деградації (degradation problem). Проблема полягає в тому, що підвищення точності класифікації можна досягти додаванням нових шарів, але це відбувається до деякого порогу, після чого точність починає зменшуватися. Основною причиною цього є затухання градієнту (vanishing gradient). На останніх шарах мережі градієнт має суттєве значення, а на перших — неприпустимо мале. Проблема деградації була вирішена командою Microsoft в мережі ResNet [30] за допомогою нових зв'язків, що забезпечують додавання вхідного сигналу до результату декількох шарів згорток (shortcut—з'єднання).

Одним з останніх засобів підвищення точності стали механізми уваги (attention) [37, 38].

Однак всі методи методи підвищення точності згорткових мереж не завжди надають задану точність в задачах діагностування у випадках, коли неможливо ввести жорстку класифікацію процесу завдяки виникненню перехідних стадій, які, в свою чергу, мають розпізнаватися. З іншої сторони, карта самоорганізації Кохонена є різновидом нейронних мереж, спеціально призначених для розв'язання задачі кластеризації, що дозволяє проводити навчання за заздалегідь не визначеними класами. В основу даної роботи було закладено припущення, що така властивість SOM дозволить вдосконалити згорткові нейронні мережі. Властивості SOM в поєднанні з мережами глибокого навчання мають дозволити точніше визначати перехідні процеси за діагностичними сигналами.

Суттєвими перевагами SOM є можливість навчання на малій кількості прикладів навчальної вибірки, швидке навчання та стійкість до зашумлених даних. Однак мережа SOM чутлива до початкової ініціалізації вагів, отже результат тестування може бути нестабільним при проведенні серії однакових

експериментів. Дослідження, представлені в роботах [48-50], вирішують цю проблему і підвищують частку коректної роботи в робочих режимах. В цих роботах запропоновано підхід, спрямований на створення структури мережі SOM, яка змінюється в процесі роботи. Базова ідея полягає в розширенні решітки SOM, якщо новий приклад навчальної вибірки суттєво відрізняється від ваг нейронної мережі. Тобто мережа може ‘зростати’ в процесі навчання.

Оскільки визначення класу поточного об’єкту здійснюється за матрицею виходів нейронів SOM, подальша класифікація здійснюється або на додаткових нейронних мережах або за візуалізацією навченої карти з виокремленим поточним прикладом. Традиційно в якості додаткової мережі в комплексі з SOM використовують багат шаровий перцепрон. В сучасних дослідженнях [100-102], SOM було поєднано зі згортковими мережами. Метою цих досліджень було покращення кластеризації SOM. В роботі [100], представлено два види SOM, які відрізняються застосуванням. В першому випадку — замість згорткового шару, в іншому — замість пулінгу.

В [101] концепція полягає в навчанні декількох SOM, кожна з яких відповідає окремій області вхідного зображення. В роботі [102] приховані шари заміняються модифікованими картами самоорганізації. Однак всі ці дослідження були спрямовані на підвищення ефективності саме карт самоорганізації Кохонена. Тому необхідно провести дослідження за концепцією підвищення точності класифікації згорткових нейронних мереж за рахунок додавання SOM.

5.2. Базова концепція SOM

Концепція SOM полягає в тому, що після навчання ваги розподіляються таким чином, що на решітці нейронів виокремлюються кластери. В ідеальному випадку, кожний кластер відповідає певному класу об’єктів, який розпізнається. Класифікація поточного об’єкту відбувається за

характеристичними ознаками і полягає в визначені нейрона-переможця, ваги якого найкраще повторюють сигнал.

Кожний нейрон в SOM має вхід, розмірність якого рівна розмірності вхідного вектору. Тобто вхідний вектор подається на кожний нейрон. При цьому кожне i -е значення вхідного сигналу пов'язане з i -м значенням вектором-ваг (рисунок 5.1).

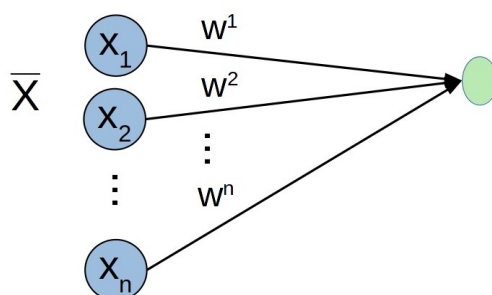


Рисунок 5.1. Представлення вхідного вектору нейроном SOM

Кожний нейрон отримує сигнал зі всього вхідного вектору. Нейроном-переможцем в мережі Кохонена може бути тільки один нейрон. Нейрон-переможець обирається по найменшому значенню дискримінантної функції серед всіх нейронів. Найчастіше як метрику використовують Евклідову відстань:

$$D = |\bar{X}_{som} - \bar{W}_j|^2, \quad (5.1)$$

де \bar{X}_{som} – поточний вхідний вектор SOM;

\bar{W}_{som} – вектор ваг j -го нейрона.

Схему карти самоорганізації зображено на рисунку 5.2.

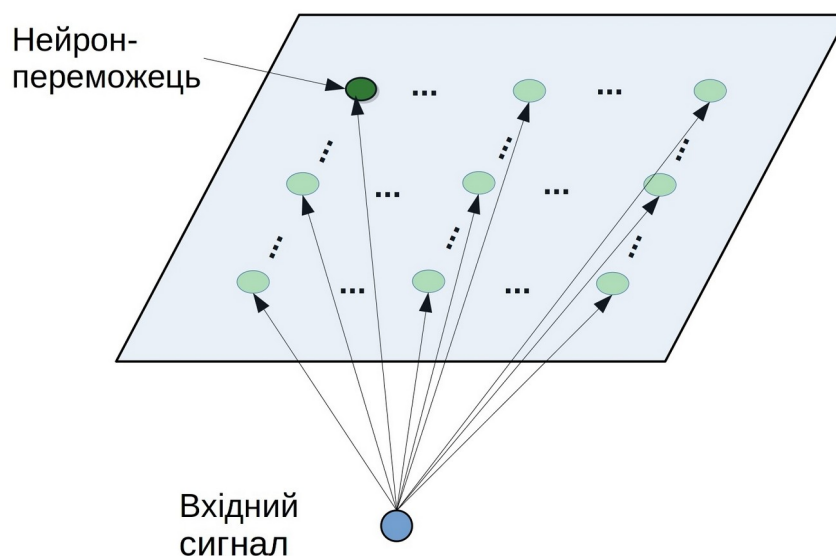


Рисунок 5.2. Схема сприйняття 1D-сигналу картою самоорганізації

Навчання карти самоорганізації складається з таких етапів [45, 103]:

1. Початкова ініціалізація.

Для всіх синаптичних ваг w_j встановлюється у випадкові величини.

2. Маркування нейронів.

Для кращого навчання карти самоорганізації у відповідних місцях покриття доцільно примусово встановити центри кластерів [81], якщо заздалегідь відома їх кількість, наприклад, розв'язується задача класифікації.

3. Конкуренція нейронів.

Із множини вхідних векторів довільно обирається вектор X та подається на SOM.

4. Пошук максимальної подібності.

Визначається нейрон-переможець на основі критерія мінімуму відстані Евкліда (5.1).

5. Генерація нового кластеру.

Новий кластер генерується, якщо вхідний вектор не належить до жодного з відомих класів. Це відповідає випадку, коли

$$d(x, w_j) > \Delta, \quad (5.2)$$

де Δ - поріг, при якому буде створено новий кластер,

$d(x, w_j)$ — відстань між векторами (5.1).

6. Процес кооперації.

Визначається топологічна околиця $h_{j,i}$, яка часто береться як функція Гауса:

$$h_{j,i} = \exp\left(\frac{-d_{j,i}^2}{2\sigma^2}\right) \quad (5.3)$$

де σ — ефективна ширина. Показує скільки нейронів буде кооперуватися.

d — відстань між нейронами.

7. Синаптична адаптація.

Кооперуються вектори синаптичних ваг всіх нейронів, що перебувають у околиці

$$w_j(n+1) = w_j(n) + \eta(n) h_{j,i}(n) (x - w_j(n)), \quad (5.4)$$

де $w_j(n+1)$; $w_j(n)$ — вектори синаптичних ваг нейрона на моменти часу відповідно n та $(n+1)$; i

$\eta(n)$ — параметр швидкості навчання.

8. Навчання закінчується, якщо зміни вагів мережі не перебільшують задане значення або досягнута максимальна кількість кроків. В протилежному випадку — повернення до етапу 2.

Проте в класичному режимі карта самоорганізації не може бути застосована в згорткових нейронних мережах. Проблема полягає в тому, що результатом роботи SOM в робочому режимі є матриця різниць відстаней між вхідним вектором і вектором ваг нейронів, за якими обирається нейрон-переможець. Цей результат необхідно співставити з реальним відображенням вхідного сигналу, що має подаватися далі на згорткові шари.

5.3. Реалізація блоку SOM в мережі глибокого навчання

Було запропоновано реалізувати відповідність нейронів-переможців конкретним фрагментам вхідного сигналу. Для цього мережа SOM поєднується з асоціативним масивом. Цей масив реалізовано як відображення решітки SOM у вигляді матриці. Елементами такого масиву є фрагменти вхідного сигналу SOM. Вхідними даними даного блоку є карта признаков матриці $w \times h$, де w, h — ширина та висота. В блоці SOM реалізована пряма відповідність між нейроном та елементом матриці.

