

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Міністерства освіти і науки України

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Міністерства освіти і науки України

Кваліфікаційна наукова праця на правах рукопису

Зозюк Максим Олегович

УДК 538.9, 539.2, 539.3, 620.3,

ДИСЕРТАЦІЯ

Згорткова нейронна мережа для прогнозування коефіцієнту пропускання
метаматеріалів в залежності від їх структури

Спеціальність 153 – Мікро- та наносистемна техніка

Галузь знань 15 – Автоматизація та приладобудування

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання чужих ідей,
результатів і текстів мають посилання на відповідне джерело

Науковий керівник Корольок Дмитро Володимирович, доктор фізико-математичних
наук, професор

Київ – 2023

АНОТАЦІЯ

Зозюк М. О. Згорткова нейромережа для прогнозування коефіцієнту пропускання метаматеріалів в залежності від їх структури. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 153 – Мікро- та наносистемна техніка (галузь знань 15 – Автоматизація та приладобудування). – Національний технічний університет України, Київ, 2022.

Робота присвячена розробці методики прогнозування коефіцієнту пропускання метаматеріалів на основі топологічної структури та складу із застосуванням технік глибокого навчання, а саме – математичні нейронні мережі з використанням операцій згортки та допоміжних.

Науково-прикладні дослідження, висвітлені в дисертаційній роботі, зосереджені на практичному дослідженні згорткових нейронних мереж для виконання завдань прогнозування у сферах проєктування, передбачення метаматеріалів з потрібними властивостями.

В сучасних науково-технічних дослідженнях методи глибокого навчання: згорткові нейронні мережі, дають можливість розробити прості й ефективні методи аналізу та прогнозу властивостей в задачах дизайну метаматеріалів. Перевагою цієї техніки є зручність в реалізації, наявність даних для цього підходу, швидкість в обчисленнях у порівнянні з точними методами прогнозування властивостей і топологічної структури на основі фізичних законів. Звичайно, нейронні мережі є доволі складним процесом, який має свої недоліки – необхідність у великій кількості даних, відносна складність оптимізації, складність постановки задачі. З розвитком технологій глибокого навчання, ці недоліки все більш нівелюються, і тому їх використання стає доступнішим.

Нейронні мережі використовуються для дизайну метаматеріалів, тобто штучних матеріалів, що мають непередбачуваний зворотний відгук на

електромагнітні хвилі. Основна ідея полягає в тому, що нейронні мережі використовуються для оптимізації топології та параметрів метаматеріалів.

Перш за все, нейронна мережа навчається на основі великої кількості вхідних-вихідних даних, де вхідні дані становлять собою деякі параметри метаматеріалу (наприклад, розмір, форма, компоненти), а вихідні дані - бажаний відгук на світло (наприклад, поглинання, відбивання, пропускання).

Після завершення навчання, нейронна мережа може бути використана для прогнозування вихідних даних на основі нових вхідних даних, тобто вона може передбачати як відгук метаматеріалу на нові електромагнітні хвилі.

Нейронні мережі можуть допомогти в процесі дизайну метаматеріалів, оскільки вони можуть досліджувати велику кількість варіантів метаматеріалів та ефективно оптимізувати їх параметри з метою досягнення бажаного функціонального відгуку.

Одне з переваг використання нейронних мереж у дизайні метаматеріалів полягає в тому, що вони можуть враховувати складні взаємозв'язки між параметрами матеріалів і функціональним відгуком. Вони здатні знаходити нелінійні залежності, які можуть бути складні для методів традиційного проєктування.

Отже, нейронні мережі дозволяють автоматизувати та оптимізувати процес дизайну метаматеріалів, забезпечуючи швидшу та ефективнішу розробку нових матеріалів з бажаними властивостями.

Для досліджень використовувались сучасні напрацювання для згорткових нейронних мереж щодо підходів, які застосовуються для подолання їх обмежень.

Було використано велику кількість інформації про метаматеріали з відповідних джерел, а саме – топологія та склад метаматеріалів.

Використовувались програмні середовища для написання цифрового коду та побудови 3D об'єктів метаматеріалів з визначеними властивостями.

Проаналізована можливість прогнозування коефіцієнту пропускання на основі інформації про структуру та складу метаматеріалів. Показано, що така можливість існує й описані основні умови, які накладаються на дані про метаматеріали, які потрібні для використання згорткових нейронних мереж при розв'язанні задач з прогнозування інформації про метаматеріали.

Встановлено, що дані збережені в «.ply» форматі, не підходять, як вхідні дані для нейронної мережі. Був використаний програмний пакет Open3D на базі Python, для конвертації початкового формату у формат «.xyzrgb», який становить собою масив, де кожен рядок – це вектор з шести чисел; перші три – це координати, три інших – це RGB числа. Показана можливість використання такого вигляду інформації в процесах згортки та для навчання нейронної мережі. Представлено умови, які накладаються на дані: однорідність даних, що означає, що всі параметри, які впливають на результат прогнозування, мають або бути однаковими, або бути внесені в дані, як інформація про властивості; масштабованість, що означає, що дані мають бути зведені до однакових діапазонів значень та масштабів.

Розроблено алгоритм прогнозування властивостей на основі структури та складу метаматеріалів із використанням згорткової нейронної мережі.

Розроблено алгоритм для збереження інформації про склад метаматеріалів. Показано, що використовуючи інформації про електромагнітні властивості компонентів є можливість передбачувати електромагнітні властивості метаматеріалів. Побудовано в цифровому середовищі 3D об'єкти метаматеріалів із закріпленням за кожним пікселем інформації про компонент. Описано алгоритм збереження цієї інформації у зручному для навчання згорткової нейронної мережі вигляді. Представлено види цифрових форматів, які можуть бути використані для збереження потрібної про метаматеріали інформації. Проаналізовано формат, який використовувався для збереження інформації про 3D об'єкт в цій роботі.

Описано процес представлення властивостей метаматеріалів у зручному для навчання згорткової нейронної мережі вигляді. Два методи було використано для

порівняння ефективності обох методів. Показано, що метод з представлення характеристик у вигляді коефіцієнтів полінома є швидшим, але не відповідним для вирішення задач з прогнозування характеристик метаматеріалів.

Проаналізовано інші формати збереження інформації про 3D структури. Досліджено, що такі формати, як «.fbx», «.obj», «.stl», «.3ds» не підходять, через такі причини: відсутність ефективного програмного функціонала (для конвертування в потрібний для нейронної мережі формат); відсутність збереження інформації про додаткові канали інформації; присутність зайвої інформації, яку немає можливості відділити від необхідної.

Розроблено архітектуру згорткової нейронної мережі для прогнозування частотних електромагнітних характеристик на основі структури та складу метаматеріалів. Показані результати прогнозування для обох випадків з представлення коефіцієнту пропускання – у вигляді масиву точок та коефіцієнтів поліному. Проаналізовано результати прогнозування та приведені способи до покращення цих результатів та оптимізації мережі для зменшення часу виконання навчання та збереження ресурсів. Вказано, що збільшення даних є найбільш ефективним методом для покращення результатів прогнозування. Попри це, методи покращення результатів, які засновані на зміні архітектури та зміни гіперпараметрів потрібно постійно оцінювати та використовувати при можливості.

Ключові слова: згорткова нейронна мережа, дизайн метаматеріалів, 3D модель, коефіцієнт пропускання, глибоке навчання, діелектрична проникність, вимірювання параметрів матеріалів, діелектричні матеріали, апроксимаційна модель, експериментально-аналітичний метод, візуалізація інформації, нелінійна модель, інтелектуальне обчислення, комп'ютерне програмування, діелектрики, чисельне моделювання, машинне навчання, математичне моделювання, передбачення.

ABSTRACT

Zozyuk Maksym Application of convolutional neural network for predicting the coefficient of passage of metamaterials depending on the structure and physical composition of metamaterials. – Qualifying scientific work on manuscript rights.

Dissertation for obtaining the scientific degree of Doctor of Philosophy in specialty 153 – Micro- and nanosystem engineering (field of knowledge 15 – Automation and instrument engineering). - National Technical University of Ukraine, Kyiv, 2022.

The work is devoted to the development of a technique for predicting the penetration coefficient of metamaterials based on topological structure and physical composition with the use of machine learning techniques, namely artificial neural networks using convolution and auxiliary operations.

Scientific and applied research, highlighted in the dissertation, is focused on the practical study of artificial convolutional neural networks for performing forecasting tasks in the fields of design, prediction of metamaterials with the required properties. The choice of these machine learning techniques is due to the inefficiency of other existing methods for the tasks of predicting properties and designing metamaterials.

In modern scientific and technical research, the methods of machine learning, namely: convolutional neural networks, occupy the most rapidly researched method in the design tasks of metamaterials and their properties. The advantage of this technique is the ease of implementation, the availability of data for this approach, the speed of calculations compared to the exact methods of predicting properties and topological structure based on physical laws. Of course, artificial neural networks are a rather complex process that has its own drawbacks - the need for a large amount of data, the relative complexity of optimization, and the complexity of problem formulation. With the development of machine learning technologies, these disadvantages are more and more eliminated, and therefore their use becomes more accessible.

Neural networks are used to design metamaterials, that is, artificial materials that have an unpredictable feedback to electromagnetic waves. The basic idea is that neural networks are used to optimize the topology and parameters of metamaterials.

First of all, the neural network is trained on a large amount of input-output data, where the input is some parameter of the metamaterial (e.g., size, shape, components) and the output is the desired response to light (e.g., absorption, reflectance, transmittance).

Once trained, the neural network can be used to predict the output data based on new input data, meaning it can predict how the metamaterial will respond to new electromagnetic waves.

Neural networks can help in the metamaterial design process because they can explore a large number of metamaterial options and efficiently optimize their parameters to achieve the desired functional response.

One of the advantages of using neural networks in the design of metamaterials is that they can account for the complex relationships between material parameters and functional response. They are able to find non-linear dependencies that can be difficult for traditional design methods.

Consequently, neural networks enable the automation and optimization of the metamaterial design process, enabling faster and more efficient development of new materials with desired properties.

The research used a range of knowledge about convolutional neural networks and the approaches used to overcome their limitations.

A large amount of information about metamaterials from relevant sources was used, namely the topology, physical composition and measurement conditions of metamaterials.

Software environments were used for writing digital code and building 3D objects of metamaterials with defined properties.

The possibility of predicting the transmission coefficient based on information about the structure, physical composition, and measurement conditions of metamaterials is analyzed. It is shown that such a possibility exists, and the main conditions imposed on data about metamaterials, which are required for the use of convolutional neural networks in solving problems of predicting information about metamaterials, are described.

An algorithm for predicting the transmission coefficient based on the structure, physical composition of metamaterials based on a convolutional neural network using experimental data of laboratory metamaterials has been developed.

It was found that data saved in ".ply" format is not suitable as input data for a neural network. The Python-based Open3D software package was used to convert the original format to the ".xyzrgb" format, which is an array where each line is a vector of six numbers; the first three are coordinates, the other three are RGB numbers. The possibility of using this type of information in convolution processes and for neural network training is shown. Conditions imposed on the data are presented: data homogeneity, which means that all parameters that affect the prediction result must either be the same or be included in the data as property information; scalability, which means that the data must be reduced to the same ranges of values and scales.

An algorithm for saving information about the physical composition of metamaterials has been developed. It is shown that using information about the electromagnetic properties of chemical elements, it is possible to predict the transmission coefficient of metamaterials. 3D objects of metamaterials are built in a digital environment with information about a chemical element attached to each pixel. The algorithm for saving this information in a form convenient for training a convolutional neural network is described. Types of digital formats that can be used to save the necessary information about metamaterials are presented. The format used to store information about the 3D object in this work was analyzed.

The process of presenting the coefficient of passage of metamaterials in a form convenient for training a convolutional neural network is described. Two methods were used to compare the effectiveness of both methods. It is shown that the method of representing experimental characteristics in the form of polynomial coefficients is faster, but not suitable for solving problems of predicting the characteristics of metamaterials.

Other formats for saving information about 3D structures were analyzed. It has been studied that such formats as ".fbx", ".obj", ".stl", ".3ds" are not suitable due to the following reasons: lack of effective software functionality (to convert to the format required for the neural network); lack of saving information about additional channels of information; the presence of redundant information that cannot be separated from the necessary information.

A convolutional neural network architecture has been developed for predicting the transmission coefficient based on the structure and physical composition of metamaterials. Prediction results are shown for both cases from the representation of the pass coefficient - in the form of an array of points and polynomial coefficients. The prediction results are analyzed and techniques are given to improve these results and optimize the network to reduce training execution time and save resources. Data augmentation is shown to be the most effective method for improving forecasting results. Nevertheless, performance improvement methods based on architecture changes and hyperparameter changes should be continually evaluated and used whenever possible.

Keywords: convolutional neural network, metamaterial design, 3D model, pass rate, dielectric materials, dielectric constant, measurement of material parameters, approximation model, experimental-analytical method, information visualization, non-linear model, intelligent computing, computer programming, dielectrics, numerical modeling, machine learning, mathematical modeling, prediction.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Праці, в яких опубліковано основні результати дисертації:

[1] M. O. Zozyuk, D. V. Koroliouk, P. I. Krysenko, A. I. Yurikov and Y. I. Yakymenko, “Prediction of characteristics using a convolutional neural network based on experimental data on the structure and composition of metamaterials,” *STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING*, vol. 11, no. 3, pp. 777–787, Jun. 2023, doi:10.19139/soic-2310-5070-1707.

[2] М.О. Зозюк та О.І. Юріков. “Використання згорткової нейронної мережі для прогнозування коефіцієнту пропускання метаматеріалів від їх структури та складу,” *Мікросистеми, Електроніка та Акустика.*, vol. 28, no. 1, pp. 271444.1-271444.10, Jul. 2023, doi:10.20535/2523-4455.mea.271444.

[3] M. O. Zoziuk, O. I. Yurikov, D. V. Koroliouk and Y. I. Yakymenko, “The Principle of Creating Quasiperiodic Surfaces under the Action of a Vibrating Dielectric Matrix,” *Microsystems, Electronics and Acoustics*, vol. 25, no. 1, 2020, pp. 5-10, Dec. 2020. doi:10.20535/2523-4455.mea.202632.

[4] P. I. Krysenko, M. O. Zoziuk, O. I. Yurikov, D. V. Koroliuk and Yu. I. Yakymenko, “Chladni Figures Simulation on a Rectangular Plate,” *Microsystems, Electronics and Acoustics*, vol. 26, no. 1, 2021, pp. 241698.1-241698.6, Dec. 2021. doi:10.20535/2523-4455.mea.241698.

[5] S.O. Dovgyi, O.I. Yurikov and M.O. Zozyuk, “On One Statistical Model of Error Rate in the Stream of Packet Data Transmission through Communication Channels,” *Cybern Syst Anal*, vol. 56, no. 5, pp. 739–744, Oct. 2020. doi:10.1007/s10559-020-00294-x.

Праці, які засвідчують апробацію матеріалів дисертації:

[1] M. O. Zozyuk, D. V. Koroliouk, V. O. Moskaliuk, A. I. Yurikov and Y. I. Yakymenko, “Creation of quasiperiodic surfaces under the action of vibrating dielectric matrices,” presented at the *2020 IEEE 40th International Conference on Electronics and*

Nanotechnology (ELNANO), Kyiv, 2020. Available: doi:10.1109/ELNANO50318.2020.9088821. Accessed on: April 22, 2020.

[2] М.О. Зозюк, Д.В. Королук, П.І. Крисенко, О.І. Юріков та Ю.І. Якименко, “Використання нейронних мереж для прогнозування властивостей метаматеріалів,” представлено на *IX Міжнародній науковій конференції імені І. І. Ляшка «Обчислювальна та прикладна математика»*, Київ, Україна, 2022.

[3] М.О. Зозюк та О.І. Юріков, “Підготовка даних для використання в системах прогнозування властивостей метаматеріалів для нейронної згорткової мережі,” представлено на *XXI Міжнародній науково – практичній конференції – Інформаційно-комунікаційні технології та сталий розвиток*, Київ, Україна, 2022.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ.....	14
ВСТУП.....	15
Розділ 1. ОГЛЯД СУЧАСНИХ ПІДХОДІВ ДО ПРОГНОЗУВАННЯ ЕЛЕКТРОМАГНІТНИХ ВЛАСТИВОСТЕЙ МЕТАМАТЕРІАЛІВ.....	21
1.1 Використання методів глибокого навчання для дизайну метаматеріалі....	23
1.2 Складнощі у підготуванні даних із застосуванням нейронних мереж.....	23
1.3 Прогнозування електромагнітних характеристик метаматеріалів за допомогою згорткової нейронної мережі.....	25
1.4 Особливості побудови нейронної мережі для прогнозування властивостей метаматеріалів.....	28
1.5 Шляхи розв’язання задач, пов’язаних з набором даних для навчання нейронної мережі.....	33
Висновки до розділу 1.....	36
Розділ 2. ВИКОРИСТАННЯ ЗГОРТКОВОЇ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ АНАЛІЗУ ДАНИХ.....	38
2.1 Підходи до побудови архітектури нейронних мереж.....	38
2.2 Використання операції згортки в нейронних мережах для можливості прогнозування властивостей 2D та 3D об’єктів.....	46
2.3 Застосування функції об’єднання для визначення властивостей вибірки...	52
2.4 Особливості використання згорткової нейронної мережі.....	58
2.4.1 Варіанти основної функції згортки.....	59
2.4.2 Структурований вихід нейронної мережі та типи даних.....	72

2.4.3 Ефективні алгоритми згортки.....	76
2.4.4 Випадкові ознаки або ознаки без супервізора.....	77
Висновки до розділу 2.....	79
Розділ 3. ПІДГОТОВКА 3D-СТРУКТУР ТА ДАНИХ ДЛЯ НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ.....	81
3.1 Методи побудови 3D структур та складнощі конвертації форматів.....	81
3.2 Реалізація властивостей при створенні цифрових 3D-структур.....	87
3.3 Відтворення характеристик для використання у нейронній мережі.....	98
3.4 Особливості вибору моделей цифрових даних.....	102
Висновки до розділу 3.....	104
Розділ 4. ПРОГНОЗУВАННЯ ВЛАСТИВОСТЕЙ МЕТАМАТЕРІАЛІВ НА ОСНОВІ ЗГОРТКОВОЇ НЕЙРОННОЇ МЕРЕЖІ	106
4.1 Варіативність у виборі параметрів системи та аргументація у їх виборі..	106
4.2 Застосування техніки навчання згорткових мереж для прогнозування властивостей метаматеріалів.....	108
4.3 Тестування нейронної мережі.....	115
4.4 Оптимізація якості прогнозування властивостей метаматеріалів.....	125
Висновки до розділу 4.....	130
ОСНОВНІ НАУКОВІ ТА ПРАКТИЧНІ РЕЗУЛЬТАТИ І ВИСНОВКИ.....	131
СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....	133
ДОДАТКИ.....	153
ДОДАТОК А. Список публікацій здобувача за темою дисертації.....	153

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

CNN – згорткова нейронна мережа (Convolutional neural network)

GNN – графова нейронна мережа (Graph neural network)

RNN – рекурентна нейронна мережа (Recurrent neural network)

FDTD – метод скінченних різниць у часовій області (Finite-difference time-domain method)

RCWA – строгий аналіз пов'язаних хвиль (Rigorous couple-wave analysis)

FEM – метод скінченних елементів (Finite element method)

NURBS – неоднорідний раціональний B-сплайн (Non-uniform rational B-spline)

API – прикладний програмний інтерфейс (Application programming interface)

ASCII – Американський стандартний код для інформаційного обміну (American Standard Code for Information Interchange)

RGB – адитивна колірна модель (Red, Green, Blue)

MLP – багатошаровий нейрон (Multilayer perceptron)

RBM – обмежена машина Больцмана (Restricted Boltzmann machine)

PCA - метод основних компонент (Principal component analysis)

SVM - метод опорних векторів (Support vector machine)

ВСТУП

Актуальність

В сучасному світі потрібні матеріали з властивостями, які неможливо отримати з використанням природних матеріалів. З розвитком композиційних матеріалів та сплавів збільшилась кількість матеріалів з особливими властивостями. Науковці знайшли закономірності при створенні різних композитів та сплавів, за допомогою яких можливо було б розраховувати властивості нових матеріалів.

Метаматеріали є наступним кроком у розвитку штучних матеріалів. Метаматеріали – це штучні структури, властивості яких обумовлені саме топологічною періодичною структурою матеріалів в різному масштабі. Простота у створенні такого типу матеріалів є великою перевагою. В залежності, які властивості потрібно отримати при створенні таких матеріалів, буде вирішуватися які фізико-математичні моделі використовувати. Існує багато моделей – електромагнітних, акустичних, електромагнітних і т. д., в залежності від того яким параметром потрібно керувати. Розв’язання рівнянь цих моделей займає велику кількість ресурсів, що призводить до пошуку інших методів розв’язку цих рівнянь, або взагалі інших підходів до прогнозування характеристик метаматеріалів. Методи глибокого навчання можуть бути використані для прогнозування та аналізу моделей розрахунку параметрів метаматеріалів [2, 4].

Одним із таких методів є застосування нейронних мереж, які були спеціально розроблені для розв’язку різноманітних задач. Підходи до оптимізації нейронних мереж, види архітектур, покращення результатів на основі корегування гіперпараметрів розроблені у [100].

Окремим типом нейронних мереж є згорткова нейромережа, яка призначена для розрахунку моделей з дуже великою кількістю параметрів. В згортковій мережі використовується операція згортки для того, щоб зменшити розмірність моделі, але водночас зберегти всю інформацію про залежність між параметрами.

В останні роки були проведені дослідження з використання згорткових нейронних мереж для прогнозування властивостей метаматеріалів [13-16], [24-27]. Використовуються, як правило, дані про двовимірну топологію метаматеріалів та відповідні експериментальні характеристики цих метаматеріалів. В деяких випадках [88], [89] використовувалися тільки часткові дані про інші виміри даних про структуру або умови вимірювання – товщина метаматеріалів, радіус наночастинок і т. д.

Перевага використання нейронних мереж для прогнозування експериментальних характеристик метаматеріалів полягає в тому, що результат отримується набагато швидше в порівнянні з фізико-математичними методами розрахунку всіх потрібних параметрів метаматеріалів.

Звичайно, існує багато обмежень нейронних мереж та умов, яких потрібно дотриматися для того, щоб отримати результат – однорідність даних, однаковість умов вимірювання, достатній об'єм даних.

З розвитком області дослідження метаматеріалів з'явилась велика кількість експериментальних досліджень метаматеріалів [90-94], які можна проаналізувати з використанням згорткових мереж.

Актуальність і новизна дисертаційного дослідження обумовлені відсутністю комплексних підходів до постановки задачі з прогнозування характеристик метаматеріалів на основі топологічної структури та складу. Необхідність врахування комплексної топологічної структури та складу метаматеріалів є новим інструментом дослідження саме цих причинно-наслідкових зв'язків. Також, потрібно дослідити обмеження у застосуванні згорткових мереж, як додатковий засіб дизайну метаматеріалів з наперед заданими властивостями.

Стан розробки даної тематики у вітчизняній та закордонній науці.

Дизайн метаматеріалів є нагальною потребою створення нових матеріалів для застосування в широкому спектрі інженерних пристроїв. Зазвичай, в літературі

представлені фізико-математичні моделі для проєктування метаматеріалів в залежності від типу потрібних властивостей.

Використання методів глибокого навчання потребує спеціального підходу до приготування даних в належному вигляді для використання нейронною мережею.

На даний час ця область стрімко розвивається у зв'язку з відкриттям нових можливостей глибокого навчання, а саме згорткових нейронних мереж [78-87].

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

Дисертаційна робота виконана на кафедрі мікроелектроніки.

Мета та завдання дослідження.

Мета – розробити алгоритм прогнозування властивостей метаматеріалів в залежності від топологічної структури, складу та встановити можливість використання згорткових нейронних мереж для прогнозування властивостей.

Для успішного досягнення мети роботи потрібно було розв'язати такі завдання:

- 1) Розробити метод прив'язки інформації про склад метаматеріалу до топологічної структури метаматеріалу.
- 2) Встановити потрібну форму представлення інформації про структуру та склад метаматеріалів для ефективного використання в згорткових нейронних мережах.
- 3) Сформулювати та оптимізувати архітектуру згорткової нейронної мережі для прогнозування параметрів метаматеріалу.
- 4) Спрогнозувати властивості метаматеріалів за допомогою згорткової нейронної мережі.
- 5) Встановити можливість використання розробленого підходу до прогнозування властивостей метаматеріалів в залежності від інформації про метаматеріали.

Об'єкт дослідження – метаматеріали та їх властивості.

Предмет дослідження – методика підготовки інформації про метаматеріали для прогнозування їх властивостей за допомогою згорткової нейронної мережі.

Методи дослідження:

Були використані програмні методи з формування інформації про структуру та склад метаматеріалів, а саме програмні середовища Blender, PyCharm.

Використані бібліотеки Python – NumPy, Pandas, PyTorch для реалізації архітектури згорткової нейронної мережі.

Наукова та практична новизна отриманих результатів.

В роботі:

- 1) Розроблена тензорна форма запису даних про метаматеріали, зручна для глибокого навчання згортокових нейронних мереж.
- 2) Розроблено метод прив'язки інформації про параметри метаматеріалів до їх просторового розташування.
- 3) Сформовано архітектуру згорткової нейронної мережі, яка використовує інформацію про топологічну структуру та склад метаматеріалів.
- 4) Встановлено обмеження, які мають накладатися на дані про метаматеріали, при їх використанні в системах з прогнозування властивостей на основі згортокових нейронних мереж.

Особистий внесок здобувача.