На рисунку 5.3 схематично представлено фрагмент двомірної решітки SOM з відповідним відображенням асоціативного масиву для тесту з розпізнавання рукописних цифр.

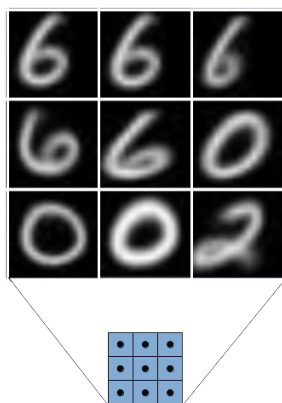


Рисунок 5.3. Асоціація зображень на фрагменті SOM у вхідному блоці

Решітка SOM може мати різну розмірність:

- для вхідного сигналу у вигляді послідовності значень (1D сигнал) — двомірна решітка;
- для вхідного сигналу — зображень (2D сигнал) — трьохмірна решітка.

Асоційований масив має таку саму розмірність, як і решітка SOM. Індекси нейрона в SOM співпадають з індексами комірки в масиві. Однак при такій системі відповідності мережа вчиться не універсально, а лише на прикладах навчальної вибірки і в подальшому не здатна до узагальнення.

Тобто мережа може перенавчається (*overfitting*). Тому запропоновано, щоб в робочому режимі блок SOM повертав сигнал, який отримано шляхом узагальнення всіх асоційованих сигналів нейрона-переможця та його околу. Вихідний сигнал блоку SOM обчислюється за такою формулою:

$$0.5 * G_{winner} + \frac{0.5 * 1}{n} \sum_{k=0}^n G_k \quad (5.5)$$

де I_{winner} — сигнал, асоційований з нейроном переможцем,
 I_n — асоційований сигнал нейрона-сусіда,
 n — кількість сусідів.

В процесі тренування SOM обирається нейрон-переможець. Після чого сигналу, асоційованому з ним, співставляється отриманий з попереднього блоку сигнал. Також оновлюються фрагменти сигналу подібно до того, як оновлюються ваги в нейронній мережі Кохонена:

$$W_{new} = (1 - t) * W_{old} + t * W, \quad (5.6)$$

де I — сигнал, що поступає на вхід,
 I_{old} — сигнал — асоційований з даною коміркою,
 t — функція, що затухає по мірі віддалення від нейрона-переможця.

Запропонований блок SOM додається до базових блоків CNN.

5.4. Архітектура гібридної мережі для розв'язання задачі класифікації

Згорткова мережа містить такі базові шари:

- шар згортки — (*Conv*) базовий компонент згорткових нейронних мереж,
- шар пулінгу — (*Pool*) для зниження розмірності вхідного сигналу,
- шар активації — (*Activation*) для внесення нелінійності в мережу,

- повноз'єднані шари (*Dense*) — для безпосередньо класифікації — зазвичай використовуються в останніх шарах мережі.

Ці шари описано в розділі 2 дисертації.

Крім цього використовують допоміжні шари, наприклад, шари нормалізації (*Norm*), шари виключення зв'язків, глобального пулінгу (Global pooling), шари об'єднання (*Concatenate*) та інші.

В даній роботі до традиційних шарів було додано блок SOM. На рисунку 5.4 представлено 3 різновиди архітектури блоку згорткової мережі: для традиційного випадку (5.4.a) та двох запропонованих архітектур з використанням блоку SOM.

результатів з подальшим поданням на наступні шари. Блок SOM з архітектури (5.4.б), (5.4.в) застосовується починаючи з другого блоку від вхідного шару мережі SOM. Це необхідно для того, щоб на першу мережу SOM було подано сигнал після шару згортки. Інакше вхідний сигнал майже без змін буде проходити по всій мережі. Таким чином, в першому (вхідному) блоці CNN шар SOM не застосовується.

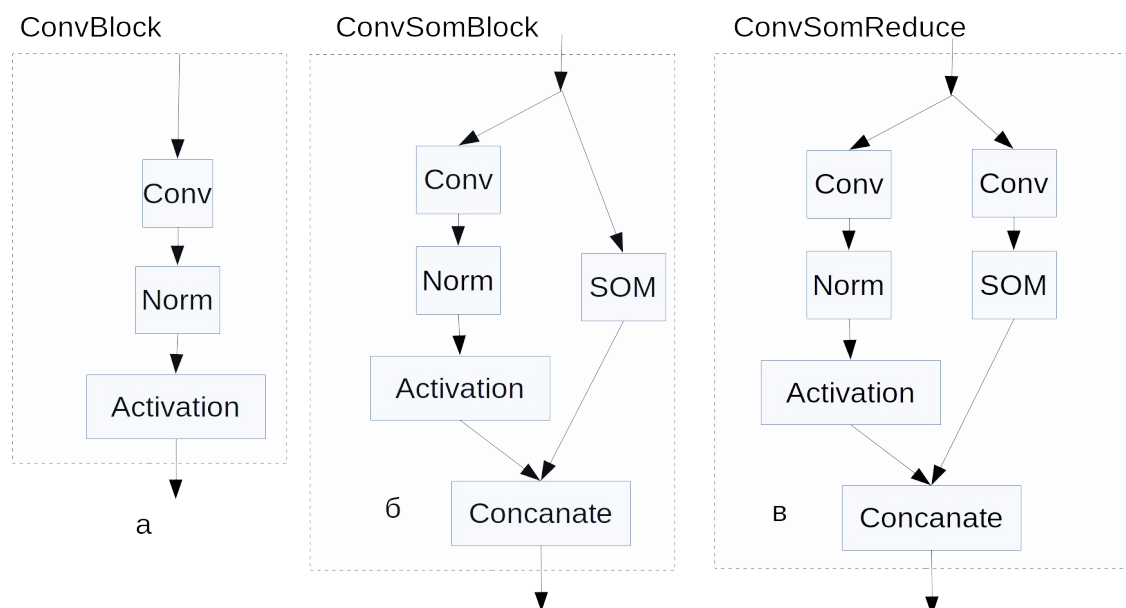


Рисунок 5.4. Базовий блок згорткової мережі: а) наразі існуючий, б) з використанням блоку SOM, в) з використанням блоку SOM з попереднім пониженням розмірності

Визначення оптимальної кількості шарів є окремою проблемою. Для її уникнення часто використовують апробовані архітектури, такі як ResNet34, ResNet50, VGG16 та інші.

Оскільки в прикладних задачах вхідні сигнали – наприклад, графічні зображення – зашумлені, потрібно зменшити їх вплив на роботу SOM. Для цього використано батч-нормалізацію (batch normalization). Таким чином, шари SOM будуть менш чутливі до шуму та зміни амплітуди вхідного сигналу. У випадку, якщо батч-нормалізація використовується в блоці згортки, використання нормалізації перед шаром SOM не потрібне.

В концепції згорткових мереж знайшло широке застосування пониження розмірності сигналу з одночасним зростанням кількості каналів. Шар SOM має вживатися для кожного каналу незалежно. Оскільки нейронна мережа на останніх шарах часто має багато каналів, для економії обчислювальних ресурсів та пам'яті на нижніх шарах перед блоком SOM використано згортку розмірністю 1×1 , яка понижує кількість каналів (рис. 5.4.в). Блок архітектури з рисунку 5.4.б доцільно використовувати, якщо кількість каналів менше 8. В іншому випадку, доцільно використовувати блок з архітектури 5.4.в.

Проте запропоноване рішення погано себе показує при розпізнаванні зображень великих розмірностей або при розв'язанні задач сегментації. Для вирішення цих задач архітектурне рішення потребує модифікації.

5.5. Архітектура гібридної мережі для розв'язання задач сегментації

В розділі 5.4 запропоновано архітектуру, ефективну для розв'язання задач класифікації на зображеннях невеликих розмірів. Однак для ефективного вирішення задачі сегментації, цю архітектуру необхідно модифікувати. Модифікація стосується підключення блоку SOM до мережі

CNN. В попередньому випадку карта самоорганізації реагувала на повний вхідний вектор, тому за наявності вхідного сигналу малої розмірності результат був задовільний. Однак зі зростанням розмірів, що насамперед стосується зображень, ефективність мережі не задовольняє поставленим вимогам.

При розв'язуванні задач сегментації в роботі запропоновано, на блок SOM подавати не всі значення вхідного сигналу, а значення його зрізів: відрізків для 1D-сигналів та фреймів — для 2D-зображень. Для цього кожен вхідний сигнал розбивається на фрагменти однакових розмірів. У випадку, якщо в залишку опинився фрагмент, який має розміри, менший розмірів вікна, він масштабується до стандартних розмірів. Визначення оптимальних для поточної задачі розмірів зрізу (кількості пікселів) здійснюється експериментально.

Таким чином, карта самоорганізації застосовується не за повним сигналом, а за його зрізом. На рисунку 5.5 наведено приклад розбиття зображення (2D-сигналу) на фрейми, розмірністю 128x128 пікселів.

Для того, щоб нейронна мережа була інваріантною (незалежною) до зсувів зображення, після зрізу в поточному шарі застосовується лише єдиний блок SOM. Однак цей блок SOM застосовується послідовно до кожного з k зрізів.

Для реалізації такого підключення в структуру блоку додається шар зрізу, входом якого є повний сигнал, виходом — серія фрагментів сигналу (рисунок 5.6).