Разом з науковим керівником було обрано напрямок, об'єкт і предмет дослідження, сформульовані мета і завдання роботи, а також обговорено та інтерпретовано отримані в ході дослідження результати. Запропонована можливість створення систем з прогнозування, які використовують комплексну інформацію про метаматеріали. Представлено основні ідеї щодо розв'язанні поставлених задач. У роботах – [98*], [153*] опублікованих у співавторстві, здобувачем було розроблено алгоритм з прогнозування властивостей на основі топологічної структури та складу метаматеріалів, представлено методику формування інформації про топологічну

структуру, склад метаматеріалів та підхід до представлення цієї інформації в потрібному для систем глибокого навчання на основі згорткових нейронних мереж. Проаналізована якість отриманих результатів та описані методи покращення алгоритму з прогнозування експериментальних характеристик. Дисертантом написано тексти статей та основні розділи у публікаціях [98*], [153*].

*джерела зі списку публікацій (Додаток А).

Апробація результатів дисертації.

Результати отримані в ході роботи над дисертацією були представлені на 3 наукових і науково-практичних конференціях і семінарах:

1) *2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO)*, 2020.

2) IX Міжнародна наукова конференція імені І. І. Ляшка «Обчислювальна та прикладна математика», присвячена 100-річчю академіка Івана Івановича Ляшка, 10.10.2022.

3) XXI Міжнародної науково – практичної конференції Інформаційно-комунікаційні технології та сталий розвиток, Київ, 14 – 16 серпня 2022.

Публікації.

За результатами досліджень опубліковано 8 наукових праць, з них:

- 5 статей у яких висвітлено основні матеріали дисертації, дві з яких у виданні, віднесеному до третього квартилю (Q3) та (Q2) відповідно до класифікації SCImago Journal and Country Rank або Journal Citations Reports, чи одноосібних монографій, що відповідають зазначеним вимогам, інша у періодичних наукових фахових виданнях категорії Б,

- 3 доповідей та тез на наукових конференціях

Структура та обсяг дисертації.

Дисертація складається із вступу, 4 розділів із висновками, загальних висновків, списку використаних літературних джерел зі 155 найменувань на 20 сторінках, 1 додатку. Загальний обсяг дисертації складає 155 сторінки, з яких основний зміст викладений на 107 сторінках, містить 52 рисунки.

РОЗДІЛ 1.

ОГЛЯД СУЧАСНИХ ПІДХОДІВ ДО ПРОГНОЗУВАННЯ ВЛАСТИВОСТЕЙ МЕТАМАТЕРІАЛІВ

1.1 Використання методів глибокого навчання для дизайну метаматеріалів

Метаматеріали – це штучні, переважно періодичні структури, які побудовані, для досягнення потрібних фізичних характеристик, які, як правило, не спостерігаються у природі. Розрізняють метаматеріали в залежності від того, з якими хвильовими збуреннями вони взаємодіють. Бувають метаматеріали, які реагують на види збурень таких як: акустичні, електромагнітні, хімічні та інші.

Для моделювання характеристик метаматеріалів використовують різні теоретичні та експериментальні підходи. Теоретичні беруть за основу наявну модель в залежності від досліджуваних властивостей. Так, для прикладу, акустичні властивості досягаються маніпулюванням модулем пружності, густиною та хіральністю [1]. Електромагнітні властивості, а саме магнітна сприйнятливість та діелектрична проникність, моделюються на основі моделей Максвелла, Лоренца, Дебая, Друде тощо [2].

Також дуже важливими є експериментальні дослідження метаматеріалів, де перевіряються теоретичні передбачення і робляться висновки про характеристики і їх залежності від деяких параметрів [3, 4, 5, 6, 7, 8].

З розвитком технологій глибокого навчання, підходи до аналізу матеріалів стали набагато розвиненішими [9]. Метаматеріали стали частиною цього процесу [10], [11], [12].

Дизайн метаматеріалів відрізняється від дизайну звичайних матеріалів тим, що метаматеріали мають штучно утворену періодичну структуру. Для ефективного

дизайну метаматеріалів потрібна якісна модель прогнозування характеристик шуканої структури. Для таких задач є різні підходи – теоретичні, емпіричні. Теоретичні – це математичні моделі та наближене моделювання. Емпіричні – це створення достатньої експериментальної бази для того, щоб побудувати зручну модель прогнозування. Одним із підходів для вирішення завдання з дизайну метаматеріалів є визначення властивостей однієї комірки, а потім наближене моделювання характеристик всієї структури. Для прогнозування акустичних властивостей було проведено низку досліджень [13-16]. Також існують і інші методи [17-23]. Дизайн електромагнітних метаматеріалів будується теоретичним чином, на основі відомих моделей [24].

Методи глибокого навчання для всіх видів метаматеріалів є потужним засобом для дизайну метаматеріалів і використовуються для всіх видів метаматеріалів [25], [26], [27]. В цій роботі було застосовано один із методів глибокого навчання, а саме згорткова нейронна мережа. Переваги наближеного моделювання характеристик метаматеріалів полягають у тому, що точне моделювання має велику складність і використовує велику кількість часу та ресурсів.

Методи глибокого навчання, а саме нейронні мережі, мали інтенсивний розвиток в останній час [28], [29], [30]. Різноманітність технік використання нейронних мереж для прогнозування характеристик метаматеріалів зростає до величезної кількості, бо нейронні мережі самі по собі є дуже гнучким засобом для розв'язання різних задач (класифікація, прогнозування і т. д.). Такі підходи сильно різняться, і як правило різниця полягає в типах даних, які є для навчання, архітектурах, умовах, а також характеристиках, які прийняті за основу досліджень. Так, акустичні властивості були досліджені в працях – [31-46]. Можливості нейронних мереж в дослідженні електромагнітних властивостей та характеристик були продемонстровані в [47-51]. На деяку кількість робіт, в області досліджень електромагнітних характеристик, будуть посилання протягом всієї дисертації.

Варто зазначити, що не тільки акустичні та електромагнітні властивості привернули увагу спеціалістів з глибокого навчання. Велика кількість робіт була опублікована у дослідженні хімічних властивостей метаматеріалів та звичайних матеріалів [52-58]. Ці роботи продемонстрували можливості нейронних мереж з прогнозування, аналізу і дизайну молекул та структур на молекулярному та атомарному рівні.

1.2 Складнощі у підготуванні даних із застосуванням нейронних мереж

В процесі постановки задачі, одна із найбільш важковирішувальних проблем є підбір правильних даних. Коректний вибір даних має бути на основі умов, які відповідають конкретній фізичній реальності. Це означає, що для того, щоб використовувати результати та можливості нейронної мережі, потрібно, щоб ці результати були якомога більше ближчими до реальних залежностей та причинно-наслідкових зв'язків. Нейронна мережа, як і будь-яка інша система наукового аналізу, працює з деякими похибками.

Наприклад, коли є тотожність: вхідні дані – вихідні дані; важливо розуміти - чи врахувала модель всі істотні чинники, які впливають на результат? Розв'язання даного питання може бути реалізовано двома шляхами – збільшенням кількості даних або збільшенням чинників, які впливають на залежність, включаючи їх в дані. Все залежить від того, які можливості є у дослідника – кількість даних та всі інші параметри для коректного аналізу моделі.

Якщо піти першим шляхом, то причинно-наслідкові зв'язки не включені в тип даних, а включені у взаємні відмінності і схожості між даними. Ці відмінності та схожості можуть бути враховані – вони враховуються нейронною мережею, яка навчається в них розпізнавати, і чим більше даних, тим краще нейронна мережа справляється з цією задачею. Як правило, цей шлях використовується, коли є

достатня кількість даних, і підготовка більш фізично реальних даних є проблематичною.

Інший шлях передбачає підготовку даних, які є дуже наближеними до реальності але, їх кількість є обмеженою у зв'язку з різними причинами – складністю, кількістю часу, потужністю і т. д. Це означає, що всі параметри, які впливають на вихідний результат мають бути враховані - причинно-наслідкові зв'язки включені в тип даних. Інша складність виникає при самому розв'язанні даної задачі – включення додаткових даних, які впливають на вихідний результат, не завжди є простим процесом. Це також потрібно враховувати при розгляді інших не прямих чинників на причинно-наслідкові зв'язки.

Насправді в реальності немає чітко визначеного алгоритму – піти шляхом набору великої кількості даних чи шляхом якості даних. В реальних задачах, це завжди компроміс і знаходження золотієї середини. Розв'язок кожної задачі є унікальним, і тому немає чітко визначеної теорії, на цей час. Існують тільки більш технічні методики глибокого навчання [59] та інші [60, 61, 62, 63], де дослідникам пропонуються способи з покращення результатів нейронних мереж в залежності від типу задачі (прогнозування, класифікація і так далі) та типу даних (зображення, 3D-об'єкти, акустичні дані та інші типи).

Оскільки кожна задача є унікальною, то відповідальність за набір даних повністю лежить на дослідниках. В області глибокого навчання кожна задача вирішується частково емпіричним підходом, і підготовка даних не є виключенням. Не автоматизована підготовка даних є в деяких випадках виключно складною задачею, особливо якщо дані не враховують весь спектр впливів на вихідний результат, тобто залежність між вхідними та вихідними даними отримується шляхом збільшення кількості даних, а не в залежності від інформативності даних. В літературі було запропоновано немало кількість методів для автоматичного підбору даних для нейронних мереж – [64-67].

1.3 Прогнозування електромагнітних характеристик метаматеріалів за допомогою згорткової нейронної мережі.

Область дизайну метаматеріалів за допомогою глибокого навчання була добре досліджена в останні роки. Запропоновані різноманітні техніки застосування, як звичайних нейронних мереж, так і згорткових [68-71]. В цілому більшість робіт була запропонована для метаповерхонь, або метаматеріалів, в яких вхідні дані були представлені у вигляді двовимірних зображень, де третя координата або була представлена у вигляді яскравості пікселя двовимірного зображення, або у вигляді ще одного каналу на вході нейронної мережі.

Є можливість виділити основні особливості прогнозування електромагнітних характеристик метаматеріалів за допомогою нейронних мереж.

Оскільки, цифрова симуляція характеристик метаматеріалів є громіздкою, то елемент бази даних для навчання нейронної мережі має бути простим, щоб була можливість підготувати велику базу даних з відповідними характеристиками.

Симуляція двовимірної метаповерхні є оптимальним варіантом для підготовки бази даних. Можливість симуляції тривимірного метаматеріалу, для задач глибокого навчання, не є легкою задачею - [72], [74]. Є можливість представити інформацію про об'єм, як двовимірне зображення, або як додатковий канал кольору або яскравості пікселя зображення [73].

Є можливість розв'язання проблеми з частковою відсутністю даних шляхом спрощення представлення даних. Наприклад, була запропонована ідея, в якій були симульовані сферичні наночастинки (метаатоми), які складались з шарів різної товщини, властивості яких і були вхідними даними нейронної мережі [75]. Симуляція властивостей таких сфер (метаатомів) є набагато простішою задачею.

Схожі задачі ставились авторами робіт [76], [77]. Там були змодельовані метаматеріали, представлені у вигляді шарів матеріалів, з їхніми товщинами та властивостями відповідно.

В обох випадках (сфери або площини) можна генерувати метаматеріали з потрібними характеристиками, але вони обмежені складністю поверхні структур, які можна генерувати за допомогою навчених нейромереж.

Зазначимо як вирішуються задачі з прогнозування електромагнітних властивостей метаматеріалів, структури яких представлені у вигляді двовимірних зображень. Симуляція характеристик метаповерхонь не є складною задачею порівняно з об'ємними структурами. Ця категорія досліджень ділиться на дві – в першій [78-81]: використовуються операція згортки (*CNN*), в другій [82-87] нейронні мережі без операції згортки. Згорткова нейромережа потрібна для тих випадків, коли двовимірне зображення представлене великою кількістю пікселів, і растрезація (процес перетворення векторного зображення у растрове) не є ефективним варіантом, де одна із причин є втрати інформації про структуру. Згорткова нейромережа знижує кількість пікселів, тим самим робить можливим зменшення кількості вхідних нейронів, і одночасно зберігає інформацію про структуру. Класична схема роботи згортки для нейромережі представлена на рис. 1.1. Згорткові нейронні мережі також можуть бути застосовані для дизайну молекул та кристалічних структур, де структура представлена у вигляді деякого відображення 3D-структури – [88], [89].

В цілому тенденція така – підготовка даних для навчання нейронних мереж є складною задачею, наприклад симуляція фізичних властивостей великих структур. Така симуляція є затратним процесом. Якщо використовувати наближені методи, то зі збільшенням структур точність падає, що призведе, до ще більшого нагромадження похибки.

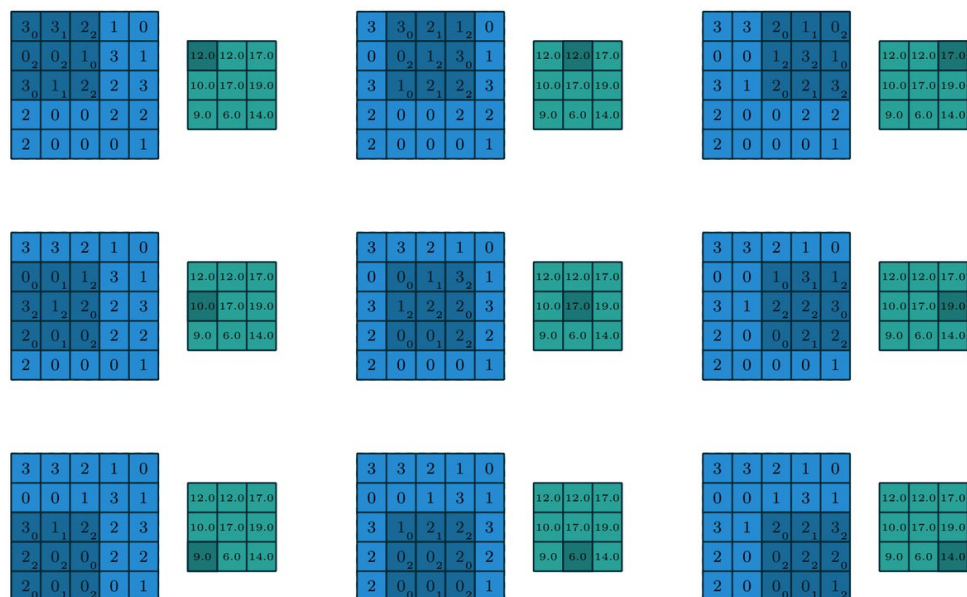


Рисунок 1.1 – Кожен синій піксель представляє число, яке несе інформацію про яскравість або інший параметр зображення (прозорість, насиченість, колір і т. д.).

Кожен темно-зелений є згорткою дев'яти попередніх темно-синіх пікселів.

Як зазначалось, вирішення таких складностей полягає в тому, щоб симулювати не великі структури (метаповерхні), в яких третя координата може бути записана у двовимірному форматі у вигляді третього або більшого числа каналів. Це можуть бути метаатоми, метаповерхні у невеликому розширенні зображення і т. д. Інший підхід створення структур, «не складними» у своєму дизайні елементами - шари матеріалів, невеликі структури і т. д. Також можна апроксимувати властивості великих структур, складених з малих сукупностей, властивості яких уже симульовані і які можна з локального плану розповсюдити в глобальний і таким чином симулювати характеристики великої структури. Всі вище зазначені роботи [78-87] в тій чи іншій степені реалізують такі підходи.

З приводу представлення графіків характеристик (або спектрів і т. д.), то це задача, яка є порівняно простою. Це може бути реалізовано різними методами – треба тільки зробити відбір ключових точок на графіку, які характеризують

«поведінку» параметра; конвертувати графік в поліноміальні коефіцієнти; представити графік у вигляді функції.

Наступним критерієм якості даних є їх однорідність. Вони мають формуватися при одних умовах і при тих самих параметрах отримання, а параметри, які не враховані, мають бути включені в задачу для прогнозування. Параметри, які не впливають на нейронну мережу, мають бути постійними для всіх даних. Це не є практично досяжною задачею, але вплив параметрів, які впливають на характеристики, мають бути зведені до мінімуму.

1.4 Особливості побудови нейронної мережі для прогнозування властивостей метаматеріалів

На побудову нейронної мережі для прогнозування електромагнітних характеристик метаматеріалів впливають такі основні чинники – вигляд даних і їх найкраще представлення. Чим більше параметрів (інформації) даних, тим більше вхідних нейронів мережі. Велика кількість нейронів призводить до збільшення часу обчислень. Якщо балансувати кількість вхідних нейронів з кількістю нейронів в інших шарах, то це призведе до зниження здатності розрізняти залежності між даними. Як правило, для задачі прогнозування властивостей від структури та інших чинників використовують повно зв'язну нейронну мережу – кожен нейрон пов'язаний з кожним нейроном в наступному і попередньому шарі.

Потрібно розуміти, як розв'язати задачу з правильністю формату даних для нейронної мережі. Якщо дані про метаматеріали займають великий об'єм, то потрібно підключати наближені методи представлення – згорткові нейронні мережі та інші методи апроксимації. Доведено, що один шар нейронної мережі є універсальним апроксиматором будь-якої функції [86]. Якщо електромагнітні властивості мають певну залежність від структури, і є розуміння, що цю залежність можна представити однією функцією, то можна припустити, що для знаходження

цієї залежності (у вигляді шару нейронної мережі, як апроксимативну функцію цієї залежності), потрібен один шар нейронної мережі. Якщо дані несуть інформацію про додаткову залежність, наприклад, у вигляді додаткових фізичних параметрів, які можуть впливати на вихідну електромагнітну характеристику, то є ймовірність, що одна функція (один шар) не буде здатен описати всі залежності одночасно і т. д. Обов'язок дослідника емпірично та на основі аналізу дізнатись, яка кількість шарів нейронної мережі потрібна, щоб охопити всі функціональні залежності.

Стосовно інших параметрів (швидкість навчання, величина вибірки для методу градієнтного спуску), то вони визначаються технічною складовою задачі – який об'єм часу є для обчислень, кількість даних (пам'ять яку вони займають) і т. д.

Ще один параметр, який потрібно визначити – це кількість нейронів в кожному шарі. Потрібно розуміти одне – чим більше нейронів, тим складніша функціональна залежність може бути апроксимована. Кількість вхідних та вихідних нейронів обирається дослідником в першу чергу (на основі вхідних даних та вихідних). Після цього формується розуміння скільки потрібно нейронів у кожному внутрішньому шарі. Як правило, кількість внутрішніх нейронів лежить в межах між кількістю вхідних і вихідних нейронів. Часто бувають винятки, і щоб апроксимувати залежність, потрібна велика кількість нейронів, а якщо ні, то кількість проміжних нейронів краще зменшити, щоб не використовувати додаткову кількість енергії та часу, а також не спотворити навчання інших шарів.

Важливо, що для різного типу даних можуть використовуватись різні типи нейронних мереж – рис. 1.2, рис. 1.3, рис. 1.4.

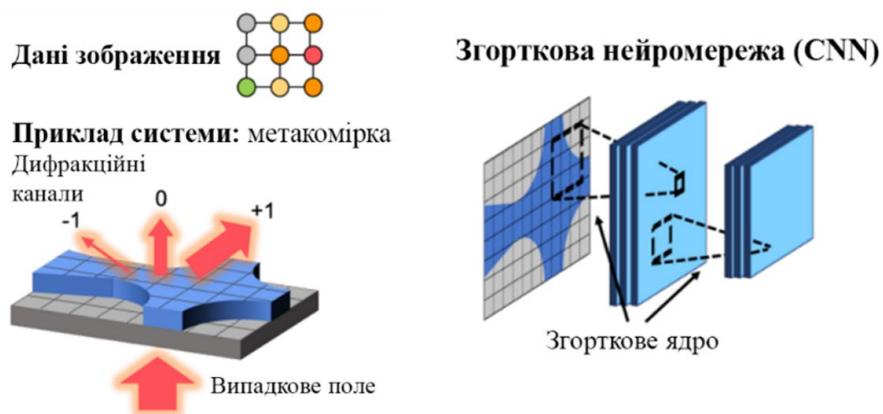


Рисунок 1.2 - Структури даних зображення добре підходять для представлення макетів пристроїв довільної форми, таких як криволінійні наноструктури.

Зображення зазвичай обробляються за допомогою згорткових шарів у згортковій нейронній мережі (*CNN*), яка використовує невеликі дво- та тривимірні ядра для перетворення зображень у представленні функцій.

Функціональну залежність можна наближено знайти – фізичні теорії, апроксимативні методи і т. д. Поставивши емпірично початкову кількість нейронів, можна оцінити кількість нейронів, яка потрібна для інших шарів.

Майже завжди емпіричний метод буде присутній, але початкове розуміння, а отже і початкові умови, з яких почнеться емпіричне дослідження, особливо потрібно, враховуючи складність і кількість функціональних залежностей, які можуть бути присутні в будь-якій задачі.

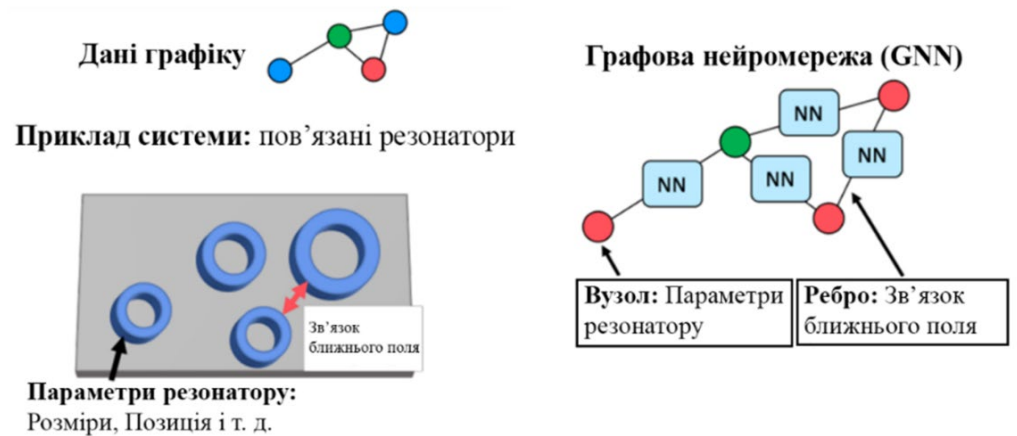


Рисунок 1.3 - Графові структури даних добре підходять для представлення фотонних систем із співзалежними частинами, такими як наноструктури, з'єднані в ближньому полі. Графи обробляються в графовій нейронній мережі (*GNN*), яка може узагальнювати електромагнітну реакцію структури на основі її атрибутів вузла та краю.

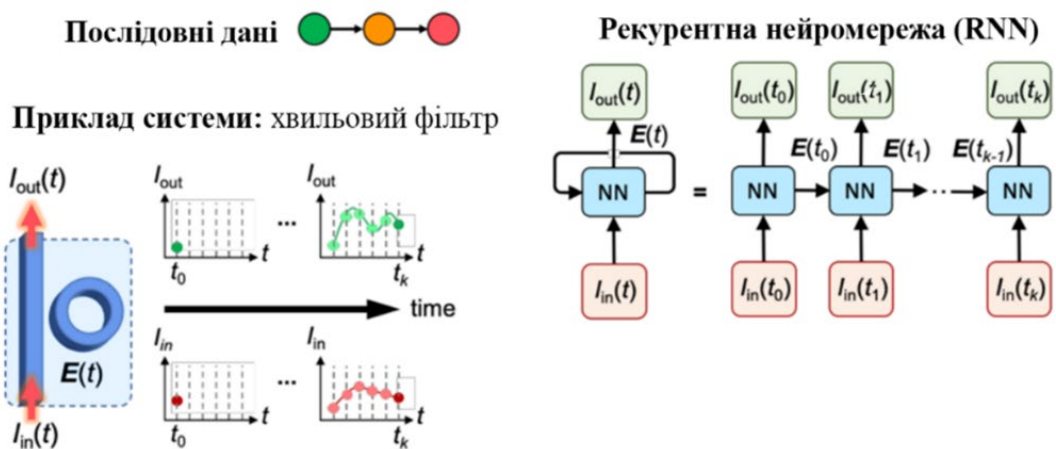
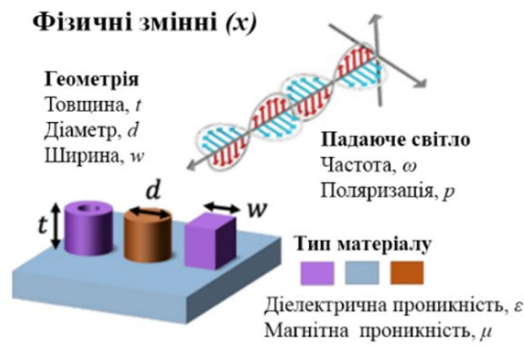


Рисунок 1.4 - Послідовні в часі явища можна зафіксувати за допомогою рекурентної нейронної мережі (*RNN*).

У наведеному тут прикладі хвильового фільтра вихід мережі I_{out} є функцією вхідної інтенсивності I_{in} і внутрішнього стану мережі, яка вивчає $E(t)$. Представлення *RNN*, розгорнутого в часі, показує, як внутрішній стан мережі оновлюється після кожного часового кроку. На рис. 1.5, рис. 1.6, 1.7, показано огляд підходів глибокого навчання для задач фотоніки.

Параметри пристрою (x, y)



Фізичні відгуки (y)

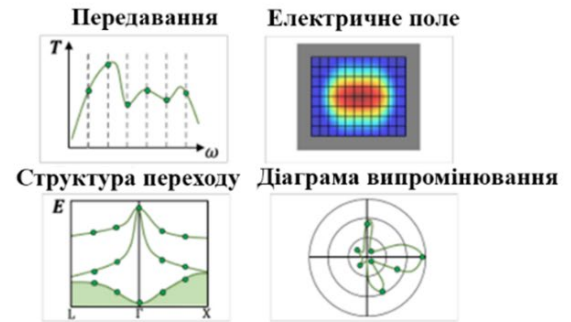


Рисунок 1.5 - Огляд глибокого навчання для фотоніки. Фотонні пристрої описуються двома типами інформації - фізичними змінними x і фізичними відгуками y .