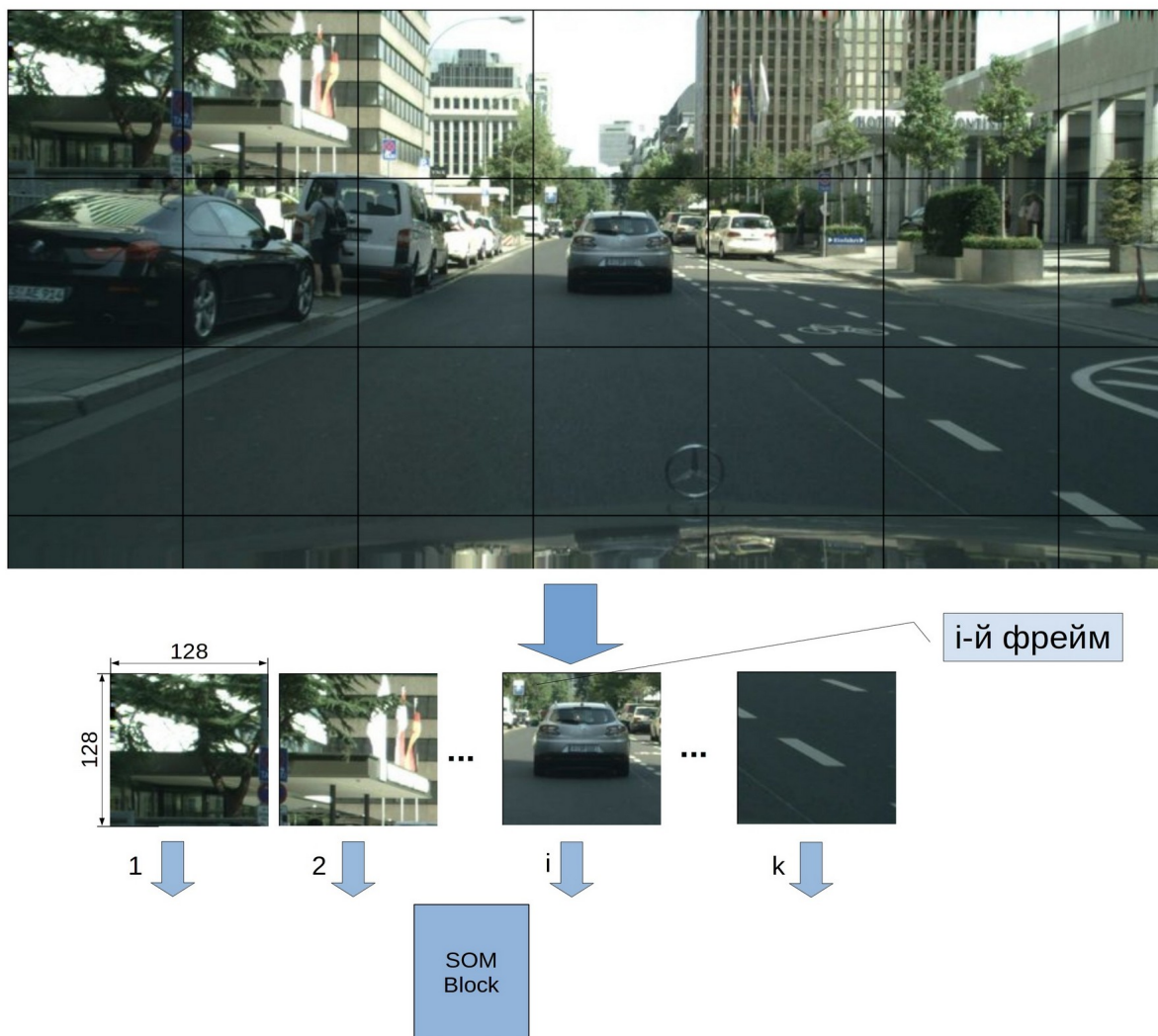


Рисунок 5.5. Схема представлення зрізів на SOM

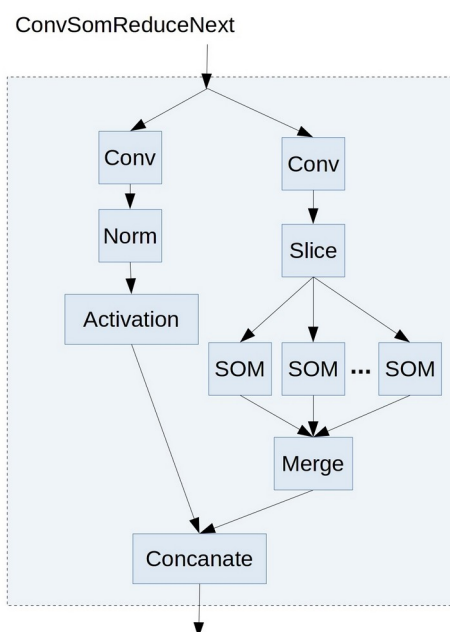


Рисунок 5.6. Структура блоків з використанням SOM з використанням зрізу

В архітектуру U-Net запропонований блок вбудовано в модифікований блок декодера (рисунок 5.7).

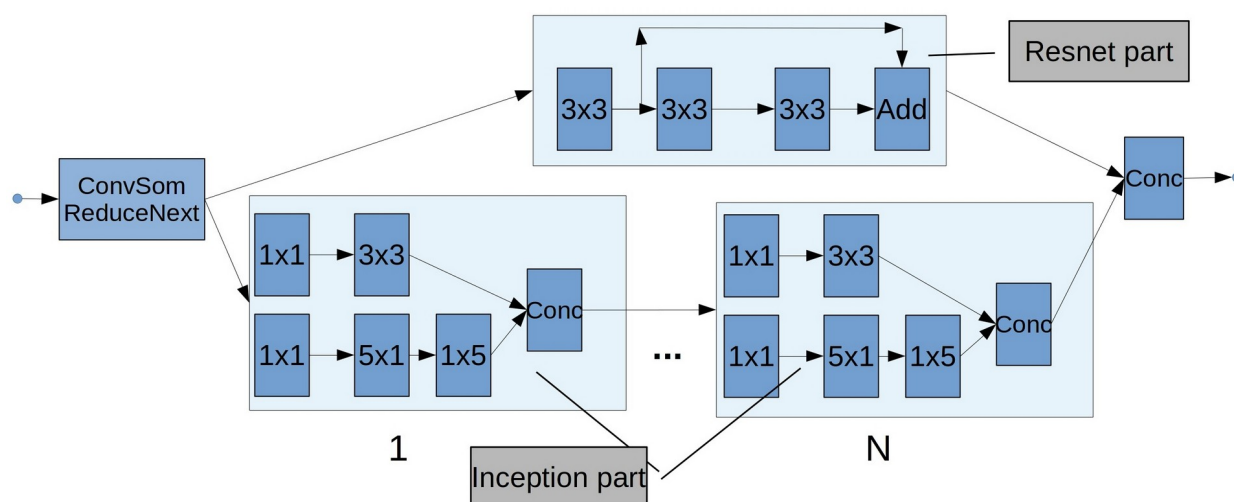


Рисунок 5.7. Запропонований блок декодера мережі U-Net

Завдяки запропонованому в розділі 3 дисертації способу співставлення нейронів вхідного шару вхідному вектору SOM та запропонованій інтеграції згорткової мережі з блоком SOM, можна досягти значного підвищення точності семантичної сегментації.

5.6. Обчислювальні експерименти для розв'язання задачі класифікації та сегментації

5.6.1. Тестові задачі

Для експериментального підтвердження ефективності запропонованого способу вбудови SOM в CNN розв'язано такі тестові задачі:

- класифікація за діагностичним сигналом, який задається як часовий ряд значень характеристичних показників,
- розпізнавання рукописних цифр на наборі даних MNIST.

Для реалізації першої задачі обрано приклад з розпізнавання кривих другого порядку за подібними фрагментами — верхніми половинами кола, еліпса, параболи. Задачу наведено в роботі [92]. Характеристичний сигнал

обчислюється рівняннями кривих другого порядку: кола, еліпса, параболи. До кожного дискретного значення випадково генерується шумове викривлення в заданому діапазоні. Таким чином вхідним сигналом є перелік дискретних значень функції. Набір даних складається з навчальної та валідаційної вибірок, кожна з яких містить по 999 оригінальних прикладів: по 333 приклади для кожної кривої. Довжина вектору вхідного сигналу — 100.

Тестова задача розпізнавання розв'язана на наборі даних (dataset) MNIST — бази даних рукописних цифр. Набір даних також складається з навчальної та валідаційної вибірок. Навчальна містить 60000 одноканальних (чорно-білих) зображень розмірністю 28×28 пікселів, валідаційна — 10000. Задача полягає в класифікації цифри на поточному зображенні.

Для задачі сегментації використано 2 набори даних, описаних в розділі 2:

1. Salt identification Challenge — задача ідентифікації сольових відкладень в надрах землі.
2. Cityscapes — розпізнавання вуличних сцен заданих об'єктів із стандартних класів.

5.6.2. Обчислювальні експерименти розв'язання задачі класифікації

Для обчислювальних експериментів побудовано 2 нейронні мережі. В першій використано традиційні згорткові блоки ConvBlock (рисунок 5.8).