Дискримінаційна нейромережа: $f: x \rightarrow y$

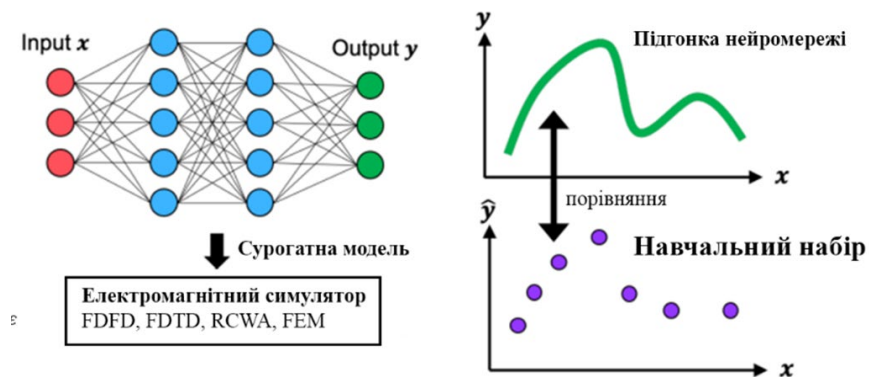


Рисунок 1.6 - Дискримінаційні нейронні мережі можуть служити сурогатними фізичними моделями. Навчена мережа моделює залежність $y=f(x)$, яка збігається з дискретними значеннями $(x, \hat{y})$ у навчальному наборі.

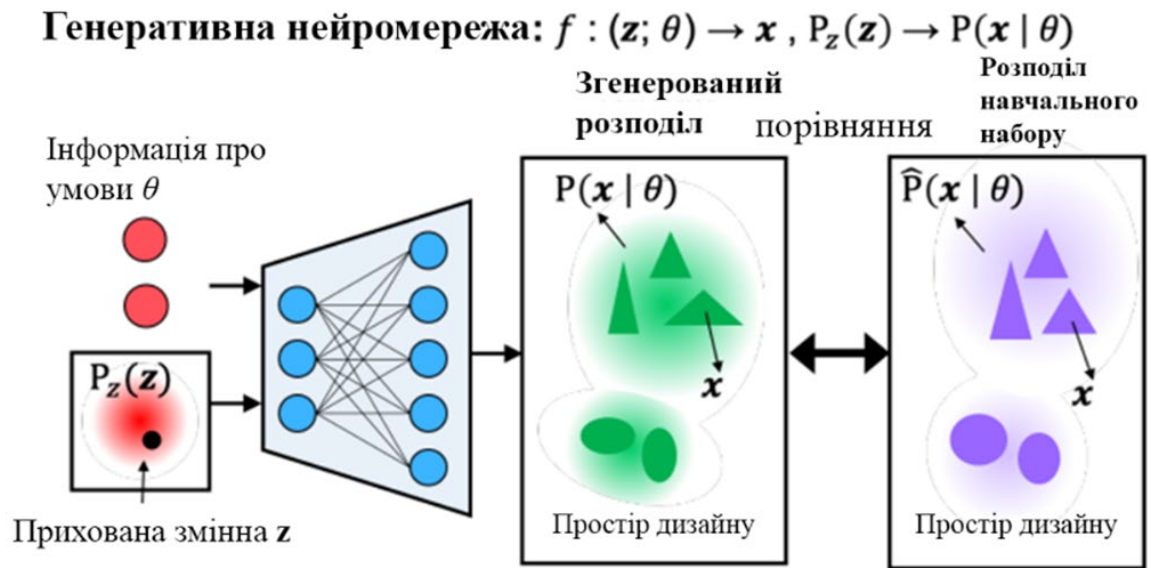


Рисунок 1.7 - Генеративні нейронні мережі відображають приховані змінні, z , та інформація про умови, θ , на розподіл згенерованих пристроїв, $\hat{P}(x|\theta)$. Метод скінченних різниць у частотній області. *FDTD*: Метод скінченних різниць у тимчасовій області; *RCWA*: строгий аналіз пов'язаних хвиль; *FEM*: метод скінченних елементів.

1.5 Шляхи розв'язання задач, пов'язаних з набором даних для навчання нейронної мережі

Існує велика кількість підходів до розв'язання задачі з набором потрібних даних для нейронних мереж. Всі вони мають свої переваги та недоліки. Різниця лежить в площині таких величин – ресурси (насамперед, час) на підготовку даних; близькість даних відносно фізичної реальності; складність у створенні відповідної нейронної мережі та відповідних прилеглих систем для таких даних та мережі.

В залежності від задачі і обставин в яких вона поставлена, кількість даних, яка потрібна, вигляд цих даних і т. д., а відповідно і підготовка цих даних можуть бути абсолютно різними.

Так дані можуть будуватись двома шляхами, або третім - суміш цих двох.

Дані можуть готуватись власноруч – тобто, кожна функціональна (яка несе залежність) комірка готується окремо. Наприклад, найпростіша задача про класифікацію рукописних чисел від одного до десяти, побудована саме таким чином (кожне число рукописне) – рис. 1.8 [91].

Інший шлях – це отримання даних змодельованим чином. Тобто, дані є апроксимативним відображенням реальних характеристик фізичних систем. Це можуть бути характеристики, отримані методами глибокого навчання функціональних залежностей теоретичних моделей.

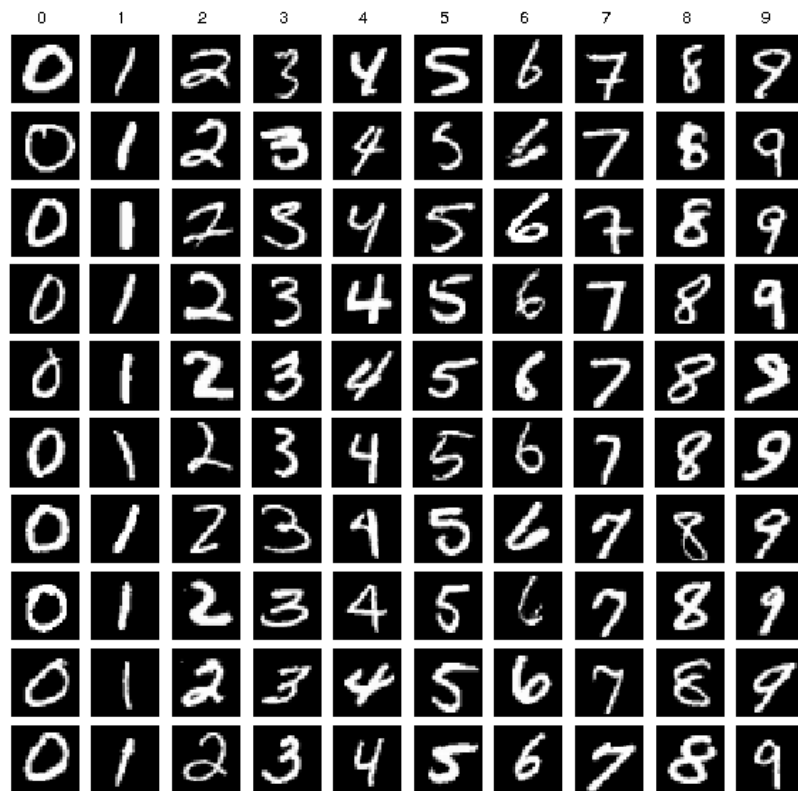


Рисунок 1.8 – Набір рукописних чисел десяти різних людей [91].

Щодо гібрида цих двох підходів (теоретичний та експериментальний), то така комбінація є сумішшю перших двох. Наприклад, дані можуть формуватись з експериментальних даних та даних, які отримані симуляцією реальної фізичної системи. Як вже зазначалось, технології глибокого навчання розвинулись глибоко і їх використання обмежується лише уявою дослідника [63]. Тому тепер на передній

план ставиться питання про те, як дані підготувати, які саме, і яка мета стоїть перед дослідниками. Всі обмеження, які накладені на процес підготовки даних, вже зазначались – складність задачі, можливість підготовки, логічність даних щодо до поставлених задач.

У підборі даних для нейронної мережі в області досліджень електромагнітних властивостей метаматеріалів, більшу частину – 95%, містить симуляція фізичних систем, тоді як кількість експериментальних досліджень метаматеріалів не така велика. Причина цього – не доскональність технологічного процесу створення метаматеріалів та витратність виробництва потрібної кількості метаматеріалів (або модифікація наявних). Проблема може бути вирішена, як зазначалось в першому розділі, виробництвом простих елементів метаматеріалів, властивості, яких потім розповсюджуються на всю площину чи об'єм для отримання характеристик складних систем. Такий метод буде гібридним.

Формуючи дані з експериментальних досліджень, є виграш в правдивості, але програш у складності постійного дотримування однакових умов вимірювання. З іншої сторони, простота симуляції даних є великою перевагою, але слабкість у правдивості та можливість похибки, яка нарастає при навчанні, є серйозним недоліком.

Кожна задача є унікальною, і тому дослідники мають підходити до кожної задачі окремо, не опираючись на якийсь конкретний алгоритм.

Висновки до розділу 1

В розділі було представлено стан розвитку області прогнозування властивостей (електромагнітних, акустичних та ін.) метаматеріалів на основі інформації про метаматеріал.

Було пояснено, що у зв'язку зі складністю симуляції та моделювання фізичних систем, використовують спрощені методи, які призначені для того, щоб підготувати дані для нейронних мереж. Дизайн метаматеріалів базується на прогнозуванні їх характеристик в залежності від структури та властивостей складових компонентів.

Представлені роботи з дослідження структур і характеристик різного типу метаматеріалів – акустичних, електромагнітних та ін. Методи глибокого навчання були застосовані багатьма дослідниками метаматеріалів, запобігаючи складності симуляції фізичних систем.

Відмічено, що в цій області є складнощі у підготованні даних для нейронних мереж. Запропоновано методи зі знаходження балансу між складністю функціональної залежності даних для нейронної мережі та кількістю реальних даних, які можуть впливати на цю залежність.

У зв'язку з потужним розвитком технологій глибокого навчання, використання цих методів не є технічно складною задачею. Набагато складнішою є задача з підготовки всіх даних та їх коректного представлення.

Описано основні тенденції, які набувають серйозних застосувань в області дизайну та прогнозування властивостей метаматеріалів за допомогою методів глибокого навчання, насамперед нейронних мереж. Представлено ідеї, які досліджувались в цій області, і показано як вони представляють нові методи зі створення унікальних метаматеріалів, за допомогою чисельного симулювання.

Показано, які є способи розв'язання типових питань, що виникають при створенні нейронних мереж для задач дизайну та прогнозування властивостей метаматеріалів. Всі основні складові та параметри не завжди мають точний алгоритм підбору, але початкові умови, при яких вони формуються, можна знайти наближеними методами. Отже, конфігурація мережі і вибір параметрів повністю залежить від самих даних та їх представлення і під них формується нейронна мережа. Технологічний аспект реалізації системи глибокого навчання не є складним питанням, порівняно з питанням про підготовку якісних даних для цих систем.

Описано шляхи формування даних для нейронних мереж, їх переваги та недоліки. Зауважено, що саме гібридний шлях є найбільш використовуваний в практичних проєктах. Складність та коштовність у підготовці експериментальних досліджень та неточність у змодельованих характеристиках є чинниками, що балансують гібридні дані.

Виняткова унікальність задач та ситуацій в області дизайну та прогнозування електромагнітних властивостей метаматеріалів призвели до складності побудови чітких алгоритмів та систем, які покликані полегшити дослідження цієї області. У зв'язку зі швидким розвитком методів глибокого навчання, з'явилась можливість у полегшенні ознайомлення дослідників метаматеріалів з цими методами, що в останні роки і показало велику кількість ідей [78-87]. Складність виникає тільки в правильно поставлених задачах та підготовці даних для систем глибокого навчання.

РОЗДІЛ 2.

ВИКОРИСТАННЯ ЗГОРТКОВОЇ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ АНАЛІЗУ ДАНИХ

Ключовим фактором проектування нейронних мереж є визначення архітектури. Архітектура належить до загальної структури мережі: скільки нейронів вона повинна мати і як вони повинні бути з'єднані.

2.1 Підходи до побудови архітектури нейронних мереж

В більшості архітектур нейронних мереж ці рівні впорядковані в ланцюгову структуру, причому кожен шар є функцією шару, який йому передував. Для приклада, в такій структурі перший шар представлений наступним чином:

$$h^{(1)} = g^{(1)} \odot (W^{(1)T} x + b^{(1)}), \quad (2.1)$$

де x – вхідна інформація, $W^{(i)T}$ – транспонована матриця вагів i -того шару, h^i – вектор виходів i -того шару, g^i – функція активації i -того шару, b^i – вектор зсувів i -того шару, $\odot$ - операція Адамара, дія функції до вектора по компонентно.

Другий шар описується:

$$h^{(2)} = g^{(2)} \odot (W^{(2)T} h^{(1)} + b^{(2)}), \quad (2.2)$$

І так далі. У цих ланцюжкових архітектурах основними архітектурними міркуваннями є вибір глибини мережі та ширини кожного шару. Мережі з навіть одним прихованим шаром достатньо, щоб вони підійшли для навчального набору [115]. Мережі з більшою кількістю внутрішніх шарів часто можуть використовувати набагато менше нейронів на шар і набагато менше параметрів і часто узагальнюють тестовий набір, але їх також часто важче оптимізувати. Ідеальну мережеву

архітектуру для завдання потрібно знайти за допомогою експериментів, керованих моніторингом помилки.

Найпростіша лінійна модель, що відображує ознаки на виходи за допомогою множення матриці, за визначенням може представляти лише лінійні функції. Її перевага полягає в тому, що її легко навчити. На жаль, часто потрібно застосовувати нелінійні функції.

Вивчення нелінійної функції потребує проектування спеціалізованого сімейства моделей для типу нелінійності, які потрібно вивчити.

Для врахування складності нелінійних функцій, які потрібно розпізнавати системою глибокого навчання, використовується універсальна апроксимаційна теорема - [125], [126]. Вона стверджує, що мережа прямого зв'язку з лінійним вихідним рівнем і принаймні одним прихованим шаром з будь-якою функцією активації (такою, як функція логістичної сигмоподібної форми) може апроксимувати будь-яку багатовимірну (більше одного прихованого шару), за Боррелем, функцію з одного скінченновимірного простору в інший з будь-якою бажаною ненульовою величиною похибки, за умови, що мережа має достатню кількість нейронів. Похідні прямої мережі також можуть як завгодно добре апроксимувати похідні функції – [127]. Підсумовуючи: будь-яка неперервна функція на замкнутій і обмеженій підмножині R^n є вимірною за Боррелем і тому може бути апроксимована нейронною мережею. Нейронна мережа також може апроксимувати будь-яку функцію, яка відображається з будь-якого скінченновимірного дискретного простору в інший. Вихідні теореми були сформульовані в термінах нейронів із функціями активації, які насичуються як для дуже негативних, так і для дуже позитивних аргументів. Своєю чергою універсальні теореми про апроксимацію були доведені для ширшого класу функцій активації, який включає зараз широко використовувану випрямлену лінійну функцію – [128].

Універсальна теорема про наближення означає, що незалежно від того, яку функцію потрібно вивчити, є інформація, що великий *MLP* (багатошаровий

перцептрон) зможе представити цю функцію. Однак не гарантується, що навчальний алгоритм зможе натренуватись на цю функцію. Навіть якщо *MLP* здатний представити функцію, навчання може бути невдалим з двох різних причин. По-перше, алгоритм оптимізації, який використовується для навчання, може бути не в змозі знайти значення параметрів, яке відповідає бажаній функції. По-друге, алгоритм навчання може вибрати неправильну функцію через перенавчання. Мережі прямого зв'язку забезпечують універсальну систему для представлення функцій у тому сенсі, що за заданою функцією існує мережа прямого зв'язку, яка наближає функцію. Немає універсальної процедури для вивчення навчальної множини конкретних прикладів і вибору функції, яка буде узагальнюватися на точки, що не входять до навчальної множини.

Універсальна теорема про наближення говорить, що існує мережа, достатньо велика для досягнення будь-якого ступеня точності, яку потрібно отримати, але теорема не визначає, наскільки великою має бути мережа. В роботі [129] показано деякі межі розміру одношарової мережі, необхідної для апроксимації широкого класу функцій.

На жаль, у гіршому випадку може знадобитися експоненціальна кількість прихованих нейронів (можливо, з одним прихованим нейроном, що відповідає кожній вхідній конфігурації, яку потрібно розрізняти). Це найлегше побачити у двійковому випадку: кількість можливих двійкових функцій на векторах $v \in \{0, 1\}^n$ дорівнює 2^{2^n} , і для вибору однієї такої функції потрібно 2^n бітів, що загалом вимагатиме $O(2^n)$ ступенів свободи.

Враховуючи приведену інформацію: мережі прямого зв'язку з одним шаром достатньо для представлення будь-якої функції, але якщо шар не достатньо великий, то є вірогідність отримати неправильне навчання та узагальнення. У багатьох випадках використання глибших моделей може зменшити кількість нейронів, необхідних для представлення бажаної функції, і може зменшити кількість помилок узагальнення.

Існують сімейства функцій, які можуть бути ефективно апроксимовані архітектурою з глибиною, більшою за деяке значення d , але які вимагають значно більшої моделі, якщо глибина є менша або рівна d . У багатьох випадках кількість прихованих нейронів, необхідних для неглибокої моделі, експоненціальна по числу n (кількість нейронів). Перші результати були отримані для схем логічних елементів – [130]. Пізніші роботи розповсюдили ці результати на лінійні порогові нейрони з додатними вагами [131], [132], а потім на мережі з безперервними функціями активації - [133], [134]. Багато сучасних нейронних мереж використовують випрямлені лінійні нейрони. В роботі [135] продемонстровано, що неглибокі мережі з широким набором не поліноміальних функцій активації, включаючи випрямлені лінійні нейрони мають властивості універсальної апроксимації, але ці результати не стосуються питань глибини чи ефективності — вони вказують лише на те, що достатньо широка мережа випрямних нейронів можуть представляти будь-яку функцію.

В роботі [135] було показано, що функції, які можна представити за допомогою глибокої мережі з випрямною функцією, можуть вимагати експоненціальної кількості прихованих нейронів з неглибокою (один прихований шар) мережею.

Точніше, вони показали, що кусково-лінійні мережі (які можна отримати з випрямних нелінійностей) можуть представляти функції з кількістю областей, яка є експоненціальною в глибині мережі. На рисунку 2.1 показано (геометрична інтерпретація), як мережа з абсолютним випрямленням створює дзеркальні відображення функції, обчисленої поверх деякого прихованого нейрона, щодо входу цього прихованого нейрона. Кожен прихований нейрон вказує, куди потрібно згорнути вхідний простір, щоб створити дзеркальні відповіді (по обидва боки абсолютної нелінійності). Компонуючи ці операції згортання, маємо експоненціально велику кількість кусково-лінійних областей, які можуть захоплювати всі типи регулярних (наприклад, повторюваних) візерунків.

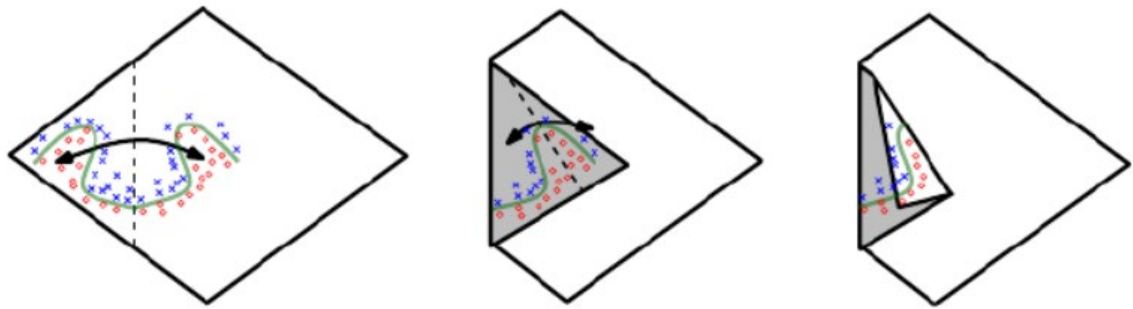


Рисунок 2.1 – Геометричне пояснення експоненціальної переваги випрямних глибоких мереж випрямляча [135].

Зліва - нейрон з абсолютним випрямленням має однаковий вихід для кожної пари дзеркальних точок на своєму вході. Дзеркальна вісь симетрії задана гіперплощиною, визначеною вагами та зміщенням нейрона. Функція, обчислена поверх цього нейрону буде дзеркальним відображенням простішого шаблону через цю вісь симетрії. По центру - функцію можна отримати шляхом згортання простору навколо осі симетрії. Праворуч - інший повторюваний візерунок можна скласти поверх першого (іншим наступним нейроном), щоб отримати іншу симетрію (яка тепер повторюється чотири рази з двома прихованими шарами).

Основна теорема в [135] стверджує, що кількість лінійних областей, вирізаних мережею глибокого випрямляча з d входами, глибиною l і n нейронами на прихований шар, становить:

$$O \left(\binom{n}{d}^{d(l-1)} n^d \right), \quad (2.3)$$

тобто експоненціально по глибині l . У випадку мереж maxout з k фільтрами на нейрон кількість лінійних областей дорівнює:

$$O(k^{(l-1)+d}), \quad (2.4)$$

Звичайно, немає жодної гарантії, що види функцій, які потрібно вивчити в програмах глибокого навчання (і зокрема для ШП), мають таку властивість.

Також, є можливість вибрати глибоку модель зі статистичних причин. Кожного разу, коли обирається конкретний алгоритм глибокого навчання, неявно заявляється про певний набір попередніх переконань, які маємо щодо того, яку саме функцію має вивчати алгоритм. Вибір глибокої моделі кодує дуже загальне переконання, що функція, яку потрібно вивчити, повинна включати композицію кількох простіших функцій. Це можна інтерпретувати з точки зору навчання як те, що вважається, що проблема навчання полягає у виявленні набору базових факторів варіації, які, своєю чергою, можуть бути описані в термінах інших, простіших основних факторів варіації. Крім того, є можливість інтерпретувати використання глибокої архітектури, як вираження переконання, що функція, яку потрібно вивчити, є комп'ютерною програмою, що складається з кількох кроків, де кожен крок використовує вихідні дані попереднього кроку. Ці проміжні виходи не обов'язково є факторами варіації, натомість вони можуть бути аналогічними лічильникам або показникам, які мережа використовує для організації внутрішньої обробки. Емпірично більша глибина дійсно призводить до кращого узагальнення для широкого спектра завдань – рис. 2.2 та рис. 2.3 [136-147]. Це свідчить про те, що використання глибоких архітектур справді є дійсно корисним методом для вивчення простору функцій.

Цей експеримент [146] показує, що збільшення кількості параметрів у шарах згорткових мереж без збільшення їх глибини не є настільки ефективним для підвищення продуктивності тестового набору.

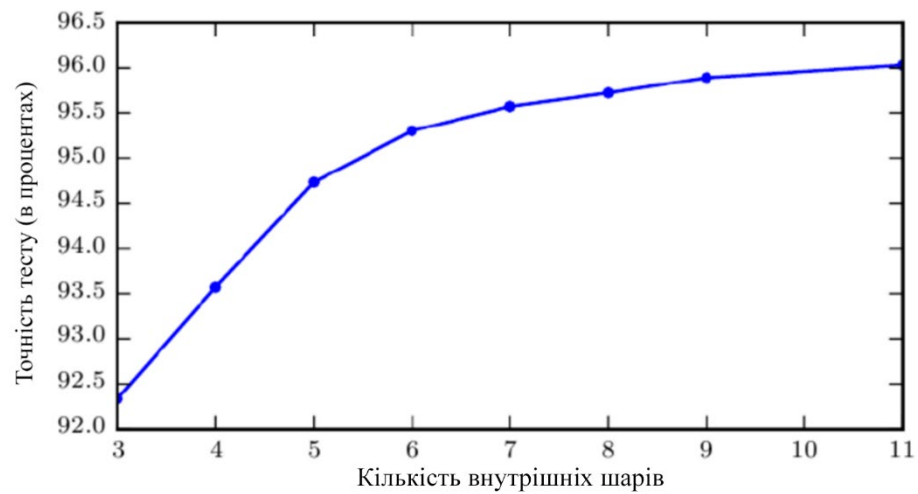


Рисунок 2.2 – Залежність точності прогнозування тесту від кількості внутрішніх шарів в нейронній мережі.

Емпіричні результати показують, що глибші мережі краще узагальнюють, коли використовуються для транскрипції багатозначних чисел із фотографій – [146]. Точність постійно зростає зі збільшенням глибини.

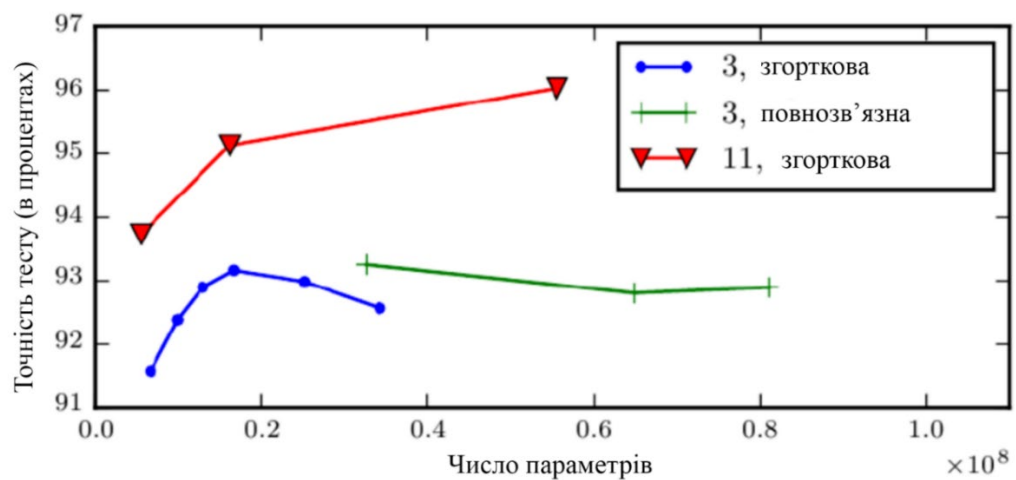


Рисунок 2.3 – Залежність точності прогнозування тесту від числа параметрів у внутрішніх шарах.

Моделі з більшою кількістю внутрішніх шарів, як правило, працюють краще. Це не тільки тому, що модель більша.

Відведена таблиця на рис. 2.3 вказує на глибину мережі, яка використовується для створення кожної кривої, і на те, чи крива представляє зміну згорткових, чи повністю з'єднаних шарів. Спостерігається, що дрібні моделі в цьому контексті перенавчаються приблизно на 20 мільйонах параметрів, тоді як в глибоких мережах є виграш навіть при кількості параметрів більше ніж 60 мільйонів. Це означає, що використання глибокої моделі виражає корисну перевагу над простором функцій, які модель може вивчати. Зокрема, це виражає переконання, що функція повинна складатися з багатьох простіших функцій, складених разом. Це може призвести або до вивчення представлення, яке своєю чергою складається з простіших представлень (наприклад, кутів, визначених у термінах ребер), або до вивчення програми з послідовними залежними кроками (наприклад, спочатку знайти набір об'єктів, а потім сегментувати один від одного, і тоді розпізнавати).

Досі було описано нейронні мережі як прості ланцюжки шарів, головними міркуваннями яких є глибина мережі та ширина кожного шару. На практиці нейронні мережі демонструють значно більшу різноманітність.

Багато архітектур нейронної мережі було розроблено для конкретних завдань. Спеціалізовані архітектури для комп'ютерного зору, які називаються згортковими мережами. Мережі прямого зв'язку також можуть бути узагальнені до рекурентних нейронних мереж для послідовної обробки, які мають свої власні архітектурні міркування.

Загалом, шари не обов'язково з'єднувати в ланцюг, навіть якщо це найпоширеніша практика. Багато архітектур створюють основний ланцюг, але потім додають до нього додаткові архітектурні функції, наприклад, пропускають з'єднання, що переходять від рівня i до рівня $i + 2$ або вище. Ці з'єднання пропуску полегшують перетікання градієнта від вихідних шарів до шарів, ближчих до вхідних.

Іншим ключовим фактором проектування архітектури є те, як саме з'єднати пару шарів один з одним. В шарі нейронної мережі, як правило, описаному лінійним перетворенням через матрицю W , кожен вхідний нейрон з'єднаний з кожним вихідним нейроном. Багато мереж, які створені для розв'язання спеціальних задач, мають менше з'єднань, тому кожен нейрон на вхідному рівні під'єднаний лише до невеликої підмножини нейронів на вихідному рівні. Ці стратегії зменшення кількості з'єднань знижують кількість параметрів і обсяг обчислень, необхідних для оцінки мережі, але часто сильно залежать від задачі. Наприклад, згорткові мережі використовують спеціальні шаблони розріджених з'єднань, які дуже ефективні для проблем комп'ютерного зору.

2.2 Використання операції згортки в нейронних мережах для можливості прогнозування властивостей 2D та 3D об'єктів

Згорткові мережі, також відомі як згорткові нейронні мережі або *CNN*, є спеціалізованим типом нейронних мереж для обробки даних, які мають відому сіткову топологію. Приклади включають дані часових рядів, які можна розглядати як одновимірну сітку, яка бере набори даних через періодичні проміжки часу, і дані зображення, які можна розглядати як двовимірну сітку пікселів. Згорткові мережі користуються величезним успіхом у практичних застосуваннях. Назва «згорткова нейромережа» вказує на те, що мережа використовує математичну операцію — згортку. Згорткові мережі — це просто нейронні мережі, які використовують згортку замість загального множення матриць принаймні в одному зі своїх шарів.

У програмах глибокого навчання вхідними даними зазвичай є багатовимірні масиви даних, а ядром зазвичай є багатовимірний масив параметрів, і всі вони адаптуються алгоритмом навчання. Ці багатовимірні масиви називаються тензорами. Оскільки кожен елемент вхідних даних і ядра має явно зберігатися окремо, зазвичай припускається, що ці функції дорівнюють нулю скрізь, крім

скінченного набору точок, для яких зберігається значення. Це означає, що на практиці є можливість реалізувати нескінченне підсумовування, як підсумовування по скінченній кількості елементів масиву.

Згортки часто використовуються над більш ніж однією віссю одночасно. Наприклад, якщо використовується двовимірне зображення I як вхідні дані, ймовірно, також потрібно використовувати двовимірне ядро K :

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n) K(i - m, j - n) \quad (2.5)$$

У випадку двовимірної згортки вихідний сигнал обмежується лише позиціями, де ядро повністю лежить у зображенні, що в деяких контекстах називається «дійсною» згорткою. На рис. 2.4 схематично показана операція згортки. Прямокутники зі стрілками показані, щоб вказати, як формується верхній лівий елемент вихідного тензора шляхом застосування ядра до відповідної верхньої лівої області вхідного тензора.

Згортка використовує три важливі ідеї, які можуть допомогти покращити систему глибокого навчання: розріджені взаємодії, спільне використання параметрів та еквівалентні представлення – далі буде пояснення. Крім того, згортка надає засоби для роботи з вхідними даними змінного розміру.

Традиційні шари нейронної мережі використовують множення матриці на матрицю параметрів з окремим параметром, що описує взаємодію між кожною вхідною одиницею та кожною вихідною одиницею. Це означає, що кожен вихідний блок взаємодіє з кожним вхідним блоком. Проте згорткові мережі зазвичай мають розріджену взаємодію (також відому як розріджена зв'язність або розріджені ваги – розріджені взаємодії). Це досягається шляхом зменшення розміру ядра порівняно зі входом. Наприклад, під час обробки зображення вхідне зображення може містити тисячі чи мільйони пікселів, але можливо виявити невеликі значущі елементи, такі як грані з ядрами, які займають лише десятки чи сотні пікселів. Це означає, що потрібно оброблювати менше параметрів, що зменшує вимоги до пам'яті моделі та покращує її статистичну ефективність. Це також означає, що обчислення результату

вимагає менше операцій. Ці покращення ефективності зазвичай досить значні. Якщо є m входів і n виходів, то для множення матриці потрібні $m \times n$ параметрів, а алгоритми, які використовуються на практиці, мають $O(m \times n)$ час виконання. Якщо обмежитися кількістю підключень, які може мати кожен вихід, до k , тоді розріджено підключений підхід вимагає лише $k \times n$ параметрів і $O(k \times n)$ часу виконання.

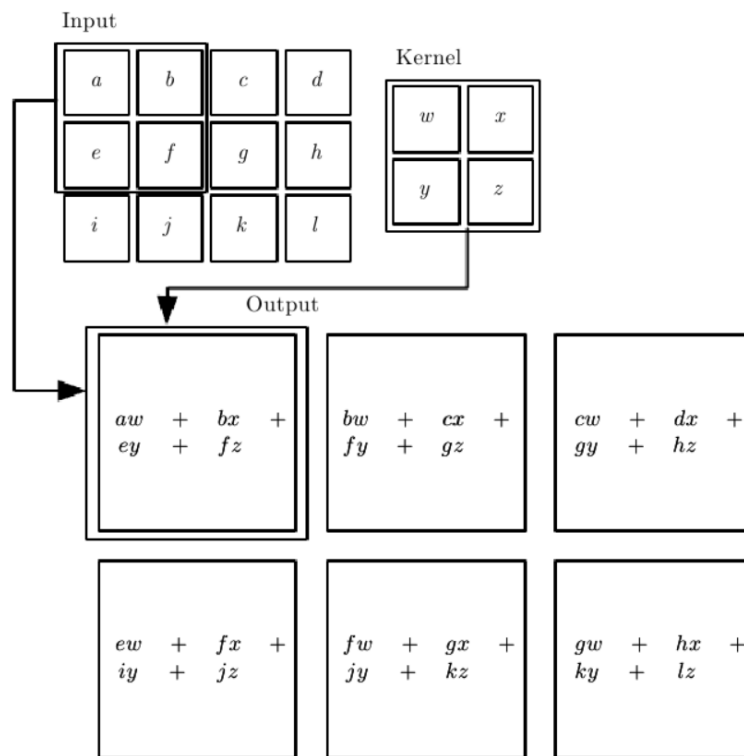


Рисунок 2.4 – Схема роботи операції 2D згортки [101].

Для багатьох практичних застосувань є можливість отримати хорошу продуктивність у задачах глибокого навчання, зберігаючи k на кілька порядків меншим за m . У глибокій згортковій мережі, нейрони у внутрішніх шарах можуть опосередковано взаємодіяти з більшою частиною вхідних даних.

Це дозволяє мережі ефективно описувати складні взаємодії між багатьма змінними, створюючи такі взаємодії з простих алгоритмічних блоків, кожен з яких описує лише розріджені взаємодії. Такі процеси схематично зображені на рис. 2.5, рис. 2.6, рис. 2.7 [100].

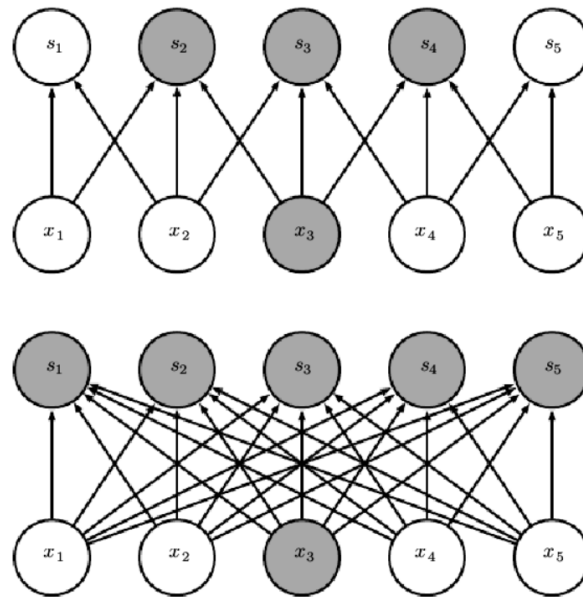


Рисунок 2.5 – Розріджений зв’язок: виділяється один вхідний блок x_3 , а також виділяються вихідні одиниці в s , на які впливає цей блок.

Коли s формується згорткою з ядром шириною 3, x впливає лише на три результати. Коли s формується множенням матриці, зв’язність більше не є розрідженою, тому на всі виходи впливає x_3 .

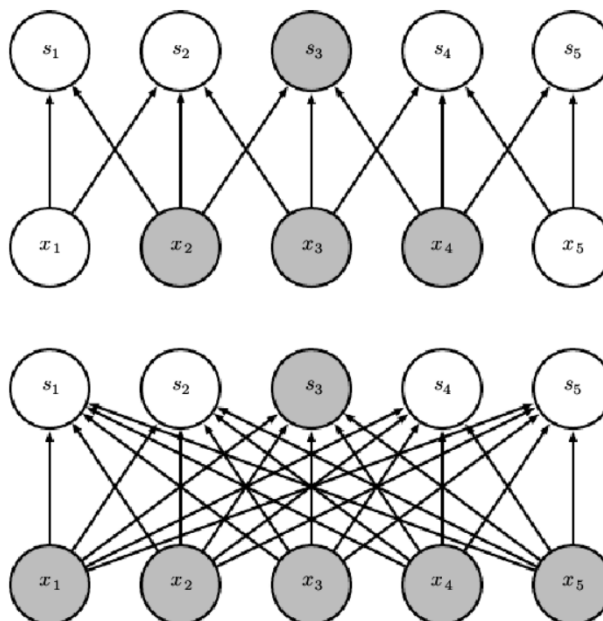


Рисунок 2.6 – Розріджений зв’язок: виділяється один вихідний блок s_3 , а також виділяються вхідні блоки в x , які впливають на цей блок.

Ці нейрони відомі як поле сприйняття s_3 . Коли s формується згорткою з ядром ширини 3, лише три входи впливають на s_3 . Коли s формується множенням матриці, зв'язність більше не є розрідженою, тому всі вхідні дані впливають на s_3 [100].

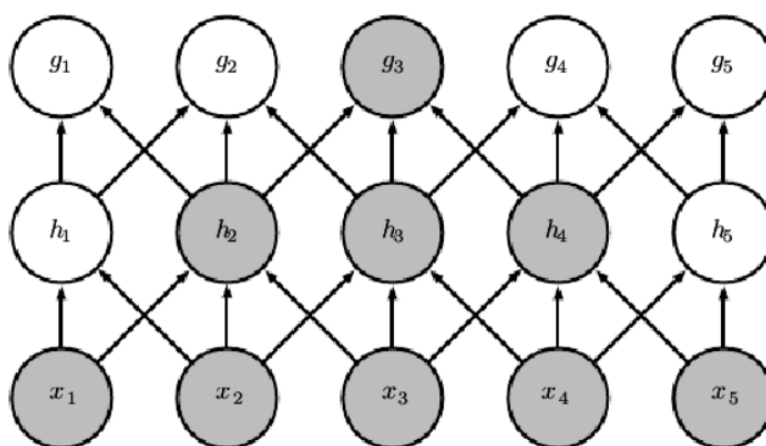


Рисунок 2.7 - Поле сприйняття нейронів у глибших шарах згорткової мережі.

Поле сприйняття нейронів у глибших шарах більше, ніж поле сприйняття нейронів у поверхневих шарах. Цей ефект посилюється, якщо мережа включає такі архітектурні особливості, як поступова згортка або об'єднання. Це означає, що навіть попри те, що прямі зв'язки в згортковій мережі дуже рідкі, нейрони у внутрішніх шарах можуть бути опосередковано пов'язані з усім зображенням або більшою частиною вхідного зображення.

Спільне використання параметрів означає використання одного параметра для кількох функцій у моделі. У традиційній нейронній мережі кожен елемент вагової матриці використовується рівно один раз під час обчислення результатів по шару. Він множиться на один елемент введення, а потім ніколи не переглядається. Як синонім спільного використання параметрів можна сказати, що мережа має зв'язані ваги, оскільки значення ваги, застосоване до одного входу, прив'язане до значення ваги, застосованого в іншому місці. У згортковій нейронній мережі кожен член ядра використовується в кожній позиції вхідних даних (за винятком, можливо, деяких граничних пікселів, залежно від проєктних рішень щодо обмежень для даних).

Спільне використання параметрів, яке використовується операцією згортки, означає, що замість вивчення окремого набору параметрів для кожного місця вивчається лише один набір. Це не впливає на час виконання прямого проходження — він все ще дорівнює $O(k \times n)$, але додатково зменшує вимоги до пам'яті моделі з k параметрами. k зазвичай на кілька порядків менше m . Оскільки m і n , як правило, мають приблизно однаковий розмір, k є незначним у порівнянні з $m \times n$. Таким чином, згортка є значно ефективнішою, ніж множення цілої матриці з точки зору вимог до пам'яті та статистичної ефективності.

Як приклад перших двох принципів у дії, рис. 2.8, видно, як розріджене підключення та спільний доступ до параметрів можуть значно підвищити ефективність лінійної функції для виявлення країв у зображенні.

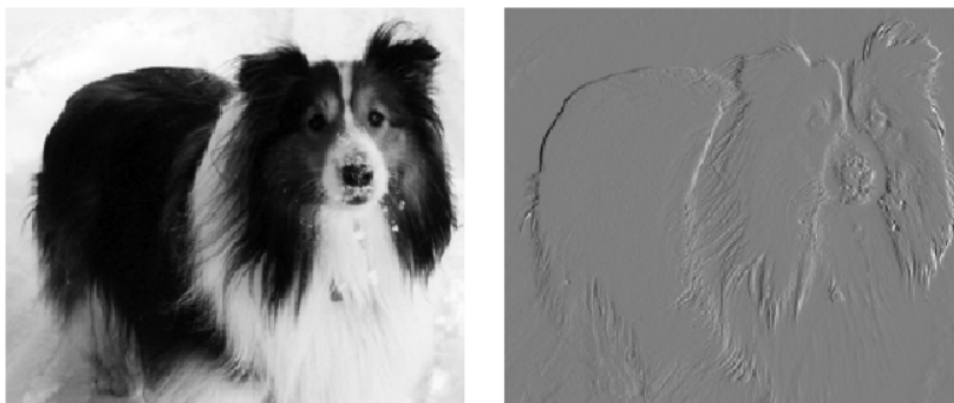


Рисунок 2.8 – Зображення праворуч було сформовано шляхом взяття кожного пікселя вихідного зображення та віднімання значення його сусіднього пікселя зліва. Це показує можливість всіх вертикально орієнтованих країв у вхідному зображенні, що може бути корисною операцією для виявлення об'єктів [101].

Згортка не є природним еквівалентом до деяких інших перетворень, таких як масштабування або обертання зображення.

Деякі типи даних не можуть бути оброблені нейронними мережами, визначеними множенням матриці на матрицю фіксованої форми. Згортка дозволяє обробляти деякі з цих типів даних.

2.3 Застосування функції об'єднання для визначення властивостей вибірки

Типовий шар згорткової мережі складається з трьох етапів. Компоненти типового шару згорткової нейронної мережі представлені на рис 2.9.

На першому етапі шар виконує кілька згорток паралельно, щоб отримати набір лінійних активацій. На другому етапі кожна лінійна активація проходить через нелінійну функцію активації, таку як випрямлена лінійна функція активації. Цю стадію іноді називають детекторною. На третьому етапі використовується функція об'єднання для подальшої зміни результатів шару. Функція об'єднання замінює вихід мережі в певному місці сумарною статистикою найближчих виходів.

Наприклад, операція максимального об'єднання показує максимальний результат у прямокутному околі. Інші популярні функції об'єднання включають середнє значення прямокутної околиці, норму L^2 прямокутної околиці або середнє зважене на основі відстані від центрального пікселя.

Для опису цих рівнів існує два загальновживаних набори термінології. За термінологією зліва, згорткова нейромережа розглядається як невелика кількість відносно складних шарів, кожен з яких має багато етапів. Відповідно до цієї термінології існує однозначне відображення між тензорами ядра та мережевими рівнями. У термінології справа, згорткова нейромережа розглядається як більша кількість простих шарів; кожен крок обробки розглядається як окремий шар. Це означає, що не кожен шар має параметри.

У всіх випадках, об'єднання допомагає зробити представлення приблизно інваріантним до невеликих трансляцій вхідних даних. Інваріантність до трансляції

означає, що якщо вхідні дані переводяться на невелику величину, значення більшості об'єднаних виходів не змінюються.

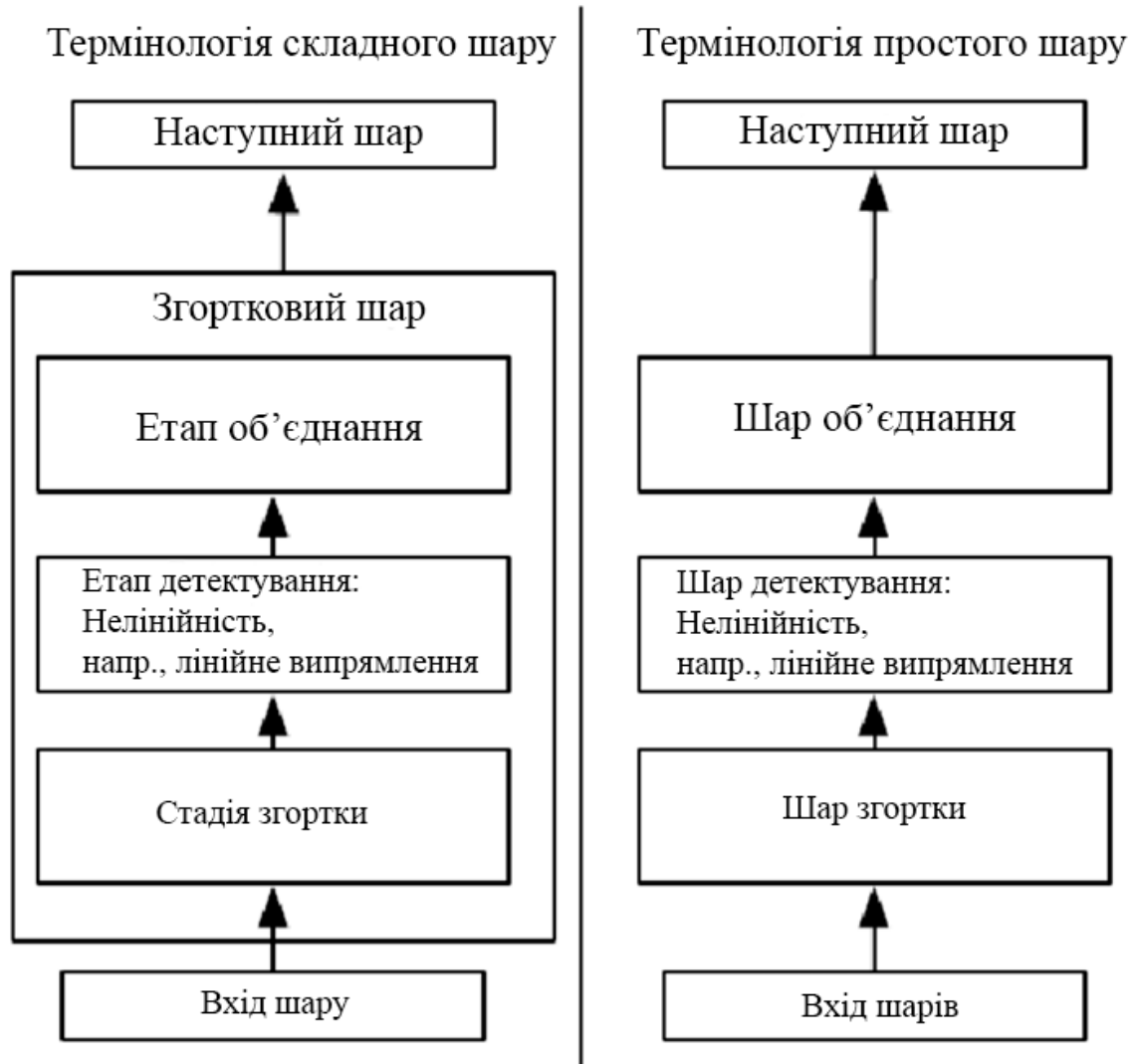


Рисунок 2.9 – Компоненти типового шару згорткової нейронної мережі.

У всіх випадках, об'єднання допомагає зробити представлення приблизно інваріантним до невеликих трансляцій вхідних даних. Інваріантність до трансляції означає, що якщо вхідні дані переводяться на невелику величину, значення більшості об'єднаних виходів не змінюються.

Інваріантність до локальної трансляції може бути дуже корисною властивістю, якщо більше дбається про те, чи присутня якась ознака, ніж про те, де вона є – рис. 2.10. Наприклад, визначаючи, чи містить зображення обличчя, не потрібно знати

розташування очей з ідеальною точністю до пікселів, потрібно просто знати, чи є око з лівого боку обличчя та око з правого боку обличчя. В інших контекстах важливіше зберегти розташування об'єкта. Наприклад, якщо потрібно знайти кут, визначений двома ребрами, що зустрічаються в певній орієнтації, потрібно зберегти розташування ребер достатньо добре, щоб перевірити, чи вони стикаються.

Зверху - вигляд результату згорткового шару. Нижній ряд показує виходи нелінійності. Верхній ряд показує результати максимального об'єднання з кроком в один піксель між областями об'єднання та шириною об'єднання в три пікселі. Знизу - вид тієї самої мережі після того, як введення було зміщено праворуч на один піксель. Кожне значення в нижньому ряді змінилося, але лише половина значень змінилася у верхньому ряді, оскільки максимальні одиниці об'єднання чутливі лише до максимального значення в околицях, а не до його точного розташування.

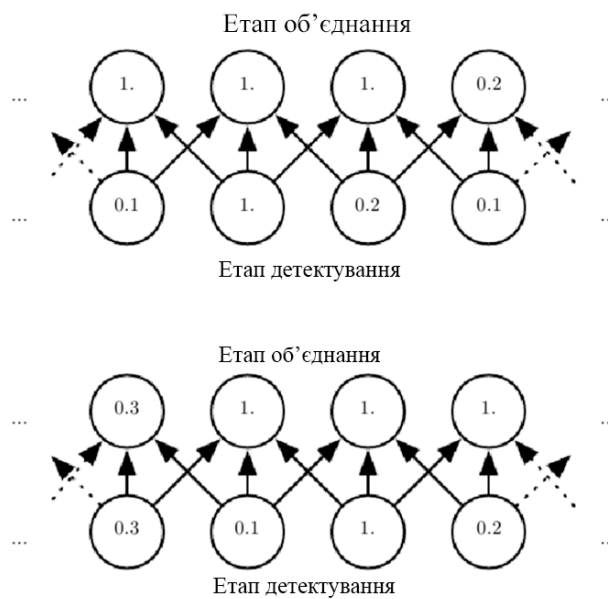


Рисунок 2.10 – Максимальне об'єднання.

Об'єднання просторових областей створює інваріантність до трансляції, але якщо потрібно об'єднати результати окремо параметризованих згорток, об'єкти можуть адаптуватись до яких трансляцій стати інваріантними.