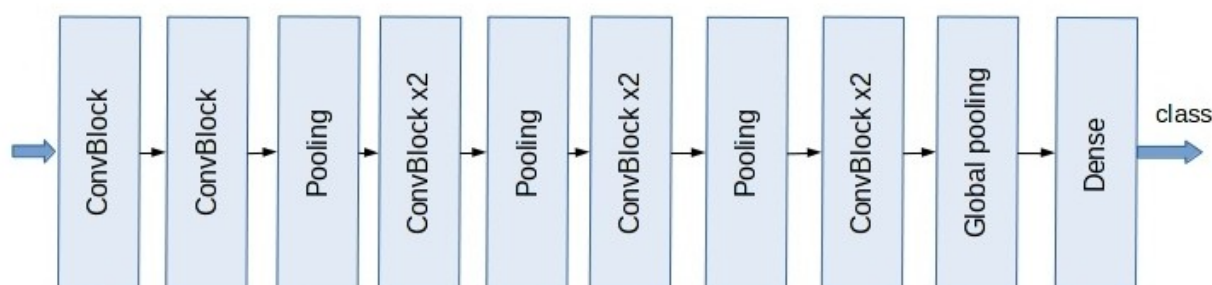


Рисунок 5.8. Архітектура базової мережі з застосуванням традиційних блоків згортки

В другій мережі запропоновані згортки з використанням SOM — ConvSomBlock і ConvSomReduce (рисунок 5.9).

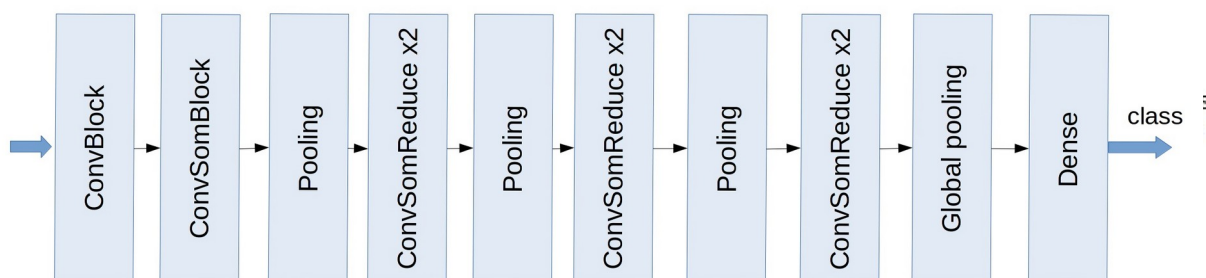


Рисунок 5.9. Архітектура гібридної мережі з застосуванням SOM-блоків

Для тестових нейронних мереж обрано прості моделі.

Для коректності постановки експериментів обидві мережі, і традиційна і модифікована, мають однакову архітектуру, відповідно, для яких проведено такі серії обчислювальних експериментів:

- на *базовій* мережі;
- на *гібридній* мережі.

Позначення блоків та шарів співпадають з описами при, наданими при визначенні архітектури гібридної мережі.

Для розв'язання першої задачі всі згортки були одномірні (1D), а матриці SOM двомірні (2D), в другій задачі відповідно — 2D та 3D. В експериментах максимальна кількість каналів SOM була 4, 8, 16 та 32.

Метрикою точності в усіх експериментах було відношення кількості правильно визначених класів до загальної кількості прикладів.

Результати експериментів наведено в таблиці 5.1.

Таблиця 5.1. Результати обчислювальних експериментів на гібридній згортковій мережі з блоком SOM

| Мережа | Кількість каналів SOM | Метрика точності |
|---------------------------------------|-----------------------|------------------|
| Задача розпізнавання за часовим рядом | | |
| Базова мережа | - | 0.9629 |
| Гібридна мережа | 4 | 0.9709 |
| | 8 | 0.9719 |
| | 16 | 0.9709 |
| | 32 | 0.9709 |
| Задача розпізнавання зображень | | |
| Базова мережа | - | 0.9937 |
| Гібридна мережа | 4 | 0.9945 |
| | 8 | 0.9949 |
| | 16 | 0.9948 |
| | 32 | 0.9945 |

Результати експерименту довели, що запропонований спосіб модифікації згорткової мережі підвищує точність класифікації для задачі розпізнавання сигналу за часовим рядом в межах з 0.9629 до 0.9719; для задачі розпізнавання зображень — з 0.9937 до 0.9949.

5.6.3 Обчислювальні експерименти розв'язання задачі сегментації

В ході експериментів були протестовані базові та модифіковані запропонованим способом мережа U-Net. В якості backbone-мережі обрано

ResNet34. Навчання проводилося на комп'ютері з відеокартою Nvidia GTX 1080TI. Розмірність батчу при тренуванні була 64.

Для підвищення якості навчання було застосовано аугментацію. Аугментацію реалізовано як функцію спотворення, яка одночасно застосовувалася до вхідного зображення і до маски. З аугментацій при навчанні застосовувалися лише поворот відносно вертикальної осі (horizontal flip) з ймовірністю 0.5. Аугментація під час тесту (test time augmentation – ТТА) при передбаченні нейронною мережею не застосовувалася.

Проведено такі серії обчислювальних експериментів:

- на *базовій* мережі U-Net;
- на *модифікованій* U-Net, вдосконаленою трьома способами оптимізацій, запропонованими в розділі 2 дисертації,
- на *гібридній* мережі U-Net вдосконаленою трьома способами оптимізацій, запропонованими в розділі 2 дисертації з додатковою модифікацією блоків декодера з використанням SOM.

Для останньої серії обчислювальних експериментів введено нові параметри моделювання — кількість каналів SOM та розмір зрізу зображення. Ці параметри відповідають запропонованому способу інтеграції SOM в згорткову нейронну мережу та потребують моделювання для експериментального визначення найоптимальніших співвідношень.

Результати обчислювальних експериментів на задачі Salt identification Challenge зведено в таблицю 5.2.

Таблиця 5.2. Результати розв'язання задачі Salt identification Challenge

| Мережа | Кількість каналів SOM | Розмір зрізу зображення | Метрика IOU |
|-----------------------|-----------------------|-------------------------|----------------|
| Базова мережа U-Net | - | - | 0.80672 |
| Модифікована U-Net | - | - | 0.81469 |
| Гібридна мережа U-Net | 16 | 4x4 | 0.81474 |
| | | 8x8 | 0.81486 |
| | | 16x16 | 0.81477 |
| | | 32x32 | 0.81442 |
| | | 64x64 | 0.81338 |
| | 32 | 4x4 | 0.81489 |
| | | 8x8 | 0.81512 |
| | | 16x16 | 0.81503 |
| | | 32x32 | 0.81424 |
| | | 64x64 | 0.81367 |
| | 64 | 4x4 | 0.81526 |
| | | 8x8 | 0.81534 |
| | | 16x16 | 0.81513 |
| | | 32x32 | 0.81449 |
| | | 64x64 | 0.81373 |

Результати обчислювальних експериментів на задачі Cityscapes зведено в таблицю 5.2.

Таблиця 5.3. Результати розв'язання задачі Cityscapes

| Мережа | Кількість каналів SOM | Розмір зрізу зображення | Метрика IOU |
|-----------------------|-----------------------|-------------------------|----------------|
| Базова мережа U-Net | - | - | 0.42502 |
| Модифікована U-Net | - | - | 0.43917 |
| Гібридна мережа U-Net | 16 | 4x4 | 0.43934 |
| | | 8x8 | 0.43946 |
| | | 16x16 | 0.43942 |
| | | 32x32 | 0.43931 |
| | | 64x64 | 0.43919 |
| | 32 | 4x4 | 0.43957 |
| | | 8x8 | 0.43962 |
| | | 16x16 | 0.43958 |
| | | 32x32 | 0.43941 |
| | | 64x64 | 0.43931 |
| | 64 | 4x4 | 0.43962 |
| | | 8x8 | 0.43984 |
| | | 16x16 | 0.43978 |
| | | 32x32 | 0.43963 |
| | | 64x64 | 0.43954 |

Експериментально визначено, що оптимальна ширина зрізу для наведених задач рівна 8x8 пікселів.

Таким чином, експериментально доведено, що інтеграція SOM в мережі глибокого навчання U-Net архітектури підвищує точність розв'язання:

- з 0.80672 до 0.81534 для мереж для задачі Salt identification Challenge;

- з 0.42502 до 0.43984 для мереж для задачі Cityscapes.

Висновки до розділу 5

1. Запропоновано спосіб асоціації відображення решітки самоорганізації з фрагментами вхідного сигналу для вбудови блоку SOM в гібридну згорткову нейронну мережу.

2. Запропоновано архітектурне рішення інтеграції згорткової мережі з блоком SOM на основі асоціації відображення решітки самоорганізації з фрагментами вхідного сигналу для підвищення точності класифікації та семантичної сегментації.

3. Проведено обчислювальні експерименти з доведенням ефективності запропонованого способу інтеграції SOM в мережі глибокого навчання. Експериментально доведено, що даний спосіб підвищує точність семантичної сегментації з 0.80672 до 0.81534 для задачі Salt identification Challenge та з 0.42502 до 0.43984 для задачі Cityscape. Експериментально визначено, що оптимальна ширина зрізу для наведених задач рівна 8x8 пікселів.

За результатами досліджень опубліковано 1 стаття [104], а також тези доповіді на міжнародній конференції [105].

ВИСНОВКИ

Основним результатом дисертаційного дослідження є створення методів розв'язання задач класифікації та семантичної сегментації на основі згорткових нейронних мереж та карт самоорганізації Кохонена, які дозволяють вирішити науково-практичну задачу підвищення точності розпізнавання за діагностичним сигналом.

У дисертації одержано такі основні результати:

1. На основі проведеного аналізу концепцій, архітектур, алгоритмів функціонування нейронних мереж визначено проблеми навчання та недостатньої точності розв'язання задач розпізнавання; обґрунтовано напрямки подальших досліджень щодо розпізнавання на основі згорткових нейронних мереж та карт самоорганізацій Кохонена.

2. Розроблено метод вдосконалення мереж глибокого навчання, призначених для семантичної сегментації, за рахунок примусового виділення контурів в додатковому блоці Собеля. Запропонований спосіб реалізовано для мереж: FPN, PSPNet, DeepLab v3, U-Net, що дозволило підвищити точність розв'язання тестових задач семантичної сегментації. Наприклад, для Salt identification Challenge з 0.80821 до 0.81463 та Cityscapes 0.42502 до 0.43603 на мережі U-Net.

3. Запропоновано блок агрегації сигналів різної розмірності, який забезпечує достатнє збільшення рецептивного поля для поточного сигналу. На основі запропонованого спосіб створено модифікації базових архітектур: ResNet18, EfficientNet-b0, WaveNet. Експериментально доведено ефективність запропонованого способу для підвищення точності класифікації сигналу великої розмірності за умови обмежених обчислювальних ресурсів. Наприклад, у випадку 1D сигналу на мережі WaveNet точність за метрикою macro F1 Score підвищено з 0.94406 до 0.94562; у випадку 2D-сигналу (зображення) точність за метрикою MAP@3 — з 0.9112 до 0.914.

4. Запропоновано метод класифікації за характеристичним сигналом з втратами у випадкові моменти часу на основі карт самоорганізації Кохонена за рахунок визначення відповідності неповного вхідного вектору вхідному шару нейронів. Модифікація карти самоорганізації на тестових задачах підвищує частку коректної класифікації в найкращому випадку з 14% до 69%.

5. Запропоновано метод асоціації відображення решітки самоорганізації з фрагментами вхідного сигналу для інтеграції блоку SOM в гібридну згорткову нейронну мережу. Експериментально доведено, що даний спосіб підвищує точність семантичної сегментації з 0.80672 до 0.81534 для задачі Salt identification Challenge та з 0.42502 до 0.43984 для задачі Cityscape.

6. Розроблено програмне середовище моделювання карт самоорганізацій Кохонена для розв'язання задачі кластеризації на основі існуючих програмних реалізацій: Encog, SOINN, ESOINN, NeuroPH, GSOM, а також власних реалізацій базового алгоритму SOM — SOMBase та вдосконаленого алгоритму — SOMmod. Середовище надає можливість обирати оптимальні за точністю розпізнавання карти самоорганізації Кохонена для розв'язання поточної задачі, а також додавати наявні розробки для експериментального доведення їх ефективності.

7. Результати дисертаційної роботи впроваджено в:

- програмному комплексу моніторингу гідрохімічного стану вод АЕС;
- програмне забезпечення розв'язання задач класифікації та семантичної сегментації на платформі машинного навчання Kaggle: (<https://www.kaggle.com/yuramuv/competitions>) SIIM-ISIC Melanoma Classification — віртуальна срібна медаль; University of Liverpool Ion Switching — віртуальна срібна медаль; Elo Merchant Category Recommendation — віртуальна срібна медаль; Quick, Draw! Doodle Recognition Challenge — віртуальна бронзова медаль; TGS Salt Identification Challenge — віртуальна бронзова медаль.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Профессиональный информационно-аналитический ресурс, посвященный машинному обучению, распознаванию образов и интеллектуальному анализу данных [Электронный ресурс] – Режим доступа до ресурсу: <http://www.machinelearning.ru/wiki/index.php?title=Классификация> (дата звернення: 17.03.2020).
2. Y.-C. Wang, M.-T. Li, J.-H. Zheng, Z.-C. Pan, “Pulsar Candidates Classification with Deep Convolutional Neural Networks”, *arXiv eprint* arXiv: 1909.05301 [Online], September 2019. (Last accessed: 12.03.2020) also *Research in Astronomy and Astrophysics*, Volume 19, Number 9, 133 (10 pp.), 2019. Available: <https://doi.org/10.1088/1674-4527/19/9/133> (Last accessed: 12.03.2020).
3. F. Morawski, M. Bejger, P. Ciecielag, “Deep learning classification of the continuous gravitational-wave signal candidates from the time-domain F-statistic search”, *arXiv eprint* arXiv: 1907.06917 [Online], July 2019. Available: <https://arxiv.org/pdf/1907.06917.pdf> (Last accessed: 17.03.2020).
4. M Shifat-E-Rabbi, X. Yin, C. E. Fitzgerald, G. K. Rohde, “Cell image classification: a comparative overview”, *arXiv eprint* arXiv: 1906.03316 [Online], June 2019. Available: <https://arxiv.org/pdf/1906.03316.pdf> (Last accessed: 12.03.2020) also *Cytometry. Part A* : the journal of the International Society for Analytical Cytology [Online], Feb 10 2020. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cyto.a.23984> (Last accessed: 17.03.2020) DOI: 10.1002/cyto.a.23984.
5. M. Böhle, F. Eitel, M. Weygandt, K. Ritter. “Layer-wise relevance propagation for explaining deep neural network decisions in mri-based alzheimer's disease classification”, *Frontiers in Aging Neuroscience* [Online], 11:194, 31 July 2019. Available: <https://doi.org/10.3389/fnagi.2019.00194> (Last accessed: 12.03.2020).

6. APTOS 2019 Blindness Detection Kaggle. (2019). At the 4th APTOS Symposium, experts will share how emerging, innovative digital technologies are used in different parts of the world to enhance ophthalmic care. Detect diabetic retinopathy to stop blindness before it's too late. Kaggle is the world's largest data science community with powerful tools and resources to help you achieve your data science goals [Online]. Available: <https://www.kaggle.com/c/aptos2019-blindness-detection>.

7. TGS Salt Identification Challenge Kaggle. (2019). TGS (the world's leading geoscience data company) is hoping Kaggle's machine learning community will be able to build an algorithm that automatically and accurately identifies if a subsurface target is salt or not [Online]. Available: <https://www.kaggle.com/c/tgs-salt-identification-challenge> (Last accessed: 17.03.2020).

8. Carvana Image Masking Challenge Kaggle. (2019). Carvana, a successful online used car startup, has seen opportunity to build long term trust with consumers and streamline the online buying process. [Online]. Available: <https://www.kaggle.com/c/carvana-image-masking-challenge>(Last accessed: 17.03.2020).

9. Dstl Satellite Imagery Feature Detection Kaggle. (2019). In this competition, Kagglers are challenged to accurately classify features in overhead imagery. [Online] Available: <https://www.kaggle.com/c/dstl-satellite-imagery-feature-detection>(Last accessed: 17.03.2020).

10. R. Hausen and B. Robertson, "Morpheus: A Deep Learning Framework For Pixel-Level Analysis of Astronomical Image Data", *arXiv preprint arXiv:1906.11248* [Online], June 2019. Available: <https://arxiv.org/pdf/1906.11248.pdf> (Last accessed: 12.03.2020).

11. H. Imai, S. Matzek, T. D. Le, Y. Negishi, and K. Kawachiya, "Fast and accurate 3d medical image segmentation with data-swapping method", *arXiv preprint arXiv: 1812.07816* [Online], 19 December, 2018.

Available: <https://arxiv.org/ftp/arxiv/papers/1812/1812.07816.pdf> (Last accessed: 12.03.2020).

12. A. Myronenko, “3D MRI brain tumor segmentation using autoencoder regularization”, in *Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II.*, 2018. pp. 311-320. Available: http://link.springer.com/10.1007/978-3-030-11726-9_28 (Last accessed: 12.03.2020).

13. H. R. Roth et al., “Hierarchical 3D fully convolutional networks for multi-organ segmentation”, *arXiv preprint arXiv:1704.06382* [Online], 21 Apr 2017. Available: <https://arxiv.org/pdf/1704.06382.pdf> (Last accessed: 12.03.2020).

14. U. Kamal, A. M. Rafi, R. Hoque, and M. K. Hasan, “Lung cancer tumor region segmentation using recurrent 3D-DenseUNet”, *arXiv preprint arXiv:1812.01951* [Online], 5 December, 2018. Available: <https://arxiv.org/pdf/1812.01951.pdf> (Last accessed: 15.03.2020).

15. V. Vapnik and A. Lerner, “Pattern Recognition Using Generalized Portrait Method”, *Automation and Remote Control*, Vol. 21, no. 6, pp. 709–715, 1963. (Translated from *Avtomatika i Telemekhanika*, Vol. 24, no. 6, pp. 774–780, Jun, 1963). Available: <http://web.cs.iastate.edu/~cs573x/vapnik-portraits1963.pdf> (Last accessed: 12.03.2020).