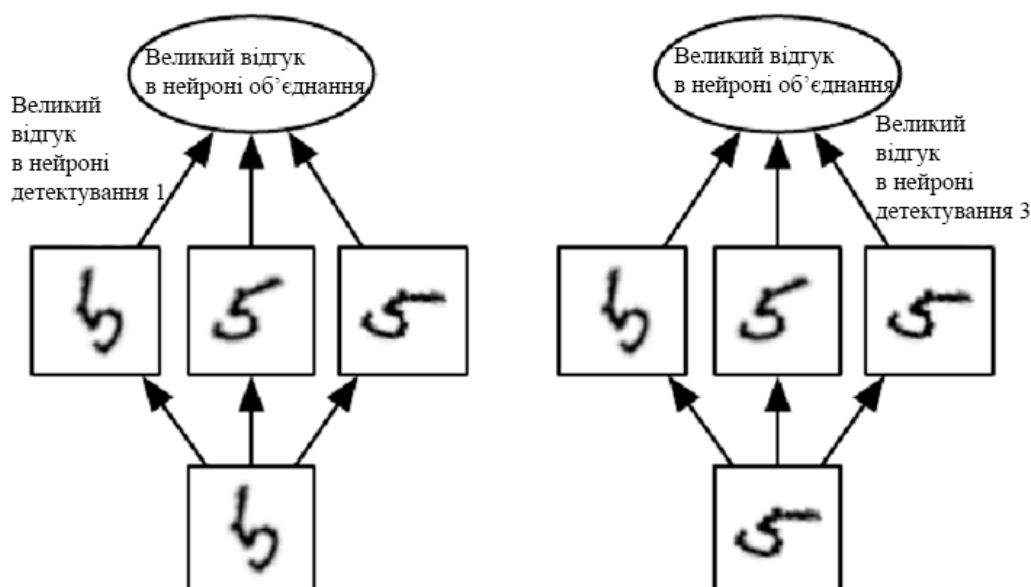


Рисунок 2.11 – Приклад навчених інваріантностей: блок об'єднання, який об'єднує кілька ознак, які вивчаються з окремими параметрами, може навчитися бути інваріантним до перетворень вхідних даних.

На рис. 2.11, вказано, як набір із трьох навчених фільтрів і максимального блоку об'єднання може навчитися ставати інваріантним до обертання. Усі три фільтри призначені для виявлення написаної від руки цифри п'ять. Кожен фільтр намагається зіставити певну орієнтацію числа п'яти. Коли у вхідних даних з'являється цифра п'ять, відповідний фільтр відповідатиме їй, що спричинить велику активацію блоку детектора. Тоді блок максимального об'єднання матиме велику активацію незалежно від того, який блок детектора було активовано. На рис. 2.11 показано, як мережа обробляє два різні входи, що призводить до активації двох різних блоків детектора. Вплив на блок об'єднання приблизно однаковий у будь-якому випадку. Цей принцип використовується мережами *maxout* та іншими згортковими мережами.

Оскільки об'єднання підсумовує відповіді по всій околиці, можна використовувати менше нейронів об'єднання, ніж нейронів детектора, повідомляючи підсумкову статистику для областей об'єднання, розташованих на відстані k пікселів, а не на 1 піксель один від одного, як показано на рис. 2.10. Така

процедура покращує обчислювальну ефективність мережі, оскільки наступний шар має приблизно в k разів менше вхідних даних для обробки. Коли кількість параметрів у наступному шарі є функцією його вхідного розміру (наприклад, коли наступний рівень повністю зв'язаний і базується на множенні матриці), таке зменшення розміру вхідних даних може призвести до покращення статистичної ефективності та зменшення вимог до пам'яті для збереження параметрів.

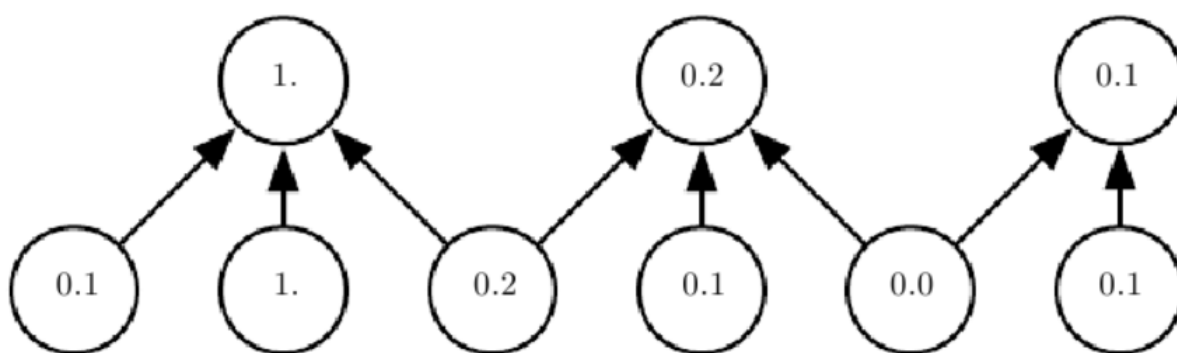


Рисунок 2.12 – Об'єднання зі зменшенням дискретизації.

Тут використовується максимальне об'єднання з шириною вибірки в три нейрони та кроком між вибірками у два нейрони. Це зменшує розмір представлення у два рази, що зменшує обчислювальне та статистичне навантаження на наступний рівень. Варто зазначити, що крайня права область об'єднання має менший розмір, але її необхідно включити в об'єднання, якщо не ігнорувати деякі блоки детектора.

Для багатьох завдань, об'єднання є важливим для обробки вхідних даних різного розміру. Наприклад, якщо потрібно класифікувати зображення змінного розміру, вхід до шару класифікації повинен мати фіксований розмір. Зазвичай це досягається зміною розміру зміщення між областями об'єднання, щоб класифікаційний рівень завжди отримував однакову кількість зведених статистичних даних незалежно від розміру вхідних даних. Наприклад, кінцевий рівень об'єднання в мережі може бути визначений для виведення чотирьох наборів

підсумкової статистики, по одному для кожного квадранта зображення, незалежно від розміру зображення.

Деякі теоретичні роботи дають вказівки щодо того, які види об'єднання слід використовувати в різних ситуаціях [101]. Також можна динамічно об'єднувати об'єкти разом, наприклад, запустивши алгоритм кластеризації на місцях об'єктів інтересу [98]. Цей підхід дає різний набір областей об'єднання для кожного зображення. Інший підхід полягає в тому, щоб вивчити єдину структуру об'єднання, яка потім застосовується до всіх зображень.

Об'єднання може ускладнити деякі типи архітектур нейронних мереж, які використовують інформацію зверху вниз, наприклад, машини Больцмана та автокодери.

Деякі приклади повної згорткової мережевої архітектури для класифікації з використанням згортки та об'єднання показані на рис. 2.13.

Конкретні кроки та глибини, використані на цьому рисунку, не рекомендовані для реального використання; вони розроблені так, щоб бути дуже дрібними, щоб поміститися на сторінку. Справжні згорткові мережі також часто включають значну кількість розгалужень, на відміну від ланцюгових структур, які використовуються тут для простоти. Зліва представлена згортка, яка обробляє фіксований розмір зображення. Після чергування між згорткою та об'єднанням для кількох шарів, тензор для згорткової карти об'єктів змінюється, щоб компенсувати інформацію, яка відкидається. Решта мережі є звичайною мережею прямого зв'язку. По центру рис. 2.13 - ілюстрація згорткової нейромережі, яка обробляє зображення змінного розміру, але все ще підтримує повністю зв'язаний шар. Ця мережа використовує операцію об'єднання з вибірками змінного розміру, але з фіксованою кількістю вибірок, щоб забезпечити вектор фіксованого розміру з 576 нейронами для повністю підключеної частини мережі. Права частина рис. 2.13 зображує згорткову нейромережу, яка не має повно зв'язаного шару нейронів. Замість цього останній згортковий шар виводить одну карту ознак на клас. Імовірно, модель вивчає карту

того, наскільки ймовірно кожен клас з'явиться в кожному просторовому місці. Усереднення карти ознак до одного значення забезпечує аргумент для активатора softmax у верхній частині.

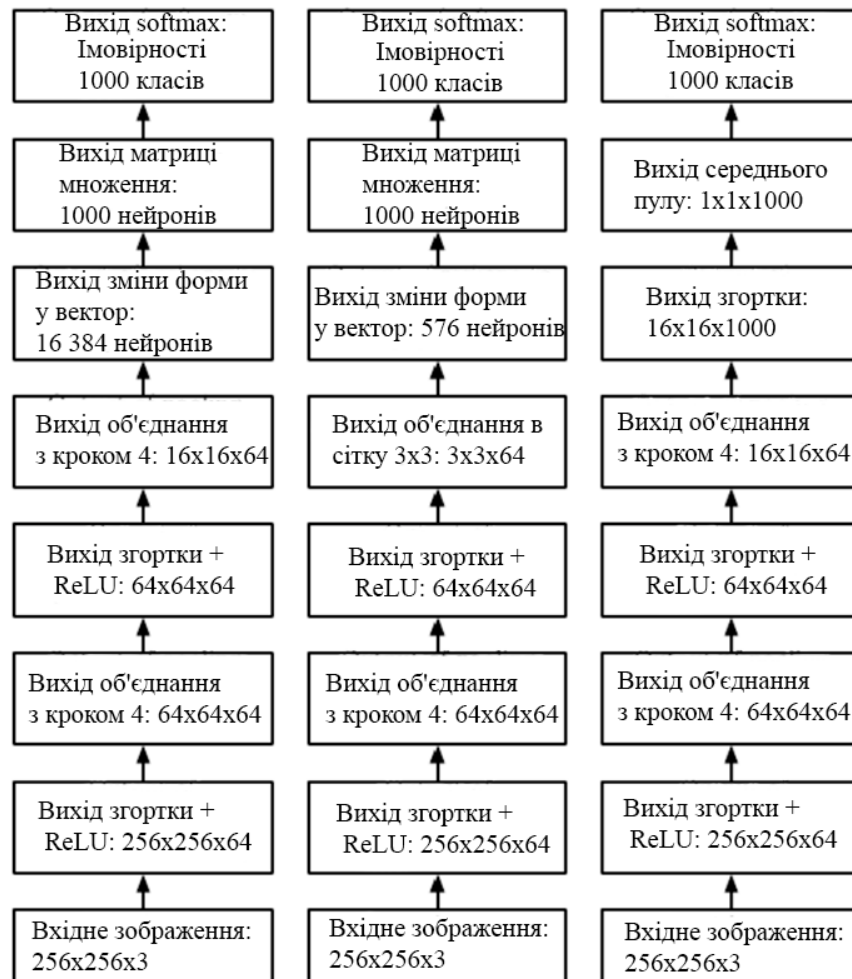


Рисунок 2.13 – Приклади архітектур для класифікації зі згортковими мережами.

2.4. Особливості використання згорткової нейронної мережі

Обговорюючи згортку в контексті нейронних мереж, зазвичай не мається на увазі стандартна операція дискретної згортки, як це зазвичай розуміють у

математичній літературі. Функції, що використовуються на практиці в даному контексті, дещо відрізняються.

По-перше, коли говориться про згортку в контексті нейронних мереж, зазвичай фактично мається на увазі операція, яка складається з багатьох застосувань згортки паралельно. Це пояснюється тим, що згортка з одним ядром може витягти лише один вид функції, хоча й у багатьох просторових розташуваннях. Зазвичай потрібно, щоб кожен рівень нашої мережі витягував багато видів функцій у багатьох місцях.

Крім того, вхідні дані зазвичай не є просто масивом реальних значень. Найчастіше, це масиви векторних спостережень. Наприклад, кольорове зображення має інтенсивність червоного, зеленого та синього кольору в кожному пікселі. У багат шаровій згортковій мережі вхід до другого рівня є виходом першого рівня, який зазвичай має вихід багатьох різних згорток у кожній позиції. Працюючи з зображеннями, зазвичай вважається про вхід і вихід згортки як про тривимірні тензори з одним індексом для різних каналів і двома індексами для просторових координат кожного каналу. Реалізації програмного забезпечення зазвичай працюють у пакетному режимі, тому вони фактично використовують 4D тензори.

2.4.1 Варіанти основної функції згортки

Оскільки в згорткових мережах зазвичай використовується багатоканальна згортка, лінійні операції, на яких вони засновані, не гарантовано будуть комутативними, навіть якщо використовується інверсія ядра. Такі багатоканальні операції є комутативними тільки тоді коли кожна операція має таку ж кількість вихідних каналів, як і вхідних.

Припустимо, що є 4-вимірний ядерний тензор K з елементом $K_{i,j,k,l}$ що забезпечує зв'язок між нейроном в каналі i вихідного сигналу та нейроном в каналі j вхідного, зі зміщенням на k рядків і l стовпців між вихідним блоком та вхідним.

Припустимо, що вхідні дані складаються зі спостережуваних даних V з елементом $V_{i,j,k}$, що дає значення вхідного нейрона в каналі i в рядку j і стовпці k . Припустимо, що результат представляється масивом Z з такими ж індексами, що й V . Якщо Z утворюється шляхом згортання K через V без інверсії K , тоді буде представлення:

$$Z_{i,j,k} = \sum_{l,m,n} V_{l,j+m-1,k+n-1} K_{i,l,m,n}, \quad (2.6)$$

де підсумовування за l , m і n стосується всіх значень, для яких дійсні операції індексування тензора всередині підсумовування. У нотації лінійної алгебри індексуються масиви, використовуючи 1 для першого запису. Це вимагає значення -1 у наведеній вище формулі.

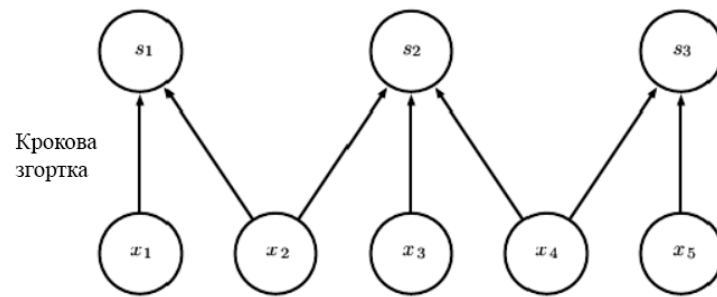
Мови програмування, такі як C і $Python$, індексують, починаючи з 0, що робить наведений вище вираз ще простішим.

Можливо, потрібно пропустити деякі позиції ядра, щоб зменшити витрати на обчислення (внаслідок того, що функції не будуть отримані точно). Є можливість розглядати це як зменшення дискретизації виходу функції повної згортки. Якщо потрібно відібрати тільки кожні s пікселів у кожному напрямку на виході, тоді є можливість визначити функцію згортки c зі зниженою дискретизацією таким чином:

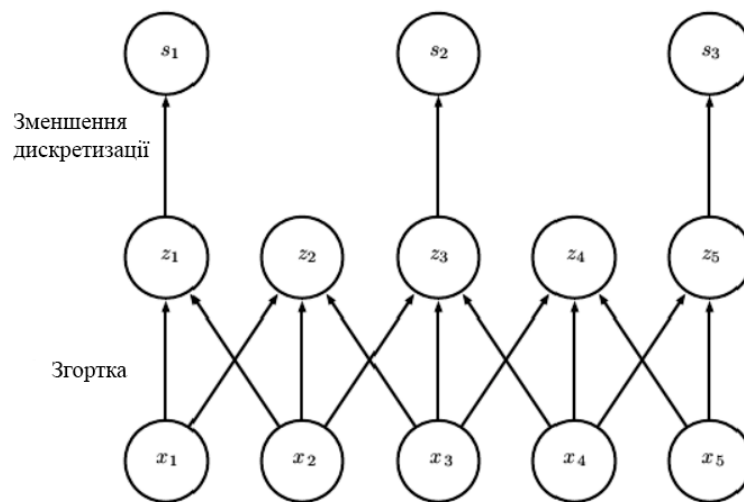
$$Z_{i,j,k}(s) = c(K, V, s)_{i,j,k} = \sum_{l,m,n} [V_{l,(j-1) \times s + m, (k-1) \times s + n} K_{i,l,m,n}], \quad (2.7)$$

Тут s – це крок цієї згортки зі зниженою дискретизацією. Також можна визначити окремий крок для кожного напрямку руху.

У прикладі, який показаний на рис. 2.15, використовується крок у два. Очевидно, що двоетапний підхід зі зменшенням дискретизації є марнотратним з точки зору обчислень, оскільки він обчислює багато значень, які потім відкидаються.



(a)



(б)

Рисунок 2.15 – Згортка з кроком. (а) Згортка з довжиною кроку два, реалізована за одну операцію. (б) Згортка з кроком більше одного пікселя математично еквівалентна згортці з одиничним кроком із наступним зменшенням дискретизації.

Однією з суттєвих особливостей реалізації будь-якої згорткової мережі є можливість неявного нульового заповнення входу V , щоб зробити його ширшим. Без цієї функції ширина представлення зменшується на один піксель відносно ширини ядра на кожному шарі. Нульове доповнення вводу дозволяє незалежно контролювати ширину ядра та розмір виводу. Без нульового доповнення потрібно вибирати між швидким скороченням просторової межі мережі та використанням

малих ядер. Обидва сценарії значно обмежують виражальну силу мережі, як на рис. 2.16.

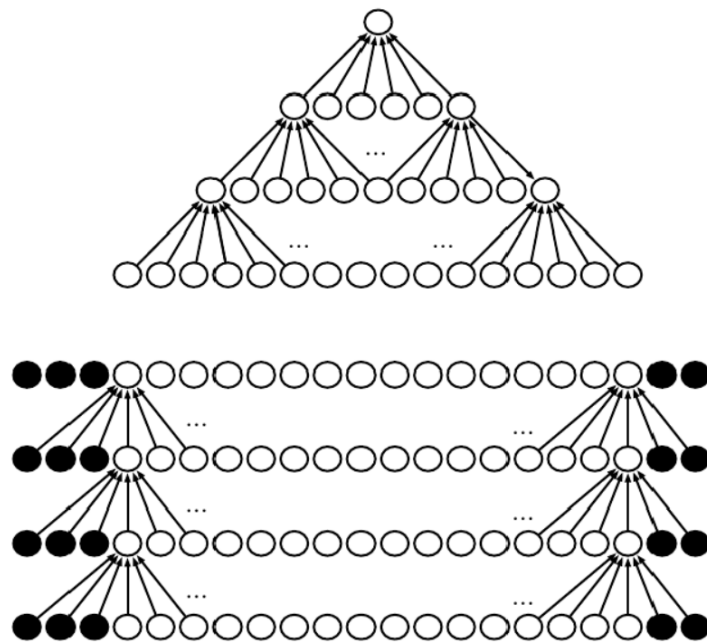


Рисунок 2.16 – Вплив нульового доповнення на розмір мережі: розглядається згортка мережі з ядром шириною в шість нейронів на кожному рівні [100].

У цьому прикладі не використовується об'єднання, тільки операція згортки зменшує розмір мережі. Зверху - у цій згортковій мережі не використовуються ніякі неявні нульові доповнення. Це призводить до того, що зображення зменшується на п'ять пікселів на кожному шарі. Починаючи з шістнадцяти пікселів, є можливість мати лише три згорткові шари, і останній шар ніколи не переміщує ядро, тож, лише два шари є справді згортковими. Швидкість скорочення можна зменшити, використовуючи менші ядра, але менші ядра менш виразні, і деяке скорочення є неминучим у такому типі архітектури. Знизу, додаючи п'ять неявних нулів до кожного шару, запобігається зменшення зображення з глибиною. Це дозволяє створити будь-яку глибину згорткової мережі.

Варто згадати три особливі випадки налаштування нульового доповнення. Перший — це крайній випадок, коли нульове доповнення не використовується

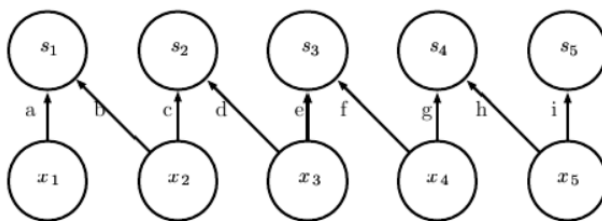
взагалі, і ядру згортки дозволено обробляти лише ті позиції, де все ядро повністю міститься в зображенні. У термінології *MATLAB* це називається допустимою згорткою. У цьому випадку всі пікселі на виході є функцією тієї самої кількості пікселів на вході, тому поведінка вихідного пікселя є дещо більш регулярною. Однак розмір виходу зменшується на кожному шарі. Якщо вхідне зображення має ширину m , а ядро має ширину k , вихід матиме ширину $m - k + 1$. Швидкість цього скорочення може бути суттєвою, якщо використовуються великі ядра. Оскільки стиснення більше ніж 0, воно обмежує кількість згорткових шарів, які можна включити в мережу. В міру додавання шарів просторовий розмір мережі в кінцевому підсумку впаде до 1×1 , після чого додаткові шари не можуть вважатися згортковими. Ще один особливий випадок параметра нульового доповнення, коли додається рівно стільки нульового доповнення, щоб розмір виходу дорівнював розміру входу. Для випадку, коли *MATLAB* викликає цю саму згортку, мережа може містити стільки згорткових шарів, скільки може підтримувати доступне обладнання, оскільки операція згортки не змінює архітектурні можливості, доступні для наступного рівня. Однак вхідні пікселі поблизу межі шару впливають на менше вихідних пікселів, ніж вхідні пікселі поблизу центру. Це може призвести до того, що пікселі рамки зображення дещо недостатньо представлені в моделі. Це мотивує інший крайній випадок, коли *MATLAB* викликає повну згортку, у якій додається достатня кількість нулів, щоб кожен піксель оброблявся k разів у кожному напрямку, що призводить до отримання вихідного зображення шириною $m + k - 1$. У цьому випадку, вихідні пікселі біля межі зображення є функцією меншої кількості пікселів, ніж вихідні пікселі поблизу центру. Це може ускладнити вивчення єдиного ядра, яке добре працює на всіх позиціях карти згорткових ознак. Зазвичай оптимальна кількість нульового доповнення (з точки зору точності класифікації набору тестів) лежить між дійсною та особливою згорткою.

У деяких випадках насправді не потрібно використовувати згортку, а скоріше локально з'єднані шари – [103]. У цьому випадку суміжна матриця на графі *MLP* однакова, але кожне з'єднання має власну вагу, визначену шестивимірним тензором

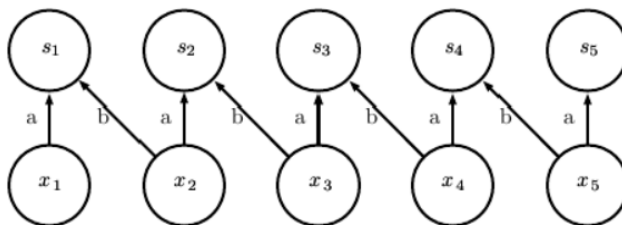
W . Індекси W відповідно: i – вихідний канал, j – вихідний рядок, k – вихідний стовпець, l – вхідний канал, m – зміщення рядка в межах входу, і n – зміщення стовця у межах входу. Тоді лінійна частина локально зв'язаного шару визначається як:

$$Z_{i,j,k} = \sum_{l,m,n} [V_{l,j+m-1,k+n-1} W_{i,j,k,l,m,n}], \quad (2.8)$$

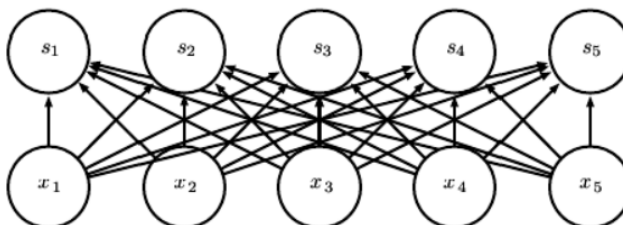
Це іноді також називають нерозділеною згорткою, оскільки це операція, подібна до дискретної згортки з малим ядром, але без спільного використання параметрів у різних місцях. На рисунку 2.17, порівнюються локальні зв'язання, згортка та повні зв'язання.



(a)



(б)



(в)

Рисунок 2.17 – Порівняння рівнів локальних зв'язків, згортки та повного зв'язку.

(а) Кожне ребро позначено унікальною літерою, щоб показати, що кожне ребро пов'язане з власним параметром ваги. (б) Згортковий шар із шириною ядра в два пікселі. (в) Повністю зв'язаний шар нагадує локально зв'язаний шар у тому сенсі, що кожне ребро має свій власний параметр (їх занадто багато, щоб явно позначати їх буквами на цій діаграмі). Однак він не має обмежених можливостей підключення, порівняно з локально підключеним рівнем.

Модель (б) має точно такий самий зв'язок, як і рівень локального підключення. Різниця полягає не в тому, які нейрони взаємодіють один з одним, а в тому, як розподіляються параметри. Локально підключений рівень не має спільного використання параметрів. Згортковий рівень неодноразово використовує ті самі два вагові коефіцієнти на всьому вході, про що свідчить повторення літер, що позначають кожне ребро.

Локально з'єднані шари корисні, коли є інформація, що кожна ознака має бути функцією невеликої частини простору, але немає жодних причин думати, що одна і та ж ознака має бути присутня в усьому просторі. Наприклад, якщо потрібно визначити, чи зображення є зображенням обличчя, нам потрібно лише шукати рот у нижній половині зображення.