16. P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features”, in *Proceedings CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, HI, USA, Dec. 2001, vol. 1 pp. 511-518. DOI: 10.1109/CVPR.2001.990517.

17. M. Maron, “Automatic indexing: an experimental inquiry”, *Journal of the Association for Computing Machinery* Vol. 8, no. 3, pp. 404–417, July 1961. Available: <https://doi.org/10.1145/321075.321084>.

18. Y. Freund and R. E. Schapire, “Game theory, on-line prediction and boosting”, in *Proceedings of the Ninth Annual Conference on Computational*

Learning Theory, Desenzano del Garda, Italy, 1996, pp. 325–332. Available: <https://doi.org/10.1145/238061.238163>.

19. А. Конушин, “Сегментация изображений” (Что такое сегментация? Разбиение изображения на неперекрывающиеся области), Лекция , 2020. [Электронный ресурс] – Режим доступа до ресурсу: <https://docplayer.ru/36006552-Segmentaciya-izobrazheniy-anton-konushin.html> (дата звернения: 18.03.2020).

20. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition”, in *Proceedings of the IEEE*, Vol. 86, issue. 11, pp. 2278–2324, Nov. 1998. DOI: 10.1109/5.726791 (Last accessed: 12.03.2020).

21. F. Rosenblatt, “The perceptron: A perceiving and recognizing automaton (Project PARA)”, *Technical Report*, New York, 85-460-1, 1957. Available: <https://blogs.umass.edu/brain-wars/files/2016/03/rosenblatt-1957.pdf> (Last accessed: 12.03.2020).

22. Галушкин А. И. Синтез многослойных систем распознавания образов. М. : Энергия, 1974. 366 с.

23. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning Representations by Back-Propagating Errors”, *Nature*, Vol. 323, issue 6088, pp. 533-536, October 1986. DOI:10.1038/323533a0 Available: https://www.iro.umontreal.ca/~vincentp/ift3395/lectures/backprop_old.pdf (Last accessed: 16.03.2020).

24. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks”, *Communications of the ACM*, Vol. 60, no. 6, pp. 84-90. May 2017. Available: https://cdn.jsdelivr.net/gh/crazyjums/crazyjums.github.io@master/images/article/pdf/ImageNet_Classification_with_Deep_Convolutional_Neural_Networks.pdf (Last accessed: 12.03.2020) <https://doi.org/10.1145/3065386>.

25. ImageNet. (2016). An image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy

is depicted by hundreds and thousands of images [Online]. Available: <http://www.image-net.org/> (Last accessed: 18.03.2020).

26. “Стэнфордский курс. Лекция 9: Архитектуры CNN” [Online]. Available: <https://www.reg.ru/blog/stehnfordskij-kurs-lekciya-9-arhitektury-cnn/> (Last accessed: 18.03.2020).

27. K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, in Proceedings 3rd International Conference on Learning Representations , *ICLR 2015*, San Diego, CA, USA, May 7-9, 2015, 14 p. Available: <https://arxiv.org/pdf/1409.1556.pdf> (Last accessed: 12.03.2020).

28. M. Ferguson, R. Ak, Y.-T. T. Lee, K. H. Law, “Automatic localization of casting defects with convolutional neural networks”, in: *IEEE International Conference on Big Data (Big Data)*, Boston, MA, USA, December 2017, pp. 1726–1735. Available: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=924455 (Last accessed: 12.03.2020) DOI: 10.1109/BigData.2017.8258115.

29. X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks”, in *Journal of Machine Learning Research*, Vol. 9: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, Chia Laguna Resort, Sardinia, Italy, 13-15 May 2010, pp. 249-256. Available: <http://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf> (Last accessed: 12.03.2020).

30. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, in *Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 27-30 June 2016, 12 p. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf (Last accessed: 14.03.2020) DOI:10.1109/CVPR.2016.90

31. S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks”, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 21-26 July 2017, pp. 1492-1500. Available: http://openaccess.thecvf.com/content_cvpr_2017/

papers/Xie_Aggregated_Residual_Transformations_CVPR_2017_paper.pdf (Last accessed: 14.03.2020) DOI:10.1109/CVPR.2017.634

32. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 27-30 June 2016, pp 2818-2826. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Szegedy_Rethinking_the_Inception_CVPR_2016_paper.pdf (Last accessed: 14.03.2020) DOI: 10.1109/CVPR.2016.308.

33. C. Szegedy, S. Ioffe, and V. Vanhoucke, “Inception-v4, inception-resnet and the impact of residual connections on learning”, in *AAAI'17: Proceedings 31rd Conference on Artificial Intelligence*, San Francisco, California, USA, 4 – 9 February 2017, June 23 2017, pp. 4278-4284. Available: aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14806/14311 (Last accessed: 14.03.2020).

34. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting”, in *Journal of Machine Learning Research*, Vol. 15, no. 1, pp. 1929-1958, 2014. Available: www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf (Last accessed: 14.03.2020).

35. C. Shorten, T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning”, *Journal of Big Data* 2019, Vol. 6, Article no 60, 06 July, 48 p., 2019. Available: <https://journalofbigdata.springeropen.com/track/pdf/10.1186/s40537-019-0197-0> (Last accessed: 14.03.2020).

36. S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, Y. Yoo, “Cutmix: Regularization strategy to train strong classifiers with localizable features”, in *ICCV 2019: Proceedings of the 17th International Conference on Computer Vision*, Seoul, Korea, 27 October–2 November 2019, 27 February 2020, pp. 6022-6032. Available: http://openaccess.thecvf.com/content_ICCV_2019/papers/Yun_CutMix_Regularization_Strategy_to_Train_Strong_Classifiers_With_Localizable_Feature

s_ICCV_2019_paper.pdf (Last accessed: 12.03.2020)
DOI: 10.1109/ICCV.2019.00612.

37. J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks", *arXiv preprint arXiv: 1709.01507* [Online], September 2017. (Last accessed: 12.03.2020) also in Proceedings of the *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 18-23 June 2018, 17 December 2018, pp. 7132-7141. Available: http://openaccess.thecvf.com/content_cvpr_2018/papers/Hu_Squeeze-and-Excitation_Networks_CVPR_2018_paper.pdf (Last accessed: 16.03.2020) DOI: 10.1109/CVPR.2018.00745.

38. A. G. Roy, N. Navab, and C. Wachinger, "Concurrent Spatial and Channel Squeeze & Excitation in Fully Convolutional Networks", *arXiv preprint arXiv: 1803.02579* [Online], 2018. Available: (Last accessed: 12.03.2020) also in Proceedings of the MICCAI 2018: International Conference on Medical Image Computing and Computer Assisted Intervention [Online], Granada, Spain, September 16-20, pp 421-429. https://doi.org/10.1007/978-3-030-00928-1_48 (Last accessed: 16.03.2020).

39. S. Ioffe, C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", in *CML'15: Proceedings of the 32nd International Conference on International Conference on Machine Learning*, Lille, France, France, 6-11 July 2015, *Proceedings of Machine Learning Research*, Vol. 37, July 2015, pp. 448–456. Available: <https://arxiv.org/pdf/1803.02579.pdf> (Last accessed: 15.03.2020).

40. V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation", *arXiv preprint arXiv:1511.00561* [Online], November 2015. *arXiv:1511.00561v3 [cs.CV]* [Online], 10 Oct 2016. Available: ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7803544 (Last accessed: 15.03.2020).

41. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation", in *MICCAI 2015: Proceedings 18th*

International Conference Munich, Germany, October 5-9 2015, Part III. Lecture Notes in Computer Science, Vol. 9351, pp.234–241, Springer, 2015. : Available: <https://arxiv.org/pdf/1505.04597.pdf> (Last accessed: 15.03.2020) DOI: 10.1007/978-3-319-24574-4_28

42. T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection”, in *CVPR 2017: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 21-26 2017, IEEE Computer Society 2017, 2017, pp. 936-944. Available: <https://arxiv.org/pdf/1612.03144.pdf> (Last accessed: 15.03.2020) DOI: 10.1109/CVPR.2017.106.

43. H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, “Pyramid scene parsing network”, in *CVPR 2017: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 21-26 2017, IEEE Computer Society 2017, 2017, pp. 6230-6239. DOI: 10.1109/CVPR.2017.660.

44. D. J Willshaw, C. Von Der Malsburg C. “How patterned neural connections can be set up by self-organization”, in *Proceedings of the Royal Society of London. Series B, Biological Sciences*, Vol. 194, no. 1117, pp. 431–445, November 1976. Available: <https://web.cs.dal.ca/~tt/CSCI650804/01cce44035005015266e0&0.pdf> (Last accessed: 16.03.2020) DOI: 10.1098/rspb.1976.0087.

45. T. Kohonen, “Self-organizing formation of topologically correct feature maps”, *Biological Cybernetics*, Vol. 43, no. 1, pp. 59–69, 1982. Available: http://www.cnbc.cmu.edu/~tai/nc19journalclubs/Kohonen1982_Article_Self-organizedFormationOfTopol.pdf (Last accessed: 16.03.2020).

46. B. Fritzke: Growing Cell Structures - a self organizing network for unsupervised and supervised learning, *Neural Networks*, Vol.7, pp.1441-1460, 1995.