Також може бути корисним створити версії згортки або локально підключених рівнів, у яких зв'язок додатково обмежений, наприклад, додатково обмежений кожним вихідним каналом і є функцією лише підмножини вхідних каналів l . Загальний спосіб реалізації такої згортки полягає в тому, щоб перші m вихідних каналів підключалися лише до перших n вхідних каналів, другі m вихідних каналів підключалися лише до других n вхідних каналів і так далі – рис. 2.15. Моделювання взаємодії між кількома каналами дозволяє мережі мати менше параметрів, щоб зменшити споживання пам'яті та підвищити статистичну ефективність, а також зменшує кількість обчислень, необхідних для виконання прямого та зворотного розповсюдження. Це досягається, не зменшуючи кількість прихованих нейронів.

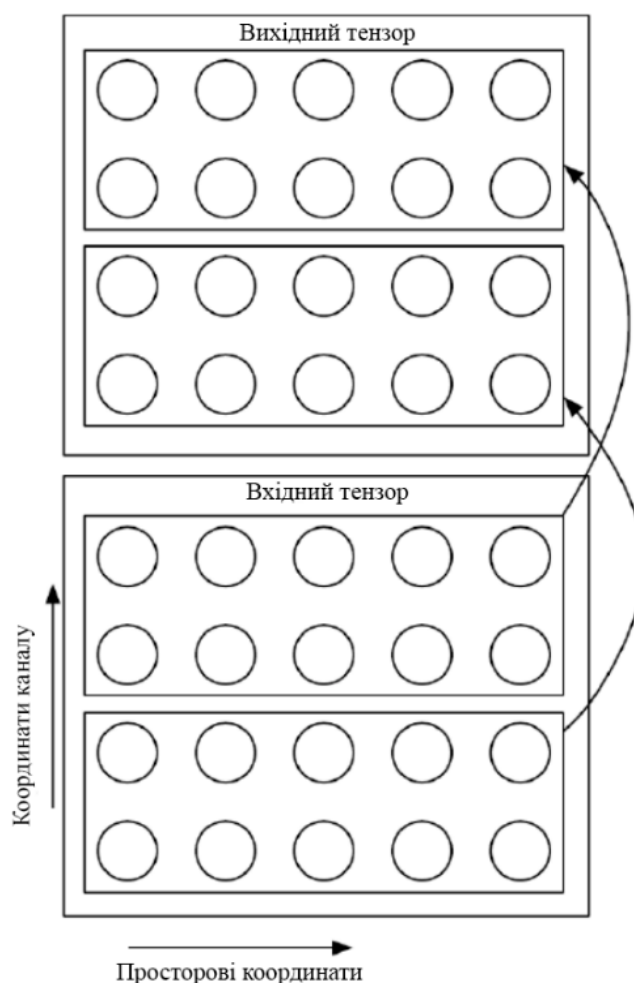
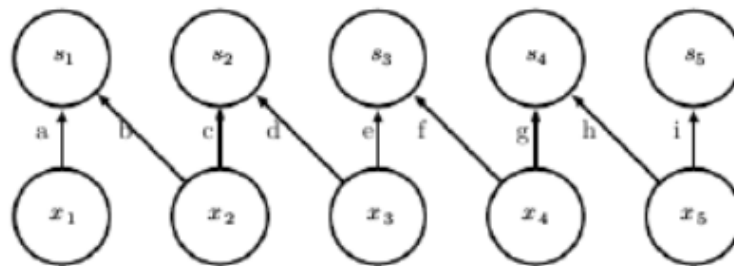


Рисунок 2.18 – Згорткова нейромережа з першими двома вихідними каналами, підключеними лише до перших двох вхідних каналів, а другі два вихідних канали підключені лише до других двох вхідних каналів.

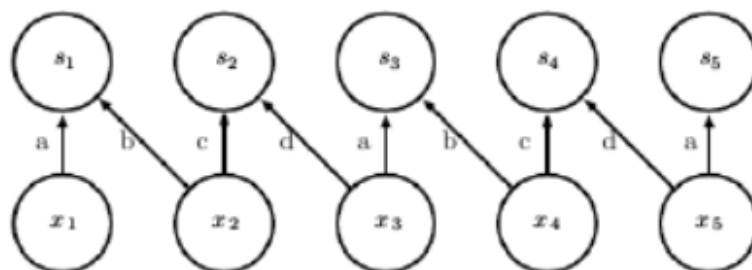
Мозаїчна згортка [104], [105] пропонує компроміс між згортковим шаром і локально зв'язаним шаром. Замість того, щоб вивчати окремий набір ваг у кожному просторовому місці, вивчаються набір ядер, які обертаються, коли йде рух у просторі. Це означає, що безпосередньо сусідні розташування матимуть різні фільтри, як у локально підключеному рівні, але вимоги до пам'яті для зберігання параметрів збільшаться лише на коефіцієнт розміру цього набору ядер, а не на розмір усієї вихідної карти ознак. На рис. 2.19 показано, для порівняння, локально з'єднані шари, плиткова згортка та стандартна згортка.

Усі три випадки мають однакові набори з'єднань між блоками, коли використовується однаковий розмір ядра. Ця діаграма ілюструє використання ядра

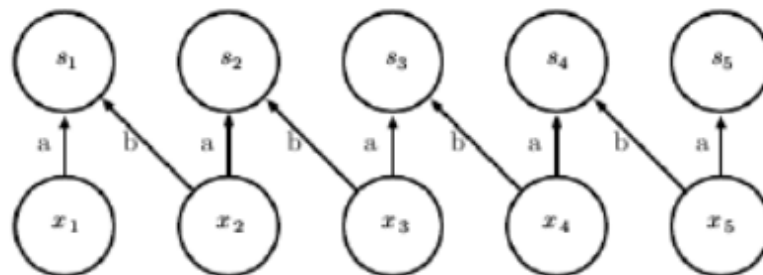
шириною у два пікселі. Відмінності між методами полягають у тому, як вони спільно використовують параметри.



(a)



(б)



(в)

Рисунок 2.19 – Порівняння локально з'єднаних шарів (а), мозаїчної згортки (б) та стандартної згортки (в) [104].

На рис. 2.19, (а), локально підключений рівень взагалі не має спільного доступу. Тут вказується, що кожне з'єднання має власну вагу, позначаючи кожне з'єднання унікальною літерою. На рис. 2.19, (б), мозаїчна згортка має набір із t різних ядер. Тут ілюструється випадок $t = 2$. Одне з цих ядер має ребра, помічені «а»

і «b», а інше має ребра, помічені як «c» і «d». Кожного разу, коли відбувається переміщення на один піксель праворуч на виході, відбувається перехід до використання іншого ядра. Це означає, що, як і локально підключений рівень, сусідні нейрони на виході мають різні параметри. На відміну від локально підключеного рівня, після того, як проходяться t усіх доступних ядер, відбувається повернення до першого ядра. Якщо два вихідних нейрони розділені кратністю в t кроків, то вони розділяють спільні параметри – показано знизу. Традиційна згортка еквівалентна мозаїчній згортці з $t = 1$. Існує лише одне ядро, і воно застосовується скрізь, як показано на схемі з ядром з вагами, які позначені всюди «a» і «b».

Щоб визначити мозаїчну згортку алгебраїчно, нехай k буде 6-вимірним тензором, де два виміри відповідають різним місцям у вихідній карті. Замість того, щоб мати окремий індекс для кожного розташування на вихідній карті, вихідні розташування циклічно проходять через набір із t різних варіантів стека ядра в кожному напрямку. Якщо t дорівнює вихідній ширині, це те саме, що локально підключений шар, тобто маємо:

$$Z_{i,j,k} = \sum_{l,m,n} V_{l,j+m-1,k+n-1} K_{i,l,m,n,j\%t+1,k\%t+1}, \quad (2.9)$$

де $\%$ — операція за модулем, з $t\%t = 0$, $(t + 1)\%t = 1$ тощо. Це рівняння є узагальненням використання різних діапазонів мозаїки для кожного виміру.

Як локально підключені шари, так і мозаїчні згорткові шари мають специфічну взаємодію з максимальним об'єднанням: блоки детектора цих шарів керуються різними фільтрами. Якщо ці фільтри навчаються виявляти різні трансформовані версії одних і тих самих базових функцій, то максимально об'єднані одиниці стають інваріантними до вивченого перетворення. Згорткові шари жорстко закодовані, щоб бути інваріантними саме до трансляцій (вивчених перетворень).

Інші операції, окрім згортки, зазвичай необхідні для реалізації згорткової мережі. Щоб виконати навчання, необхідно мати можливість обчислити градієнт відносно ядра, враховуючи градієнт відносно виходів.

У деяких простих випадках цю операцію можна виконати за допомогою операції згортки, але багато цікавих випадків, зокрема випадок кроку, який більший за 1, не мають цієї властивості.

Варто пам'ятати, що згортка є лінійною операцією, тому її можна описати як множення матриці (якщо спочатку змінимо форму вхідного тензора на звичайний вектор). Залучена матриця є функцією ядра згортки. Матриця є розрідженою, і кожен елемент ядра копіюється в декілька елементів матриці. Цей вигляд допомагає вивести деякі інші операції, необхідні для реалізації згорткової мережі.

Однією з таких операцій є множення на транспоновану матрицю, яка визначена згортою. Це операція необхідна для зворотного розповсюдження похідних помилок через згортковий рівень, і тому вона потрібна для навчання згорткових мереж, які мають більше одного прихованого рівня. Ця сама операція також потрібна, якщо ми хочемо реконструювати видимі нейрони з прихованих нейронів [106]. Реконструкція видимих нейронів — це операція, яка зазвичай використовується в моделях таких як автокодери, *RBM* (Обмежена машина Больцмана) і розріджене кодування.

Транспонована згортка необхідна для побудови згорткових версій цих моделей. Подібно до операції градієнта ядра, ця вхідна операція градієнта може бути реалізована за допомогою згортки в деяких випадках, але загалом вимагає реалізації третьої операції. Потрібно подбати про координацію цієї операції транспонування з прямим проходженням. Розмір виходу, який має повернути операція транспонування, залежить від поведінки нульового заповнення та кроку операції прямого поширення, а також від розміру вихідного представлення прямого поширення. У деяких випадках кілька розмірів вхідних даних для прямого розповсюдження можуть призвести до однакового розміру вихідного представлення, тому для операції транспонування, розмір початкових вхідних даних повинен бути представлений явно.

Цих трьох операцій — згортка, зворотне розповсюдження від виходу до вагів і зворотне розповсюдження від виходу до вхідних даних — достатньо для обчислення

всіх градієнтів, необхідних для навчання будь-якої глибокої прямої згорткової мережі, а також для навчання згорткових мереж із реконструйованими функціями на основі транспонування згортки. В роботі – [106] повне виведення рівнянь у загальному багатовимірному випадку з декількома прикладами. Щоб дати уявлення про те, як працюють ці рівняння, буде представлена двовимірною версією для простого прикладу.

Нехай потрібно навчити згорткову мережу, яка містить поступову згортку стека ядра K , застосовану до багатоканального зображення V з кроком s , визначеним $c(K, V, s)$ (див. рівняння 2.7). Припустимо, що потрібно мінімізувати деяку функцію витрат $J(V, K)$. Під час прямого розповсюдження нам потрібно буде використовувати саму s для виведення Z , яка потім поширюється через решту мережі та використовується для обчислення функції витрат J . Під час зворотного розповсюдження отримаємо такий тензор G , що:

$$G_{i,j,k} = \frac{\partial}{\partial Z_{i,j,k}} J(V, K), \quad (2.10)$$

Щоб навчити мережу, потрібно обчислити похідні щодо вагових коефіцієнтів у ядрі. Для цього є можливість використати функцію:

$$g(G, V, s)_{i,j,k,l} = \frac{\partial}{\partial K_{i,j,k}} J(V, K) = \sum_{m,n} G_{i,m,n} V_{j,(m-1) \times s + k, (n-1) \times s + l}, \quad (2.11)$$

Якщо цей рівень не є нижнім рівнем мережі, потрібно буде обчислити градієнт відносно V , щоб зворотно скорегувати помилку далі. Для цього є можливість використати функцію:

$$h(K, G, s)_{i,j,k} = \frac{\partial}{\partial V_{i,j,k}} J(V, K) = \sum_{l,m, (l-1) \times s + m = j} \sum_{n,p, (n-1) \times s + p = k} \sum_q K_{q,i,m,p} G_{q,l,n}. \quad (2.12)$$

Нейронні мережі – автокодери, є мережами прямого зв'язку, навченими копіювати вхідні дані у вихідні дані. Простим прикладом є алгоритм *PCA* (Метод основних компонент), який копіює вхід x до наближеної реконструкції r за

допомогою функції $W^T Wx$. Як правило, загальні автокодери використовують множення на транспонування вагової матриці, як це виконується в *PCA*. Щоб зробити такі моделі згортковими, є можливість використовувати функцію h для виконання транспонування операції згортки. Припустимо, що є приховані нейрони H в тому самому форматі, що й Z , і потрібно визначити реконструкцію:

$$R = h(K, H, s), \quad (2.13)$$

Щоб навчити автокодер, потрібно отримати градієнт відносно R у вигляді тензора E . Щоб навчити декодер, потрібно отримати градієнт відносно K . Це задано як $g(H, E, s)$. Щоб навчити кодер, потрібно отримати градієнт відносно H . Це задано як $c(K, E, s)$. Також можна диференціювати через g за допомогою c і h , але ці операції не потрібні для алгоритму зворотного поширення в будь-якій стандартній мережевій архітектурі.

Як правило, лінійна операція не використовується окремо для перетворення від входів до виходів у згортковому шарі. Зазвичай також додається деякий член зсуву до кожного виходу перед застосуванням нелінійності. Це підіймає питання про те, як розділити параметри між зміщеннями. Для локально з'єднаних шарів природно надавати кожному нейрону власне зміщення, а для мозаїчної згортки природно поділяти зміщення з тим самим шаблоном мозаїки, що і ядра. Для згорткових шарів є типовим мати одне зміщення на канал виходу і розподіляти його між усіма місцями в кожному представленні згортки. Однак, якщо вхід має відомий фіксований розмір, також можливо дізнатися окреме зміщення в кожному місці вихідного представлення. Розділення зміщень може дещо знизити статистичну ефективність моделі, але також дозволяє моделі коригувати відмінності в статистиці зображення в різних місцях. Наприклад, при використанні неявного нульового доповнення блоки детектора на краю зображення отримують менший загальний вхідний сигнал і можуть потребувати більших зміщень.

2.4.2 Структурований вихід нейронної мережі та типи даних

Згорткові мережі можна використовувати для виведення багатовимірного структурованого об'єкта, а не просто для прогнозування класу в завданнях класифікації або справжнього значення в завданнях регресії. Зазвичай цей об'єкт є просто тензором, створеним стандартним згортковим шаром. Наприклад, модель може видавати тензор S , де $S_{i,j,k}$ є ймовірністю того, що піксель (j, k) входу мережі належить до класу i . Це дозволяє моделі позначати кожний піксель на зображенні та малювати точні маски, які повторюють контури окремих об'єктів.

Проблема, яка часто виникає, полягає в тому, що вихідна площа може бути меншою за вхідну – рис. 2.20. У типах архітектур, які зазвичай використовуються для класифікації окремого об'єкта в зображенні, найбільше зменшення просторових розмірів мережі відбувається завдяки використанню шарів об'єднання з великим кроком. Щоб створити вихідне представлення такого ж розміру, як і вхідне, можна взагалі уникнути об'єднання [104]. Інша стратегія полягає в тому, щоб просто отримувати представлення міток з нижчою роздільною здатністю [105]. Нарешті, в теорії, є можливість використовувати оператор об'єднання з одиничним кроком.

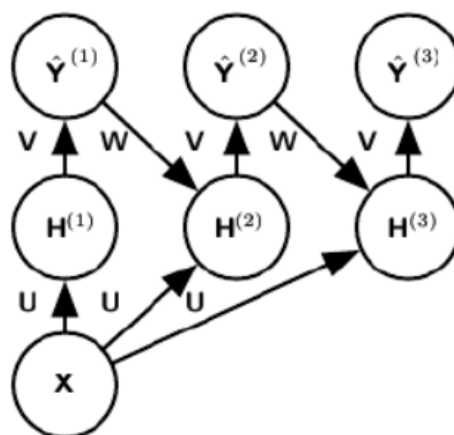


Рисунок 2.20 – Приклад рекурентної згорткової мережі для маркування пікселів [100].

Вхідними даними є тензор зображення X з осями, що відповідають рядкам зображення, стовпцям зображення та каналам (червоний, зелений, синій). Мета полягає в тому, щоб вивести тензор міток $\hat{Y}$ із розподілом ймовірностей за мітками для кожного пікселя. Цей тензор має осі, що відповідають рядкам зображення, стовпцям зображення та різним класам. Замість того, щоб виводити $\hat{Y}$ за одну ітерацію, рекурентна мережа ітераційно оцінює $\hat{Y}$, використовуючи попередню оцінку $\hat{Y}$, як вхідні дані для створення нового оцінювання. Ті самі параметри використовуються для кожного оновленого оцінювання, і тому є можливість уточнювати скільки завгодно разів. Тензор ядер згортки U використовується на кожному кроці для обчислення прихованого представлення, заданого вхідним зображенням. Тензор ядра V використовується для отримання оцінювання міток із заданими прихованими значеннями. На всіх кроках, крім першого, ядра W згортаються над $\hat{Y}$, щоб забезпечити вхідні дані для прихованого шару. На першому часовому кроці цей член замінюється нулем.

Однією зі стратегій піксельного маркування зображень є створення початкового припущення міток зображення, а потім уточнення цього початкового припущення за допомогою взаємодії між сусідніми пікселями. Повторення цього етапу уточнення кілька разів відповідає використанню однакових згорток на кожному етапі, розподіляючи ваги між останніми шарами глибокої мережі [104]. Це робить послідовність обчислень, що виконуються послідовними згортковими шарами з вагами, розподіленими між шарами, певним типом рекурентної мережі [105]. Архітектура такої рекурентної згорткової мережі показана на рис. 2.20.

Після того, як зроблено прогноз для кожного пікселя, можна використовувати різні методи для подальшої обробки цих прогнозів, щоб отримати сегментацію зображення на області [110], [111], [112].

Загальна ідея полягає в припущенні, що великі групи суміжних пікселів мають тенденцію бути пов'язаними з тією самою міткою. Графічні моделі можуть описувати ймовірнісні зв'язки між сусідніми пікселями. В альтернативі, є

можливість навчити згорткову мережу таким чином, щоб максимізувати наближення до цілі навчання графічної моделі [113], [114].

Дані, які використовуються в згортковій мережі, зазвичай складаються з кількох каналів, кожен з яких є спостереженням іншої величини в певній точці простору чи часу. В таблиці наведені приклади з типами даних з різною розмірністю та кількістю каналів.

Насправді розмірність об'єктів, які можуть бути використані у процесах згортки може бути будь-якою, все залежить тільки від того, як правильно представити ці дані.

Таблиця 2.1. Приклади різних форматів даних, які можна використовувати в згорткових мережах

Розмірність даних	Одноканальний	Багатоканальний
1D	Форма аудіосигналу: основна вісь – відлік часу. Відбувається дискретизація часу і вимірюється амплітуда форми сигналу один раз за часовий крок.	Дані про анімацію скелета: анімація тривимірних персонажів, відтворених комп'ютером, створюється шляхом зміни пози «скелету» з часом. У кожен момент часу поза персонажа описується специфікацією кутів кожного з суглобів у скелеті персонажа.
2D	Аудіо дані, попередньо оброблені за допомогою перетворення Фур'є: є можливість перетворити форму аудіо сигналу на двовимірний тензор де рядкам	Дані кольорового зображення: один канал містить червоні пікселі, другі зелені пікселі та треті сині пікселі. Ядро згортки рухається по горизонтальній і вертикальній осях зображення, надаючи

	відповідають різні частотам, а різним стовпцям відповідають різні моменти часу.	еквівалентність трансляції в обох напрямках.
3D	Об'ємні дані: поширеним джерелом таких даних є технологія медичної візуалізації, наприклад, комп'ютерна томографія.	Кольорові відеодані: одна вісь відповідає часу, друга – висоті відеокадру, а третя – ширині відеокадру.

Поки що було обговорено лише випадок, коли кожен приклад у даних тренування та тестування мають однакові просторові розміри. Однією з переваг згорткових мереж є те, що вони також можуть обробляти вхідні дані з різними просторовими обсягами. Такі вхідні дані просто не можуть бути представлені традиційними нейронними мережами на основі множення матриць. Це є вагомою причиною для використання згорткових мереж, навіть якщо витрати на обчислення та переобладнання є суттєвими проблемами.

Наприклад, розглянемо колекцію зображень, де кожне зображення має різну ширину та висоту. Незрозуміло, як моделювати такі вхідні дані за допомогою вагової матриці фіксованого розміру. Згортка проста у застосуванні; ядро просто застосовується різною кількістю разів залежно від розміру вхідних даних, і результат операції згортки відповідно масштабується. Згортку можна розглядати як множення матриць; те саме ядро згортки змінює різний розмір подвійної блокової матриці для кожного розміру вхідних даних. Іноді вихідні дані мережі можуть мати змінний розмір, а також вхідні дані. Наприклад, якщо потрібно призначити мітку класу кожному пікселю вхідних даних. В інших випадках мережа повинна створювати вихідні дані фіксованого розміру, наприклад, якщо потрібно призначити одну мітку класу всьому зображенню. У цьому випадку треба зробити деякі додаткові кроки проєктування, як вставлення рівня об'єднання, розміри областей об'єднання якого пропорційні розміру вхідних даних, щоб підтримувати фіксовану кількість об'єднаних виходів – рис. 2.14.

Зауважимо, що використання згортки для обробки вхідних даних змінного розміру має сенс лише для вхідних даних змінного розміру при наявності різної кількості спостережень однакового типу, різну довжину записів у часі, різну шкалу спостережень у просторі тощо. Згортка не має сенсу, якщо вхідні дані мають змінний розмір при додатковому включенні іншого виду спостережень. Наприклад, якщо обробляються заявки на вступ до університету, і наші характеристики складаються як з оцінок, так і зі стандартизованих результатів тестування, але не кожен абітурієнт склав стандартизований тест, тоді немає сенсу згортати однакові ваги для обох характеристик, що відповідають оцінкам і функціям, що відповідають результатам тесту.

2.4.3 Ефективні алгоритми згортки

Сучасні згорткові мережеві програми часто включають мережі, що містять більше одного мільйона нейронів. Потужні реалізації, що використовують ресурси паралельних обчислень, є вкрай важливими. Однак у багатьох випадках є можливість прискорити згортку, вибравши правильний алгоритм згортки.

Згортка еквівалентна перетворенню вхідних даних і ядра в частотну область за допомогою перетворення Фур'є, виконання поточкового множення двох сигналів і перетворення назад у часову область за допомогою зворотного перетворення Фур'є. Для деяких типів задач це є набагато швидшим та ефективнішим рішенням, ніж проста реалізація дискретної згортки.

Якщо d -вимірне ядро можна виразити як тензорний добуток d векторів по одному вектору на вимір, ядро називається роздільним. Коли ядро роздільне, проста згортка неефективна. Це еквівалентно складанню d одновимірних згорток з кожним із таких векторів. Такий підхід є значно швидшим, ніж виконання однієї d -вимірної згортки з її тензорним добутком. Ядро також містить менше параметрів для

представлення у вигляді векторів. Якщо ядро має ширину w елементів у кожному вимірі, тоді звичайна багатовимірна згортка потребує $O(w^d)$ часу виконання та простору для зберігання параметрів, тоді як роздільна згортка вимагає $O(w \times d)$ часу виконання та простору для зберігання параметрів. Звичайно, не кожному згортку можна представити таким чином.

Розробка швидших способів виконання згортки або наближеної згортки без узгодження точності моделі є активною областю досліджень. Навіть методи, які підвищують ефективність лише прямого розповсюдження є корисними, оскільки в загальних умовах типово виділяти більше ресурсів на розгортання мережі, ніж на її навчання.

2.4.4 Випадкові ознаки або ознаки без супервізора

Як правило, найгрозоміздкішою частиною навчання згорткової мережі є вивчення ознак. Вихідний рівень зазвичай є відносно невеликим через невелику кількість ознак, які надаються як вхідні дані для цього рівня після пропускання через кілька рівнів об'єднання. Під час навчання з вчителем із градієнтним спуском кожен градієнтний крок вимагає повного пропускання прямого та зворотного розповсюдження всією мережею. Один зі способів зменшити громіздкість навчання згорткової мережі є використання ознак, які не отримуються з навчання з супервізором.

Існує три основні стратегії для отримання ядер згортки без навчання з супервізором. Один з них - ініціалізувати їх випадковим чином. Інший варіант — розробити їх вручну, наприклад, налаштувавши кожне ядро на виявлення країв у певній орієнтації чи масштабі. Нарешті, є можливість вивчити ядра за допомогою неконтрольованого (без супервізора) критерію. Наприклад, в роботі [115], застосовано кластеризація k -середніх до невеликих ділянок зображення, а потім

використано використання кожного отриманого центроїда як ядра згортки. Вивчення ознак за неконтрольованим (без супервізора) критерієм дозволяє визначати їх окремо в класифікаційному шарі у верхній частині архітектури. Потім є можливість отримати ознаки для всього навчального набору лише один раз, по суті створюючи новий навчальний набір для останнього шару. Вивчення останнього шару зазвичай є проблемою опуклої оптимізації, припускаючи, що останній шар обробляється технікою на зразок логістичної регресії або *SVM* (метод опорних векторів).

Випадкові фільтри часто добре працюють у згорткових мережах - [116-119]. В роботі [117] показано, що шари, які складаються зі згортки, що йде після об'єднання, природним чином стають частотно-селективними та трансляційно-інваріантними, коли їм присвоюють випадкові ваги. В роботі [117] стверджується, що це забезпечує незатратний спосіб вибору архітектури згорткової мережі: спочатку оцінюється продуктивність кількох згорткових мережевих архітектур шляхом навчання лише останнього рівня, потім береться найкраща з цих архітектур і навчається вся архітектура, використовуючи більш затратний підхід.

Проміжним підходом є вивчення ознак, але з використанням методів, які не вимагають повного пропускання вперед і назад на кожному кроці градієнта. Як і у випадку з багатошаровими нейронами, використовується малозатратне попереднє навчання, щоб навчити перший шар ізольовано, потім витягуються всі ознаки з першого шару лише один раз, потім навчається другий рівень ізольовано, враховуючи ці ознаки, і так далі. Канонічним прикладом незатратного пошарового попереднього навчання згорткової моделі є згорткова нейромережа глибоких переконань – [120]. Згорткові мережі дають можливість виконати стратегію попереднього навчання на один крок вперед, ніж це можливо з багатошаровими нейронами. Замість того, щоб тренувати весь згортковий шар за раз, є можливість тренувати модель невеликої ділянки, наприклад, - [115] для випадку з k -середніми.

Потім є можливість використовувати параметри з цієї моделі, щоб визначити ядра згорткового шару. Це означає, що можна використовувати неконтрольоване навчання для навчання згорткової мережі без використання згортки під час процесу навчання. Використовуючи цей підхід, є можливість навчити дуже великі моделі та мати високі витрати на обчислення лише під час виведення - [121-124]. Цей підхід був популярний приблизно з 2007–2013 років, коли мічені набори даних були невеликими, а обчислювальна потужність була більш обмеженою. Сьогодні більшість згорткових мереж навчаються суто контрольованим способом, використовуючи повне пряме та зворотне поширення через всю мережу на кожній ітерації навчання.

Як і в інших підходах до неконтрольованого попереднього навчання, залишається важко зрозуміти причину деяких переваг, які спостерігаються за допомогою цього підходу. Попереднє навчання без супервізора може запропонувати певну регулярність порівняно з навчанням з супервізором, або воно може просто дозволити навчати набагато більші архітектури завдяки меншій обчислювальній затратності навчання.

Використовуючи різні підходи до використання згортки та об'єднання, є можливість вирішувати різного типу задачі, які, як правило, пов'язані від різного виду даних – зображення, відео та інше.

Висновки до розділу 2.

В першому підрозділі розглянуто та описано основні підходи до побудови архітектури нейронних мереж для задач прогнозування та моделювання багатofункціональних залежностей. Зазначено про апроксимативну теорему: будь-яку складну нейронну мережу з необхідною точністю є можливість замінити (апроксимувати) нейронною мережею з одним прихованим шаром і необхідною кількістю нейронів в цьому шарі, а також, що деякі задачі більш ефективно моделюються нейронними мережами з кількістю прихованих шарів більше ніж один.

Вказано, що на даний час не існує чітких теорій, а тільки відокремленні дослідження різного типу задач, тобто – яка кількість прихованих шарів та кількість нейронів має бути в нейронних мережах, які створені розв'язувати поставлені задачі, тому що кожна задача є досить унікальною, щоб досліджувати архітектуру нейронних мереж тільки методом зменшення похибки.

Зауважено, що широкий розвиток застосування нейронних мереж до прикладних задач дає змогу знайти майже будь-яке близьке за типом задачі архітектурне рішення, і що, як правило, залишається видозмінити або доповнити наявне рішення для своєї конкретної задачі. Задача, яка розв'язувалась в цій роботі, використовувала архітектуру у вигляді 3D-згортки відповідної програмної бібліотеки, але саме таке архітектурне рішення (які шари, кількість нейронів, кількість шарів, гіперпараметри) підбирались звичайним емпіричним методом – методом найменшої похибки.

В другому підрозділі описано, як відрізняється звичайний процес згортки та процес згортки залучений в нейронних мережах.

В третьому підрозділі описано, що в нейронних мережах використовуються декілька додаткових (не завжди необхідних) процедур – об'єднання (pooling), виявлення (detection).

В четвертому підрозділі описано основні ідеї в залученні процесу згортки в нейронних мережах: розріджені взаємодії, спільне використання параметрів та еквіваріантні представлення. Також потрібно пам'ятати, що згортка може використовуватись, коли елементи вибірки у вхідних даних мають змінний розмір. В цілому згорткові нейронні мережі застосовуються в задачах розпізнавання образів в 2D, 3D-об'єктах, а також обробці аудіо сигналів. Основні принципи застосування процесу згортки (та додаткових) були описані в другому підрозділі.

РОЗДІЛ 3

ПІДГОТОВКА 3D-СТРУКТУР ТА ДАНИХ ДЛЯ НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ

3.1 Методи побудови 3D-структур та складнощі конвертації форматів

Існують різні підходи до побудови 3D-моделей. Сутність процесу побудови 3D-моделей для навчання нейронних мереж полягає у формуванні зручного для нейронних мереж цифрового представлення всієї інформації про 3D-моделі. Молекулярна структура матеріалів може бути представлена у цифровому вигляді, здатному для аналізу штучного інтелекту та поставлена у відповідність істотним параметрам метаматеріалів, які треба вивчати. Основна задача полягає у поставленні строгої відповідності – інформація про структуру (матеріал) – фізична характеристика структури (матеріалу). Звичайно, існує нескінченна кількість чинників, які впливають на експериментальну фізичну характеристику, але точно зрозуміло, що чим більше є інформації про структуру та інші параметри, які впливають на фізичну характеристику, тим менший потрібен розмір вибірки для навчання нейронної мережі.

Для прикладу було представлено деяку кількість робіт – підрозділ 1.2, де було використано різну інформацію про структуру. Теорія нейронних мереж була достатньо розроблена, щоб представляти різні типи залежностей – часові, просторові. Як правило, для всіх випадків був вибраний такий тип формування вибірки, при якому можна було в короткий час сформувати велику кількість інформації. Наприклад, 2D-структури з умовами вимірювання, або параметри електромагнітних метаструктур з деякими просторовими властивостями. Підхід представлений в дисертаційному дослідженні полягає у формуванні цілісної інформації про структуру метаматеріалу, а також про склад метаматеріалу у вигляді

інформації про фізичні або хімічні властивості компонентів, з яких складається структура.

Видів 3D моделювань є багато, але зазначимо тільки декілька:

Полігональне моделювання

Це найстаріший метод моделювання, за допомогою якого можна створити абсолютно будь-яку форму, але це не є точним наближенням. Полігональне моделювання заощаджує ресурси комп'ютера, тому моделі, створені таким способом, частіше використовуються для 3D візуалізації. Оскільки, точність такого методу низька, в деяких випадках потрібно використовувати інші більш точні методи [97].

Найбільшим недоліком полігонального моделювання є те, що всі об'єкти повинні складатися з малих плоских поверхонь, а полігони повинні мати дуже малий розмір, інакше краї об'єкта будуть мати огранований вигляд. Це означає наступне: якщо для об'єкта передбачається збільшення, його необхідно моделювати з великою кількістю полігонів високою щільністю), попри те, що більшість із них буде зайвою при наступних редагуваннях – див. рис. 3.1.

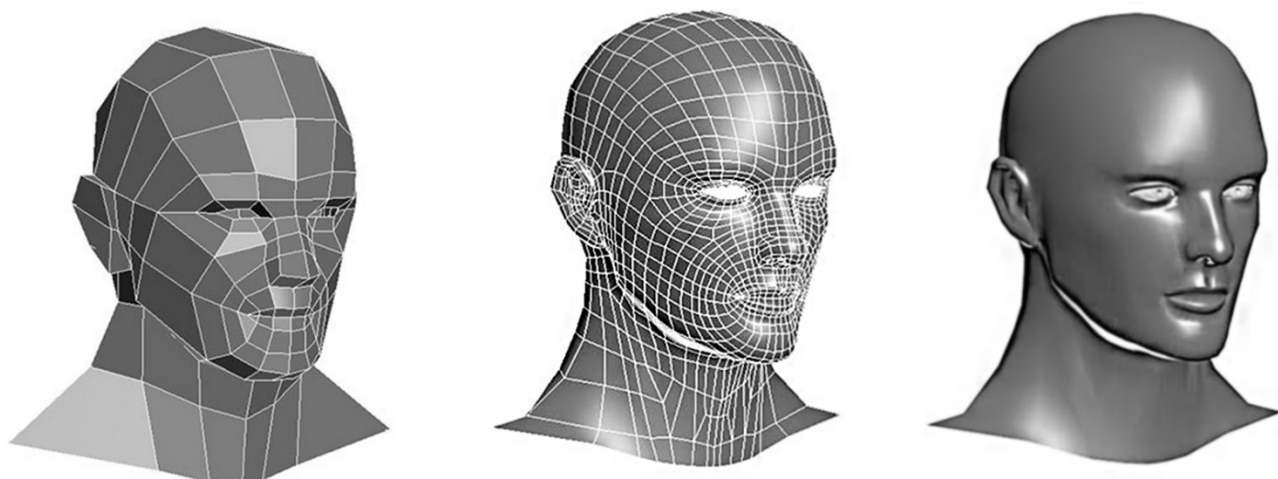


Рисунок 3.1 – Приклад моделювання 3D-об'єкта полігональним методом.

Векторне (NURBS) моделювання

Широко використовується в точному машинобудуванні, оскільки кожна лінія або поверхня є функцією координат і цю функцію можна точно обчислити в будь-якій точці поверхні. Це дозволяє створювати максимально точну форму моделі незалежно від її фізичного розміру. Можна створювати велику кількість різноманітних геометричних та біологічних форм.

На відміну від растрової графіки, в якій зображення представлене у вигляді графічної матриці, що складається з пікселів, векторна графіка заснована на математичному описі елементарних геометричних об'єктів, які зазвичай називають «примітивами», таких як: точки, лінії, сплайни, криві Безьє, кола та кола, багатокутники [99].

Основна відмінність цього методу від полігонального моделювання полягає в плавності. *NURBS* модель складається не з полігонів, а з кривих ліній, проте при візуалізації, об'єкт все одно перетворюється на полігони, хоча всередині системи моделювання він залишається побудованим в кривих лініях.

В залежності від задачі, можуть використовуватись обидва способи створення 3D-моделей. Все залежить від обчислювальних можливостей; необхідної кількості інформації про об'єкт для того, щоб нейронна мережа виконувала свою задачу відповідно; часу на підготовку всіх 3D-моделей для вхідної вибірки.

Інша річ яку варто зазначити, це складність у конвертації інформації про 3D-структуру у вигляді, потрібному для навчання нейронної мережі. Знову ж таки все залежить від того, як поставлена задача, і чи правильно поставлена відповідність: вхідні дані – вихідні дані.

Для скорочення кількості вхідних нейронів для нейронної мережі, зазвичай використовують операцію згортки по всім даним для того, щоб зберегти інформацію про відмінності в елементах вибірки, одночасно при цьому зменшивши навантаження на обчислювальну машину.

В цілому величина вхідних даних може бути достатньо великою, потрібно лише враховувати можливості згортки цих даних і представленні їх у потрібному для нейронної мережі вигляді.

Також, важливо представити дані у потрібному для згортки вигляді. В залежності від того, яким методом створювались 3D-моделі, та в якому програмному середовищі будуть створюватись 3D-моделі, буде залежати чи є взагалі можливість конвертації даних у потрібну форму для згортки. Більшість програмних середовищ мають свої унікальні формати для збереження 3D-об'єктів.

Для зберігання інформації про 3D структури метаматеріалів доцільно використати наступні формати:

1) *STL*

STL - це скорочення від "Stereolithography", один із найстаріших форматів файлів. Він описує геометрію поверхні 3D-об'єкта, не відображаючи колір, текстуру чи інші атрибути. Формат *STL* має розширення «.stl». Ці файли генеруються програмою автоматизованого проєктування (*CAD*). Для редагування файлів *STL* можна використовувати такі програми, як *FreeCAD*, *Blender*, *MeshLab*, *MeshMixer*, *SketchUp*, *SculptGL* та *3DSlash*.

STL-файл спрощує 3D-поверхню в теселяцію – серію маленьких трикутників, кількість яких збільшується, коли потрібно якнайкраще відтворити вигнуті поверхні. Коли потрібно використовувати велику кількість трикутників, розмір *STL*-файлу 3D-моделі швидко збільшується.

Основні характеристики:

- *STL* записується як апроксимована геометрія поверхні 3D моделі з використанням трикутної сітки.
- *STL* зберігає тільки геометрію моделі та, отже, є одним із найпростіших і компактних форматів 3D файлів.

- Формат *STL* представляється як в кодуванні *ASCII*, так і у бінарному представленні. Бінарні файли більш поширені, оскільки вони компактніші.

2) ***OBJ***

Його розширення – «.obj». Його перевага полягає в тому, що він також кодує інформацію про колір та текстуру, яка зберігається в окремому файлі з розширенням «.mtl». Файли *OBJ* дозволяють використовувати не тільки трикутні грані з примиканням однієї грані до іншої. Їх можна відкрити за допомогою таких програм, як *Autodesk Maya*, *Blender* та *MeshLab*.

Основні характеристики:

Формат файлу *OBJ* підтримує як апроксимований, так і точний запис геометрії поверхні. При використанні першого варіанта сітка поверхні не обмежується трикутними гранями. Якщо користувач хоче, він може використовувати інші багатокутники, наприклад, чотирикутники. При використанні точного запису використовуються плавні криві та поверхні, такі як *NURBS*.

Формат *OBJ* також може містити інформацію про колір та текстуру. Ця інформація зберігається в окремому файлі з розширенням «.mtl» (бібліотека шаблонів матеріалів). Файл *OBJ* не підтримує запис інформації про анімацію.

3) ***FBX***

Підтримує геометрію та властивості 3D-об'єкту, такі як колір та текстура, а також скелетну анімацію. *AutoDesk* використовує файл *FBX* з розширенням «.fbx», як обмінний формат для свого портфеля програмного забезпечення, такого як *AutoCAD*, *Fusion 360*, *Maya*, *3DS Max* та ін.

4) ***DAE (Collada)***

Файли *Collada* належать до нейтрального формату.

Основні характеристики:

Формат *Collada* підтримує геометрію та властивості, пов'язані із зовнішнім виглядом, такі як колір, матеріал, текстура та анімація. Крім того, це один із рідкісних форматів, що підтримують кінематику та фізику. *Collada* зберігає дані з використанням мови розмітки *XML*.

5) *3DS*

3DS - це формат файлів, який зберігає тільки саму основну інформацію про геометрію, зовнішній вигляд, сцени та анімацію. Він використовує трикутну сітку для апроксимованої геометрії поверхні, загальна кількість трикутників обмежена числом 65 536. Він дозволяє зберігати такі властивості, як колір, матеріал, текстура, пропускну здатність тощо. Є можливість також зберігати джерела світла, але формат не підтримує спрямовані джерела світла. Формат *3DS* є бінарним та зберігає інформацію фрагментами. Це дозволяє синтаксичним аналізаторам пропускати фрагменти, які вони не розпізнали, та дозволяє розширювати формат. Маючи розширення «.3ds», цей формат файлів для 3D-друку також має ту перевагу, що його сприймають більшість наявних популярних програм, таких як *3dsMax*, *ABViewer*, *Blender*, *MeshLab*, *messiahStudio*, *Rain Swift 3D*, *SketchUp*, *TurboCAD* та ін.

6) *PLY*

PLY – формат файлів опису геометрії, відомий також як *Polygon File Format* та *Stanford Triangle Format*. Він був розроблений головним чином для зберігання тривимірних даних для 3D сканерів. Формат підтримує відносно простий опис об'єкта як список плоских полігонів. *PLY* може зберігати безліч властивостей об'єкта, що включає: колір та прозорість, нормалі до поверхні, текстурні координати й т. д. Формат дозволяє мати різні властивості передньої та задньої грані полігону. Існує дві версії формату *PLY*: *ASCII* та у вигляді бінарного файлу.

Вибір потрібного формату залежить від декількох факторів:

- 1) Можливості конвертації у потрібний для згортки формат;

- 2) Додаткової інформації про об'єкт, яку потрібно зберегти (напр. колір, анімація, текстури і т. д.);
- 3) Обсяг пам'яті, який займає інформація про об'єкт у цьому форматі;
- 4) Існування програмного API (*Application Programming Interface*).

Останній фактор означає те, що унікальність задачі ставить різні потреби до вигляду вхідних даних, і не завжди існує можливість конвертувати формат в потрібний вигляд. Звичайно, з розвитком інформаційних технологій ця проблема вже не така суттєва, але якщо задача стоїть таким чином, що немає можливості отримати дані у потрібному вигляді, то завжди є варіант зі створення власного програмного алгоритму по конвертації даних. Це пов'язано з тим, що будь-які дані можна отримати у форматі *ASCII* або іншому. Якщо є інформація про те, як саме представляються дані про 3D-об'єкт та інші (текстура, колір і т. д.), то завжди є можливість побудувати власний алгоритм конвертації даних.

Для розв'язку задач роботи всі 3D-об'єкти були збережені у форматі «.ply», тому що цей формат зберігає інформацію про структуру (координати всіх пікселів та *RGB* значення «колір пікселя») у найбільш зручному кодуванні даних та існування програмного *API* для конвертації у зручний для задачі вигляд.

3.2 Реалізація властивостей при створенні цифрових 3D-структур

В цьому розділі буде описано процедури розв'язку таких задач:

- 1) Побудова цифрових 3D-моделей метаструктур.
 - 2) Цифрове закріплення властивостей матеріалів за координатами пікселів.
 - 3) Інтерпретація 3D-моделей у потрібному вигляді.
- 1) *Побудова цифрових 3D-моделей.*

Була проведена у програмному середовищі Blender. Дані для побудови були взяті у наступних роботах [92-96]. Була можливість збільшити кількість джерел реальних

даних, але необхідні умови до набору даних для нейронної мережі в такому випадку могли б бути не задовільнені. Умови, які накладались на роботи [92-96] виглядають так:

- Вказані всі топологічні характеристики метаструктур.
- Вказані всі матеріали, з яких побудовані метаструктури.
- Експериментальні характеристики однакового типу.
- Умови, при яких проводились експериментальні дослідження мають бути однаковими або ж, різниця між ними має бути врахована при навчанні нейронної мережі.

Дані про метаматеріали, представлені на рис. 3.2-3.6. Тут показні всі розміри потрібні для побудови 3D структур в програмному пакеті для створення тривимірної графіки.

Всі структури були використані у формуванні даних для навчання нейронної мережі. Вибірки даних формувались окремо в залежності від частотних діапазонів, в яких були дослідженні дані зразки.

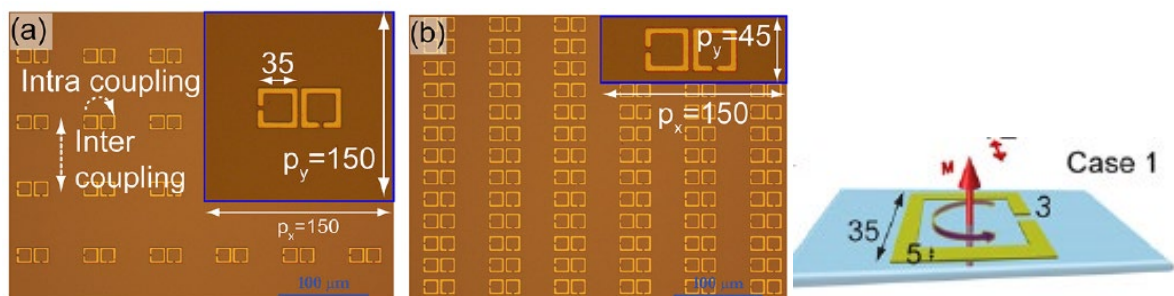


Рисунок 3.2 – Схематичний вигляд метаматеріалу - 1. Всі одиниці в мікрометрах. Товщина антен 1 мкм. Товщина підкладки 5 мкм. (а) $p_y = 150$ мкм. (б) $p_y = 45$ мкм [92].

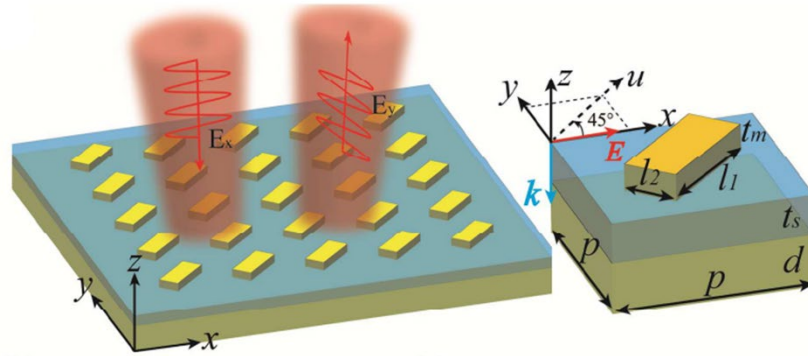


Рисунок 3.3 - Схематичний вигляд метаматеріалу - 2: $p = 600$ нм, $l_1 = 390$ нм, $l_2 = 165$ нм, $t_m = 80$ нм, $t_s = 110$ нм, $d = 150$ нм [93].

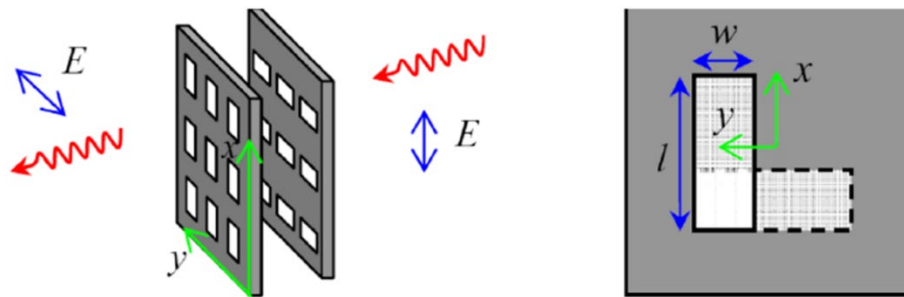


Рисунок 3.4 - Схематичний вигляд метаматеріалу - 3. $l = 750$ нм, $w = 280$ нм, $t = 150$ нм (товщина металічного шару), $d = 1000$ нм (відстань між вирізами) h - товщина діелектричного шару [94].

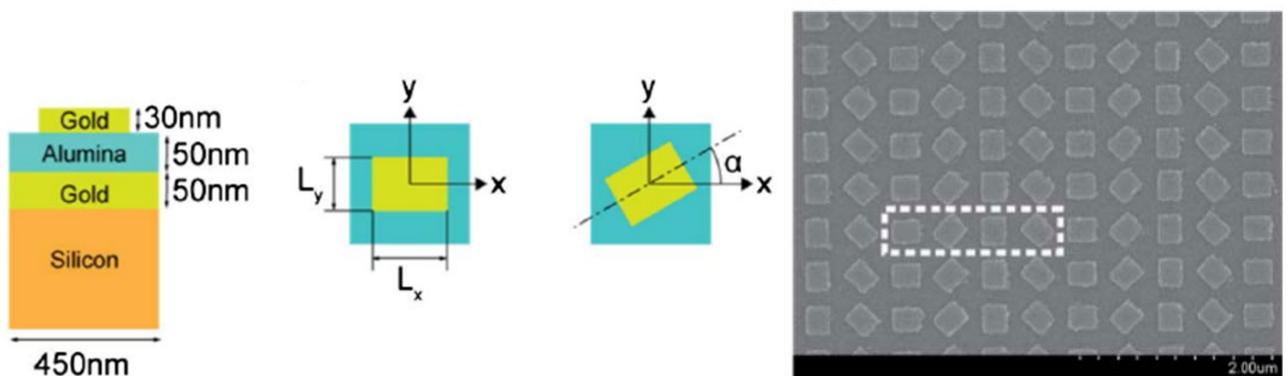


Рисунок 3.5 - Схематичний вигляд метаматеріалу - 4. $\alpha = 0^\circ; 45^\circ; 90^\circ; 135^\circ$. $L_x = 280$ нм, $L_y = 230$ нм [95].

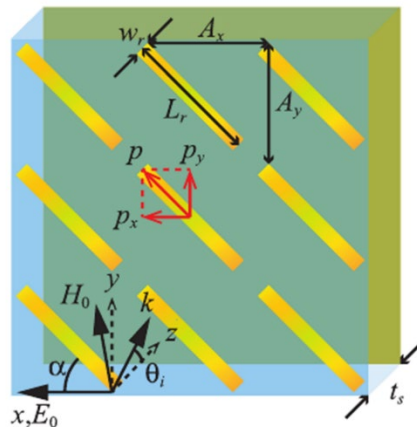


Рисунок 3.6 - Схематичний вигляд метаматеріалу - 5. $t_s = 33$ мкм, $A_x = A_y = 68$ мкм, $L_r = 82$ мкм, $w_r = 10$ мкм; товщина антен і шару золота 200 нм; товщина проміжку між смугами 10 мкм [96].

Оскільки більшість електромагнітних параметрів у різних діапазонах сильно залежать від частоти випромінювання, було взято середні значення по частоті для всіх зразків. Це звичайно приводить до не точного опису елементів метаструктур, але з іншого боку потрібно встановити фізичну величину, яка характеризує всі елементи для можливості навчання нейронної мережі на залежностях між різними фізичними значеннями величин елементів. Експерименти проводились для різних діапазонів – навчання відбувалось, як на даних про метаструктури, які досліджувались в різних частотних діапазонах, так і окремо. Метаструктури на рис. 3.2 та рис 3.6 досліджувались в діапазоні від 0.3 до 0.6 ТГц. Метаструктури на рис. 3.3, 3.4, 3.5 досліджувались в діапазоні від 131 до 374 ТГц.