47. M. - C. Su and H. - T. Chang, “Genetic-algorithms-based approach to self-organizing feature map and its application in cluster analysis”, in *1998 IEEE International Joint Conference on Neural Networks : Proceedings IEEE World*

Congress on Computational Intelligence, Anchorage, AK, USA, 4-9 May 1998, (*Cat. No.98CH36227*), Anchorage, AK, 1998, vol. 1, pp. 735-740. DOI: 10.1109/IJCNN.1998.682372

48. S. Furao, “An Algorithm for Incremental Unsupervised Learning and Topology Representation”, Ph.D Thesis, Tokyo Institute of Technology, 2006. Available: http://haselab.info/papers/shen_doctoralThesis.pdf (Last accessed: 12.03.2020)

49. S. Furao, T. Ogura, and O. Hasegawa, “An enhanced self-organizing incremental neural network for online unsupervised learning”, *Neural Networks*, Vol. 20, issue 8, pp. 893-903, October 2007. DOI: 10.1016/j.neunet.2007.07.008 Available: http://haselab.info/soinn/img/pdf/02_soinn-tutorial.pdf (Last accessed: 16.03.2020) (Download papers and program of SOINN <http://www.isl.titech.ac.jp/~hasegawalab/soinn.html>).

50. M. Cao, A. Li, Q. Fang, and B. J. Kroger, “Growing self-organizing map approach for semantic acquisition modeling”, in *CogInfoCom 2013: proceedings of the 4th IEEE International Conference on Cognitive Infocommunications*, Budapest, Hungary, December 2–5 2013, 2013, pp. 33–38. Available: <https://doi.org/10.1109/CogInfoCom.2013.6719269> (Last accessed: 15.03.2020).

51. A. El Golli, “Speeding up the self organizing map for dissimilarity data”, in *Proceedings of International Symposium on Applied Stochastic Models and Data Analysis*, Brest, France, 2005, pp. 709-713.

52. Z. Zhao, “Improvements to Kohonen self-organising algorithm”, *Electronics Letters*, vol. 30, issue 6, pp. 502-503, 17 Mar 1994. DOI: 10.1049/el:19940346.

53. Ю. В. Москаленко, “Вдосконалення алгоритмів функціонування карт самоорганізації”, у *Сучасні аспекти розробки програмного забезпечення : зб. тез II наук.-практ. конф. молодих вчених і спеціалістів в області розробки програмного забезпечення*, Київ, 22 трав. 2017 р. Черкаси : Чабаненко Ю.А., 2017. С. 51-53.

54. M. Lin, Q. Chen, and S. Yan, “Network in network”, in *ICLR 2014: Conference Track Proceedings of the 2nd International Conference on Learning Representations*, Banff, AB, Canada, April 14-16, 2014, *arXiv preprint arXiv:1312.4400* [Online], 16 Dec 2013. Available: <https://arxiv.org/pdf/1312.4400.pdf> (Last accessed: 15.03.2020).

55. Gradual training on large datasets[Online] Available: <https://examples.dask.org/machine-learning/incremental.htm>.

56. V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines”, in *ICML10: Proceedings of the 27th International Conference on Machine Learning*, Haifa, Israel, June 21-24 2010, 2010, pp. 807-814. Available: <https://icml.cc/Conferences/2010/papers/432.pdf> (Last accessed: 12.03.2020).

57. K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”, in *ICCV 2015: Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 7-13 Dec. 2015, 18 February 2016, pp. 1026-1034. DOI: 10.1109/ICCV.2015.123 Available: Feb 2015. <https://arxiv.org/pdf/1502.01852.pdf> (Last accessed: 12.03.2020).

58. V. Iglovikov, S. Seferbekov, A. Buslaev, A. Shvets, “TernausNetV2: Fully Convolutional Network for Instance Segmentation”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Salt Lake City, UT, USA, 18–22 June 2018; pp. 233–237. Available: <https://arxiv.org/pdf/1806.00844.pdf> (Last accessed: 12.03.2020).

59. S. Das, V. Pune, “Comparison of Various Edge Detection Technique”, *International Journal of Signal Processing, Image Processing and Pattern Recognition*, Vol. 9, no. 2, pp. 143–158, 2016. Available: <https://pdfs.semanticscholar.org/2605/3e11e9cf73b168b91646b63446f119276feb.pdf> (Last accessed: 12.03.2020). DOI: 10.14257/ijsp.2016.9.2.13.

60. О. Й. Піцун, *Методи і засоби опрацювання біомедичних зображень в системах автоматизованої мікроскопії* : дис. канд. техн. наук : 05.13.23 / Тернопіл. нац. екон. ун-т, нац. ун-т «Львів. політехніка». Львів, 2018, 166 с.

62. Kaggle TGS Salt Identification Challenge - Semantic Segmentation competition from TGS to segment salt deposits beneath the Earth's surface [Online] Available: <https://www.kaggle.com/c/tgs-salt-identification-challenge> (Last accessed: 12.03.2020).

63. Cityscapes: dataset to learn deep neural networks: [Online] Available: <https://www.cityscapes-dataset.com/> (Last accessed: 12.03.2020).

64. Cityscapes Image Pairs: semantic image segmentation of urban pair images are presented: [Online] Available: <http://www.kaggle.com/dansbecker/cityscapes-image-pairs> Available: (Last accessed: 12.03.2020).

65. Mingxing Tan and Quoc V Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, 2019.

66. A. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

67. Wavenet with SHIFTED-RFC Proba and CBR. Available at: <https://www.kaggle.com/nxrprime/wavenet-with-shifted-rfc-proba-and-cbr>

68. University of Liverpool — Ion Switching. Available at: <https://www.kaggle.com/c/liverpool-ion-switching/>

69. Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Focal loss for dense object detection. *ICCV*, 2017

70. S. S. M. Salehi, D. Erdogmus, and A. Gholipour, “Tversky loss function for image segmentation using 3d fully convolutional deep networks,” in *International Workshop on MLMI*, 2017.

71. Th A Sorensen. 1948. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol. Skar.*, 5:1–34.

72. Quick, Draw! Doodle Recognition Challenge). Available at: <https://www.kaggle.com/c/quickdraw-doodle-recognition>

73. S. Shapovalova, Y. Moskalenko, “Semantic segmentation accuracy improvement based on forced edge detection”, *Адаптивні системи автоматичного управління*, 2019, Т. 2, № 2(35), С. 79-87. Available: <http://asac.kpi.ua/article/view/197435/197629> (Last accessed: 15.03.2020) DOI: 10.20535/1560-8956.35.2019.197435.

74.S. Shapovalova , Y. Moskalenko “Methods for increasing the classification accuracy based on modifications of the basic architecture of convolutional neural networks”. *ScienceRise*. 2020. No 6 (71). P. 10–16. DOI:10.21303/2313-8416.2020.001550

75. S. Shapovalova, Y. Moskalenko, “Segmentation based problem solving on convolutional neural network”, in *Modern Aspects of Software Development: Proceedings 6th International Scientific and Practical Virtual Conference of Software Development Specialists*, Kyiv, Ukraine, 24 June 2019, 2019, pp. 47-53. URL: (Last accessed: 12.03.2020).

76. L. I. Passoni, and et al., “Unsupervised learning segmentation for dynamic speckle activity images”, *arXiv preprint arXiv 1408.3818* [Online], 17 Aug 2014. Available: <https://arxiv.org/abs/1408.3818> URL: (Last accessed: 12.03.2020).

77. D. Fustes, and et al., “An approach to the analysis of SDSS spectroscopic outliers based on self-organizing maps-Designing the outlier analysis software package for the next Gaia survey” [Online], *Astronomy & Astrophysics*, Vol. 559, id. A7, 10 pp., 28 October 2013. Available: <https://www.aanda.org/articles/aa/pdf/2013/11/aa21445-13.pdf> (Last accessed: 15.03.2020) DOI: 10.1051/0004-6361/201321445.

78. H Meusinger, et al. “A large sample of Kohonen selected E+ A (post-starburst) galaxies from the Sloan Digital Sky Survey” [Online], *Astronomy & Astrophysics*, Vol. 597, id. A134, 20 January 2017, 29 pp.

Available: <https://www.aanda.org/articles/aa/pdf/2017/01/aa29139-16.pdf> (Last accessed: 12.03.2020) DOI: 10.1051/0004-6361/20162139.

79. D. Fraccalvieri, L. Bonati, F. Stella, “Self Organizing Maps to efficiently cluster and functionally interpret protein conformational ensembles”, In Proceedings Wivace 2013: Electronic Proceedings in Theoretical Computer Science (EPTCS 130) - Italian Workshop on Artificial Life and Evolutionary Computation [Online], Milan, Italy, 1-2 July 2013. 2013, pp.83-86, *arXiv preprint* arXiv: 1309.7694 [Online], September 2013. DOI: 10.4204/EPTCS.130.13 Available: <https://arxiv.org/pdf/1309.7694.pdf> (Last accessed: 15.03.2020).

80. M.- C. Su, T.- K. Liu, H. -T. Chang, “Improving the Self-Organizing Feature Map Algorithm Using an Efficient Initialization Scheme”, *Tamkang Journal of Science and Engineering*, Vol. 5, no. 1, pp. 35–48, 2002. Available: <http://jase.tku.edu.tw/5-1/5-1-5.pdf> (Last accessed: 12.03.2020).

81. С. І. Шаповалова, Г. І. Шараєвський, “Комп’ютерне моделювання карти самоорганізації для розв’язання задачі розпізнавання сигналів”, *Вісник національного університету “Львівська політехніка”. Автоматика, вимірювання та керування* : зб. наук. пр., No 574, 2007, С. 75–81. Available: http://vlp.com.ua/files/13_2.pdf (Last accessed: 12.03.2020).

82. B. Conan-Guez, F. Rossi, A. El Golli, “Fast algorithm and implementation of dissimilarity self-organizing maps”, *Neural Networks*, Vol. 19, no. 6-7, pp. 855-863, 2006. *arXiv preprint* arXiv:0709.3461 [Online], September 2007. (Last accessed: 12.03.2020) Available: <https://arxiv.org/pdf/0709.3461.pdf> (Last accessed: 15.03.2020) DOI: 10.1016/j.neunet.2006.05.002.

83. E. Cuadros-Vargas, R. F. Romero, and K. Obermayer, “Speeding up algorithms of SOM Family for Large and High Dimensional Databases”, in *WSOM 2003: Proceedings of the Workshop on Self organizing Maps*, Kyushu, Japan, 2003, pp. 167-172. Available: ni.tu-berlin.de/fileadmin/fg215/articles/cuad_03_wsom.pdf (Last accessed: 12.03.2020).

84. M. Cao, A. Li, Q. Fang, E. Kaufmann, and B. J. Kröger, “Interconnected growing self-organizing maps for auditory and semantic acquisition modeling” [online], *Frontiers in Psychology*, Vol. 5, Article 236, 20 March 2014, 20 pp. Available :www.ncbi.nlm.nih.gov/pmc/articles/PMC3960950/ (Last accessed: 12.03.2020) DOI: 10.3389/fpsyg.2014.

85. С. Білоусов, Алгоритм Улучшенной Самоорганизующейся Растущей Нейронной Сети (ESOINN) // habr.com [Электронный ресурс] – Режим доступа: <https://habr.com/post/206116> (Дата звернення: 11.09.2018).

86. Дж. Ту, Р. Гонсалес, *Принципы распознавания образов*. М. : Мир, 1978, 414 с.

87. Encog Machine Learning Framework // GitHub [Online] Available: <https://github.com/encog/encog-java-core> URL: (Last accessed: 12.03.2020).

88. Neuroph - Java Neural Network Platform Neuroph // GitHub [Online] Available: <https://github.com/neuroph/neuroph> URL: (Last accessed: 12.03.2020).

89. Self-Organizing Incremental Neural Network // GitHub [Online] Available: <https://github.com/fukatani/soinn> URL: (Last accessed: 12.03.2020).

90. An enhanced self-organizing incremental neural network for online unsupervised learning // github [Online] Available: <https://github.com/BelBES/ESOINN> URL: (Last accessed: 12.03.2020).

91. Growing Self-Organizing Map // GitHub [Online] Available: <https://github.com/philippludwig/pygsom>

92. S. Shapovalova, Yu. Moskalenko, “Increasing the share of correct clustering of characteristic signal with random losses in self-organizing maps”, *Eastern-European Journal of Enterprise Technologies*, Vol. 2, no 4(98), pp. 13-21, 2019. DOI: 10.15587/1729-4061.2019.160670.

93. Ю. В. Москаленко, С. І. Шаповалова, “Класифікація кривих другого порядку по неповному вхідному вектору”, на *Сучасні проблеми наукового забезпечення енергетики* : матеріали XIII Міжнар. наук.-практ. конф. аспі-

рантів, магістрантів і студентів, м. Київ, 21–24 квіт. 2015 р. У 2 т. / Нац. техн. ун-т України “Київський політехнічний інститут” ім. І. Сікорського, Київ : НТУУ “КПІ”, 2015, Т. 2, С. 129. URL: <http://aesitf.kpi.ua/wp-content/uploads/2015/04/2015vol2.pdf>.

94. Ю.В. Москаленко, “Класифікація зашумлених діагностичних сигналів”, на *Сучасні проблеми наукового забезпечення енергетики* : матеріали XVII Міжнар. наук.-практ. конф. аспірантів, магістрантів і студентів. м. Київ, 23–26 квіт. 2019 р. У 2 т. / Нац. техн. ун-т України “Київський політехнічний інститут” ім. І. Сікорського, Київ : НТУУ “КПІ”, 2019, Т.2, С. 92. URL: https://tcf.kpi.ua/files/pdf/tezi-tom2_2022.pdf.

95. Github. [Online] Available: <https://github.com/> (Last accessed: 12.03.2020).

96. Iris Data Set. [Online] Available: <http://archive.ics.uci.edu/ml/datasets/Iris> (Last accessed: 12.03.2020).

97. Handwritten digit database. [Online] Available: <http://cis.jhu.edu/~sachin/digit/digit.html>. (Last accessed: 12.03.2020).

98. Ю. В. Москаленко, “Середовище моделювання нейронних мереж для розв’язання задачі кластеризації”, *Математичне та комп’ютерне моделювання. Технічні науки*, 2019. Вип. 20, С. 68–78.

99. Ю. В. Москаленко, С. І. Шаповалова, “Середовище моделювання карт самоорганізації для розпізнавання геометрично схожих фігур”, на *Сучасні аспекти розробки програмного забезпечення* : зб. тез II наук.-практ. конф. молодих вчених і спеціалістів в області розробки програмного забезпечення, Київ, 22 травня 2015 р., Черкаси : Чабаненко Ю. А., 2015, С. 51–53.

100. H. Dozono, G. Niina, Gen, S. Araki, “Convolutional self-organizing map”, in *CSCI 2016: Proceedings of the International Conference on Computational Science and Computational Intelligence*, Las Vegas, NV, USA, 15-17 Dec. 2016, 20 March 2017, pp 767–771. Available: <https://doi.org/10.1109/CSCI.2016.0149> (Last accessed: 12.03.2020).

101. N. Liu, J. Wang, Y. Gong, “Deep Self-Organizing Map for visual classification”, in *Neural Networks (IJCNN)*. International Joint Conference, Killarney, Ireland, 12-17 July 2015, 01 October 2015, pp. 1–6. DOI: 10.1109/IJCNN.2015.7280357.

102. T. Trappenberg, P. Hollensen, P. Hartono, “Classifier with hierarchical topographical maps as internal representation”, in *Intelligent engineering systems (INES)*: Proceedings of the IEEE 19th International Conference, Bratislava, Slovakia, 3-5 Sept. 2015, 23 November 2015, pp. 341–345. Available: <https://arxiv.org/pdf/1412.6567.pdf> (Last accessed: 12.03.2020) DOI:10.1109/INES.2015.7329752.

103. С. Хайкин, *Нейронные сети: полный курс*, 2-е изд. испр. / пер. с англ. Н. Н. Куссуль. М. : Вильямс, 2016, 1104 с.

104. S. Shapovalova, Y. Moskalenko, “Rate increase of the objects classification on the convolutional neural networks with the self-organization maps implementation”, у *Сучасні проблеми моделювання – Modern Problems of Modeling* : зб. наук. пр., Мелітополь : Вид-во МДПУ ім. Б. Хмельницького, 2020, Вип. 17, С. 145–155. DOI: 10.33842/2313-125X/2019/17/145/155

105. Ю. В. Москаленко “Гібридні нейронні мережі на основі карт самоорганізації Кохонена”, на *Комп’ютерне моделювання та оптимізація складних систем (КМОСС-2019)* : матеріали V-ї Міжнар. наук.-техн. конф., м. Дніпро, 6–8 листоп. 2019 р. / М-во освіти і науки України, ДВНЗ «Український державний хіміко-технологічний університет», Дніпро : Баланс-клуб, 2019, С. 133–134. URL: <http://orgconf.com/infolist/kmoss2019.pdf>.

Додаток а

АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

Національна академія медичних наук України
Державна установа
“ІНСТИТУТ ГЕРОНТОЛОГІЇ
ім. Д.Ф.Чеботарьова”

Адреса: 04114, Київ, Вишгородська 67
телефон: (044) 430-40-68, факс: (044) 432-99-56, 430-41-34

для телеграм: Київ-114, Геронтологія
транспортний код: 4378

електронна пошта: ig@geront.kiev.ua

“16” 12 2020 р.

№ 01/137

АКТ

**про впровадження наукових результатів дисертаційної роботи
на здобуття наукового ступеня доктора філософії
Москаленка Юрія Володимировича**

Цей акт засвідчує, що програмне забезпечення попередньої діагностики злоякісних пухлин за знімками родимих плям, розроблене Москаленко Ю.В. в співавторстві з Шаповаловою С.І. в Національному технічному університеті України «Київський політехнічний інститут імені Ігоря Сікорського» на основі набору даних Melanoma Classification, представлених Society for Imaging Informatics in Medicine (SIIM) та International Skin Imaging Collaboration (ISIC) на платформі машинного навчання Kaggle, прийнято до впровадження у відділі клінічної фізіології та патології внутрішніх органів та в лабораторії математичного моделювання процесів старіння.

Дослідження, проведені в дисертаційній роботі, дозволили підвищити точність розпізнавання.

Директор Інституту
академік НАМН України

В.В. Безруков

Завідувач відділу клінічної фізіології та
патології внутрішніх органів
академік НАМН України

О.В. Коркушко

Завідувач лабораторії
математичного моделювання
процесів старіння доктор мед. наук

А.В. Писарук