Для структури, яка представлена на рис. 3.2, матеріал підкладки – кремній, а матеріал антен – золото. Для кремнію – діелектрична проникність – 12, магнітна проникність $\mu = 1,25 \cdot 10^{-6}$ (із загальних положень, оскільки кремній це діамagnetик). Для золота: $\epsilon = -74$, $\mu = -36 \cdot 10^{-6}$. Відносна провідність золота $\rho = 0,4 \cdot 10^8$ См/м. Інша інформація про структуру за посиланням - [92].

Для структури, яка представлена на рис. 3.3, жовтим кольором позначено золото, голубим – діоксид кремнію, та підкладка кремній. Для діоксиду кремнію діелектрична проникність – 2.1, магнітна проникність – 1 (із загальних положень, через те що діоксид кремнію це діамagnetик). Діелектрична проникність для золота була взята, як середнє значення по частоті (тому що для частот експерименту – ближній інфрачервоний діапазон, діелектрична проникність змінюється в залежності від частоти), тому $\epsilon = -74$. Магнітна проникність золота: $\mu = -36 \cdot 10^{-6}$. Відносна провідність золота $\rho = 0,4 \cdot 10^8$ См/м. Інша інформація про структуру за посиланням - [93].

Для структури, яка представлена на рис. 3.5, жовтим кольором позначено золото, синім діелектричний заповнювач (Al_2O_3) та оранжевий кремній. Для золота значення діелектричної, магнітної проникності та відносна провідність такі ж, як для структури на рис. 3.3. Діелектричний заповнювач має такі властивості $\epsilon = 10$, $\mu = 1,25 \cdot 10^{-6}$ [95].

Для структури, яка представлена на рис. 3.6, жовтим позначено золото, синім діелектричний поліамід з діелектричною проникністю $\epsilon = 4$, магнітною $\mu = 1,25 \cdot 10^{-6}$. Для золота $\epsilon = 7$; $\mu = -36 \cdot 10^{-6}$. Відносна провідність $\rho = 0,4 \cdot 10^8$ См/м. Деталі експерименту за посиланням [96].

Для тестування будуть приведені дослідження тільки метаструктури представленої на рис. 3.4. Це тестування проводилось за допомогою нейронної мережі, яка була навчена на структурах, які досліджувались в однакових діапазонах. Так структура складається з двох металічних пластин, між якими знаходиться ізоляційна прокладка. Ця структура досліджувалась як лінійний поляризатор, для різних геометричних розмірів структури (відстані між металічними пластинами, відстані між вирізами, шириною та довжиною вирізів). Коефіцієнт пропускання ізолюючої проклади $n = 2$. Матеріал підкладки SiN . Магнітна проникність була вибрана, як $\mu = 1$, тому що не було встановлено точну інформацію про цей параметр для цього матеріалу, але попри це діелектрик, то зазвичай це число в межах до 1.

Отримано, що $\epsilon = 4$, відповідно до залежності показника заломлення від відносно діелектричної проникності. Як метал було використано срібло, де діелектрична проникність моделювалась за допомогою моделі Друде. Для навчання і тестування було взяте середнє значення діелектричної проникності срібла на частотах дослідження метаструктури – $\epsilon = -75$. Магнітна проникність $\mu = -26 \cdot 10^{-6}$. Відносна провідність срібла $\rho = 0,6 \cdot 10^8$ См/м. Тестування відбувалось для цієї метаструктури з різними h (товщина діелектричного шару). Ця структура може бути використана як кругові поляризатори, поворотних пластин (45° - 90°). Інша інформація про структуру за посиланням - [94]. Досліджуваний діапазон цього матеріалу важливий тим, що за допомогою нього є можливість досліджувати біологічні структури (сенсори), створювати альтернативні джерела генерації енергії на основі поглинання сонячного світла (фотовольтаїчні елементи), плазмонних хвилеводів та ін. [3].

Потрібно зазначити, що всі значення встановлювались із врахуванням інформації, яка представлена у вказаних джерелах та при відсутності потрібної інформації про компоненти встановлювалась в залежності від частотних діапазонів експериментів. Для навчання нейронної мережі були використані тільки цілі частини значень, тобто без 10^{-6} . Для кремнію відносна питома провідність встановлена, як 0, тому що всюди кремній був використаний як підкладка, і має високий опір.

Опис та детальний аналіз експериментів із залученням даних метаструктур за відповідними посиланням – [92] [93] [95] [96].

На рис. 3.7-3.10 показані знімки з програмного середовища 3D-моделей метаструктур. Розміри всіх зразків відповідні до реальних експериментальних зразків вказаних на рис. 3.2-3.6.

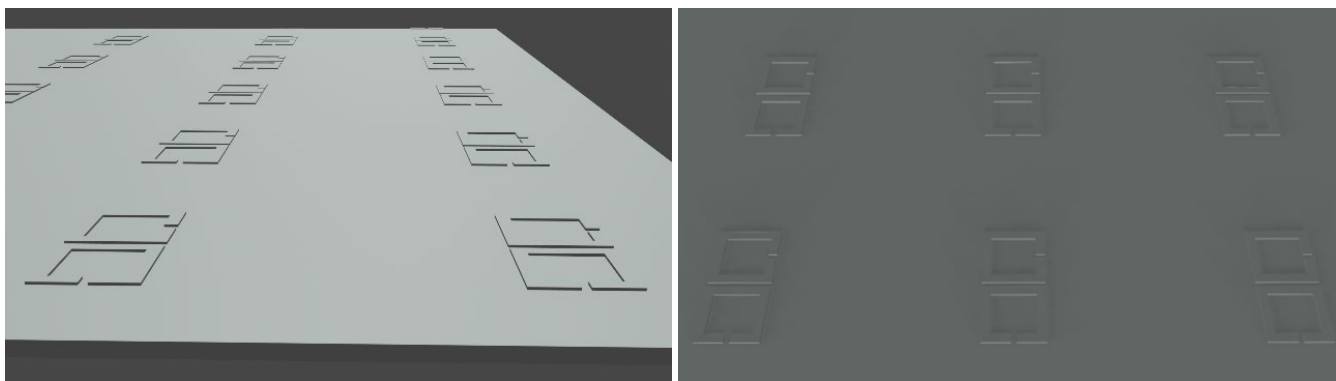


Рисунок 3.7 – Знімки 3D структур, розміри представлені на рис. 3.2 (а).

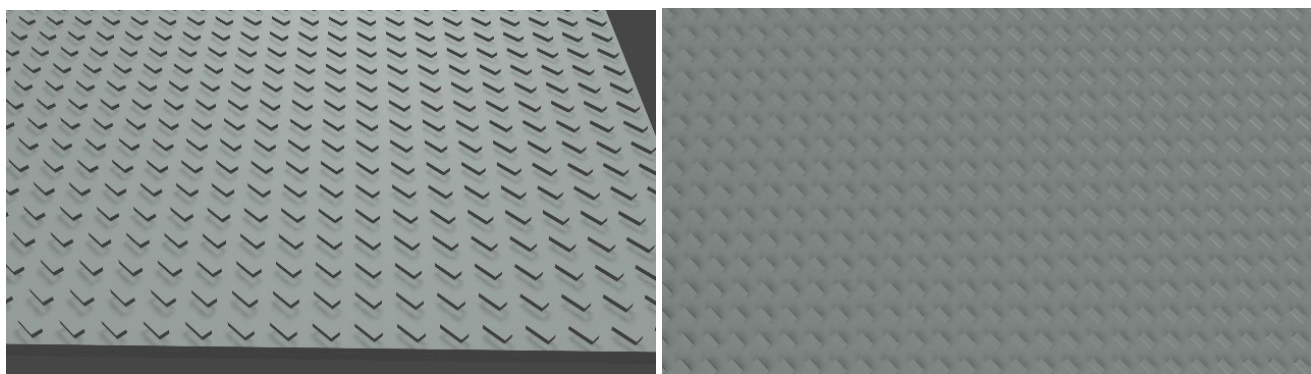


Рисунок 3.8 – Знімки 3D структур, розміри представлені на рис. 3.3.

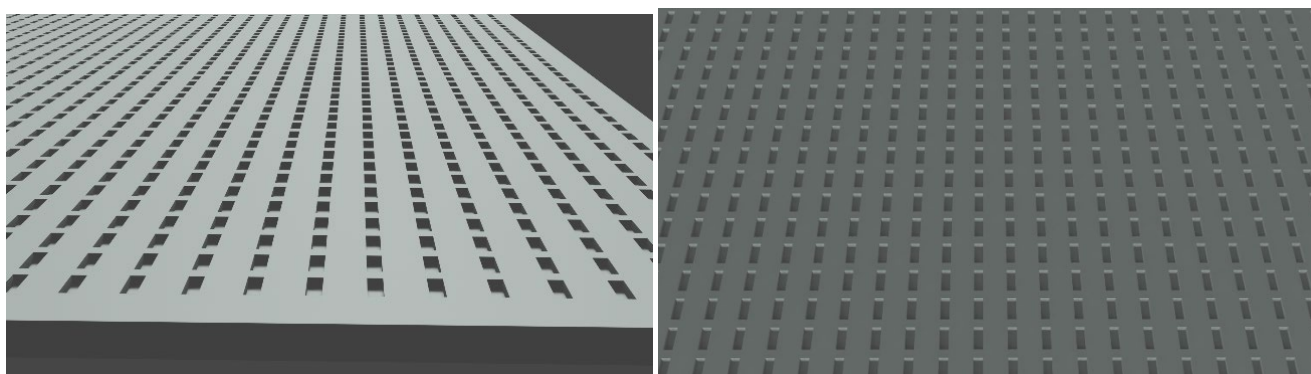


Рисунок 3.9 – Знімки 3D структур, розміри представлені на рис. 3.4.

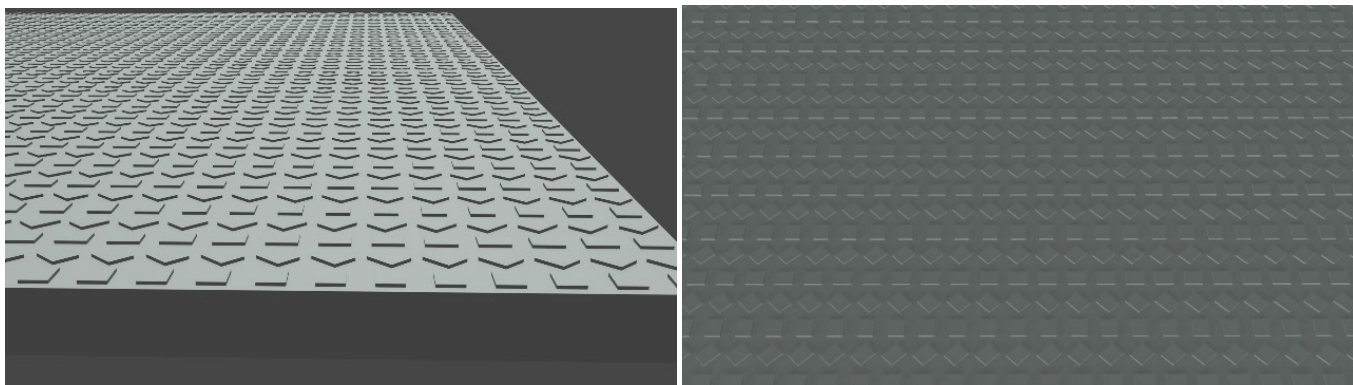


Рисунок 3.10 – Знімки 3D структур, розміри представлені на рис. 3.5.

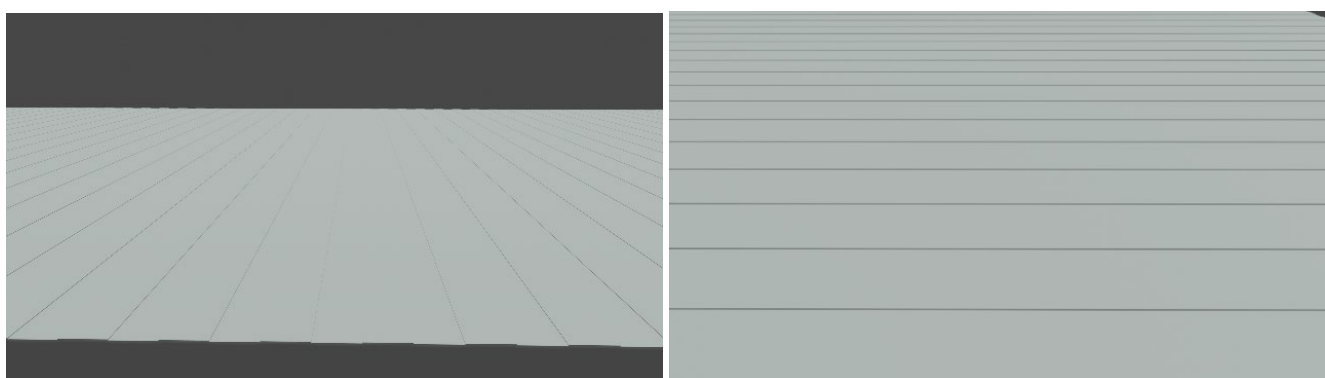


Рисунок 3.10 – Знімки 3D структур, розміри представлені на рис. 3.6.

На рис. 3.10 показані тільки верхні частини 3D структури, тому що суттєву частину структуру складно показати внаслідок великої відмінності в розмірах між антенами та підкладкою. Всі знімки були зроблені з виглядом зверху та під нахилом.

2. *Цифрове закріплення властивостей матеріалів за координатами пікселів.*

Ця задача вирішувалась таким чином: закріплення за кожним пікселем *RGB* даних – це означає, що за кожним цифровим пікселем закріплюється трійка значень у діапазоні $[0-255]$ кожна. Тобто кожній трійці значень зіставлявся матеріал, наприклад – золото – $[0, 156, 255]$, полімер – $[12, 35, 0]$ і т. д. (рис. 3.11).

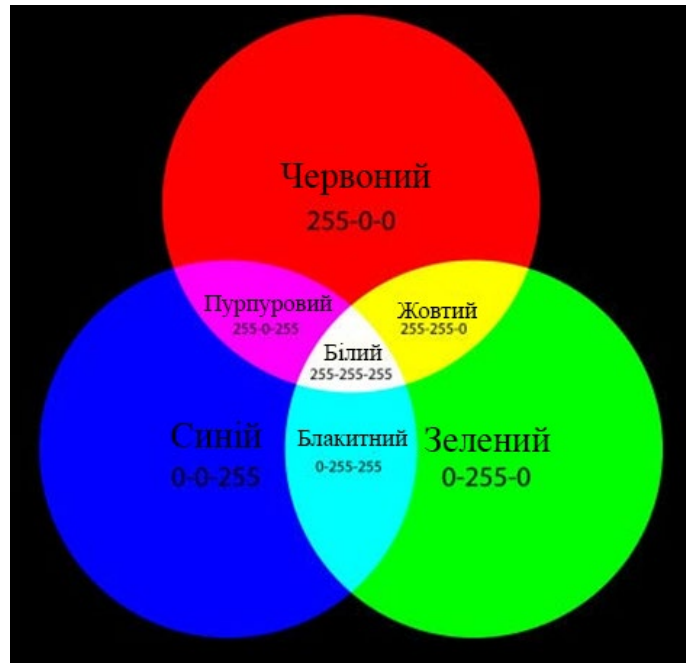


Рисунок 3.11 – Схематичне представлення трійки значень *RGB*.

Причиною вибору трійки значень *RGB* є те, що цей процес є дуже спрощеним за допомогою програмного середовища *Blender* (цифрове фарбування).

Наступна процедура полягала у правильному цифровому представленні цих моделей, тому що вони зберігаються у форматі, рідному для програмного середовища *Blender*.

Таких форматів є деяка кількість, а саме: *.dae*, *.abc*, *.obj*, *.ply*, *.stl*, *.fbx*, *.glb/.gtlf*, *.x3d* та інші. Всі вони мають свої особливі представлення 3D-моделей, сцен, цифрової інформації, яку можна створювати в цьому програмному середовищі.

Вибір правильного формату залежить від таких чинників – можливість роботи з такими даними та можливість конвертації у потрібний, для нейронної мережі, формат даних.

Найкращим форматом для представлення таких даних був шестивимірний тензор, де кожен елемент представлений у вигляді вектора (3.1).

$$X = [x, y, z, R, G, B], \quad (3.1)$$

де x, y, z – координати по відповідних осям, R, G, B – значення інтенсивності червоного, зеленого та синього відтінків відповідно.

Існує проблема в тому, що такий тензор напряму не може бути отриманий за допомогою вбудованих форматів програмного середовища. Тому був використаний програмний підхід. Зберігаючи всі моделі у форматі «.ply», використовуючи бібліотеку *Open3D (Python)*, потрібно отримати дані у форматі «.xyzrgb».

«.xyzrgb» - це формат, де є шестивимірний тензор у вигляді масиву даних, де кожен стовпець несе інформацію про просторове знаходження та властивість матеріалу, а кожен рядок – це елемент метаструктури.

Через те, що *RGB* представлення матеріалів не несе абсолютно ніякої інформації про реальні властивості, потрібно конвертувати значення *RGB* у властивості матеріалу.

Оскільки в цій роботі прогнозувались електромагнітні характеристики метаструктур, а саме залежність коефіцієнта пропускання від частоти випромінювання, то і відповідні властивості матеріалів, з яких складаються метаструктури, були вибрані для конвертації. Тобто, *RGB* проєкція була конвертована в $\epsilon\chi\sigma$, і кожен елемент виглядає таким чином:

$$X = [x, y, z, \epsilon, \chi, \sigma], \quad (3.2)$$

де ϵ - діелектрична проникність, χ - магнітна проникність, σ – питома провідність.

Властивості компонентів були встановлені у відповідність до експериментальних досліджень – рис. 3.2-3.6.

3. Представлення даних у потрібному, для нейронної мережі, вигляді.

В цій роботі використовується згорткова нейромережа. Основна задача – це правильно представити дані для вхідного шару нейронної мережі. Такі умови накладаються на вигляд даних:

- Кожен рядок масиву має бути унікальним (не може бути два рядки з однаковими координатами).

- При будь-якій конвертації даних має бути збережена залежність між фізичними властивостями матеріалів.

- Масштабування параметрів має бути коректно приведеним.

При виконанні всіх вищевказаних умов, згортова 3D нейромережа буде працювати коректно, і при незадовільнених результатах потрібно звернути увагу саме на правильність відповідності «вхідні дані – вихідні дані» або на недоліках підготовки даних (всі конвертації й т. і.).

Причинність у накладанні саме таких умов на вхідні дані полягає в наступному: якщо два однакових рядки будуть присутні, а отже, «важливість» цього рядка буде більшою порівняно з іншими, що означатиме, що він буде враховуватись функціонально та фізично неправильно; третя умова впливає із суттєвим збільшенням енергії та часу у відповідність зі збільшенням довжини мантиси параметрів. Звичайно, зменшувати масштабування потрібно тільки до тієї межі, коли варіабельність у чисел достатня, щоб вплинути на результат прогнозування, тому були проведені попередні тестування при різних масштабуваннях.

Звичайно, кількість параметрів, які можливо включити у властивості матеріалів, не обмежується трьома. Зрозуміло, що чим більше параметрів, тим більша масивність обчислень. З іншого боку, потрібно розуміти, що тільки ті параметри, які описують фізичну природу експерименту, варто включити в параметри матеріалів. Зрозуміло, що в золота, полімері тощо, є такі фізичні параметри, як жорсткість, хімічний потенціал, теплота плавлення та інші. Фізично вони впливають на всі виміри з цими матеріалами. Було б правильніше їх включити у вхідні дані тільки тоді, коли відбувається прогнозування властивостей, відповідних даним параметрам. Наприклад, якщо потрібно прогнозувати частоти акустичного пробою того чи іншого матеріалу, то до вхідних параметрів потрібно було б включити коефіцієнт жорсткості або модуль пружності. Це пов'язано з тим,

що всі параметри, які визначались як еталонні для кожного матеріалу, були обчислені з відповідних теоретичних моделей, для яких і були визначені ці параметри. Звичайно, якщо ввести залежність параметрів, які невідповідні до своїх властивостей, наприклад – хімічний потенціал – залежність сили струму від напруги, то при достатньому розмірі вибірки, результат прогнозування може бути важливим. Результат буде не найкращим, тому що існують інші параметри, які впливають з не меншим вкладом, ніж хімічний потенціал. Тобто, при збільшенні кількості параметрів, які впливають на вихідні властивості матеріалу, буде збільшуватися якість прогнозування, не збільшуючи при цьому кількість елементів вибірки.

3.3 Відтворення характеристик для використання у нейронній мережі.

Представлення експериментальних характеристик не є таким складним процесом, як представлення структури та складу матеріалів у цифровому форматі. Представляти експериментальні дані можливо різним чином. До класичних способів наочного представлення можна віднести таблиці, ряди, розподіли, графіки, гістограми, схеми, графи, малюнки. Дослідники в області глибокого навчання як правило знають, в якому вигляді потрібно накопичувати інформацію про експериментальні дослідження, якщо вони неопосередковано беруть участь в експериментальних дослідницьких групах. Якщо приходить займатись збором інформації з джерел, які містять результати реальних фізичних вимірювань, то потрібно в залежності від задачі приймати ті чи інші рішення з приводу приведення інформації до потрібного вигляду.

Інший фактор являє собою розмір вибірки експериментальної інформації. На дані можуть накладатися обмеження - кількість необхідної пам'яті для обробки даних. Це пов'язано з тим, що при роботі нейронної мережі, достатню кількість часу займає процес прорахунку помилки. В цьому процесі всі вихідні дані порівнюються з тими даними, які нейронна мережа обчислила. Тобто в залежності від того, в

якому вигляді представлені дані, буде залежати чи потрібно використовувати додаткові процедури згорткової мережі для скорочення та, одночасно, зберігання інформації.

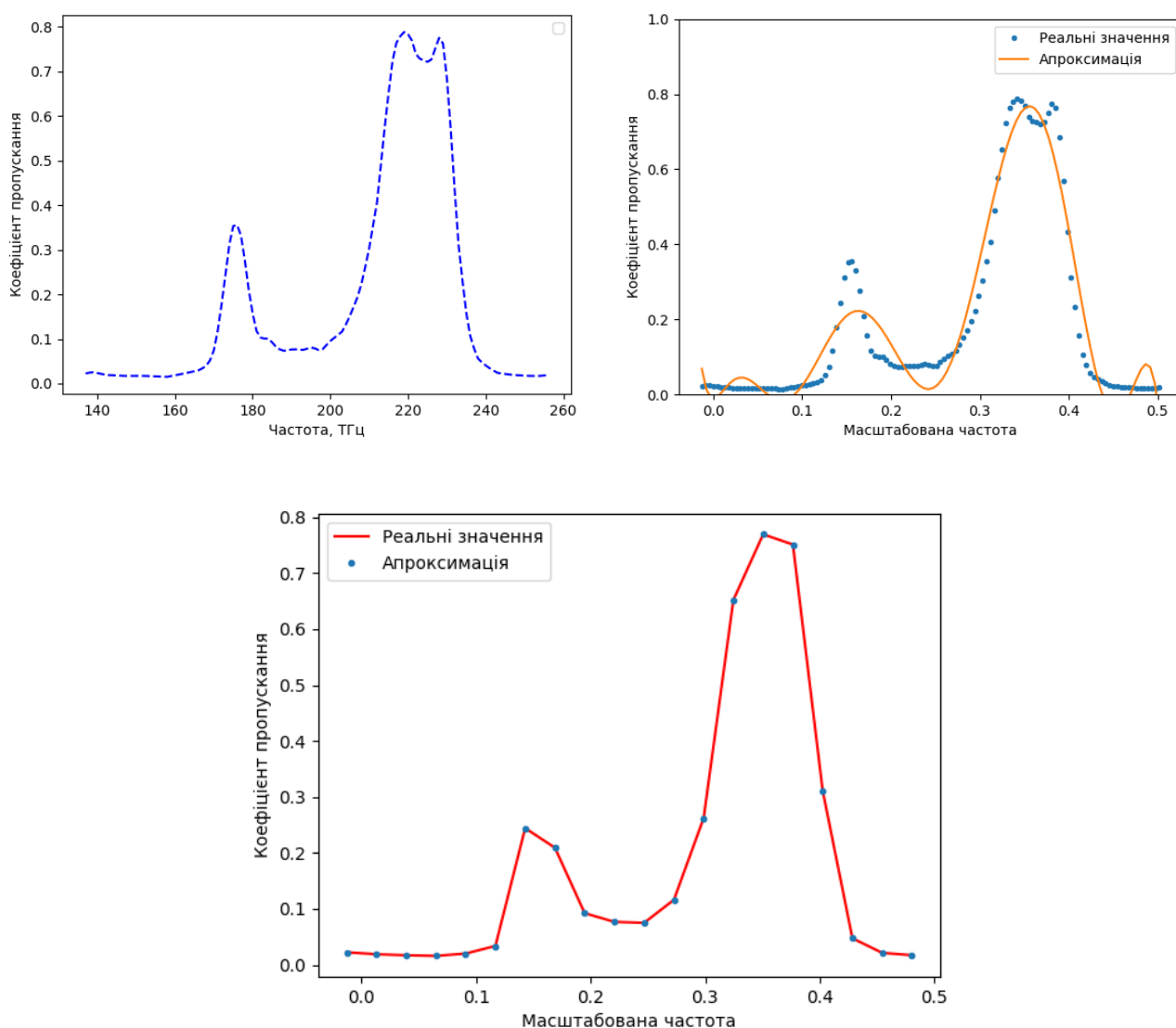
Так, наприклад, в цій роботі використані експериментальні дослідження, які були представлені у вигляді графіків залежності коефіцієнта пропускання від частоти випромінювання. Для представлення інформації про графік залежності, можна використати два методи: представити графіки у вигляді табличних точок на графіку; представити графіки у вигляді набору коефіцієнтів полінома у відповідному степені. Для того, щоб зберегти інформативність про всі зміни в графіках, вистачить двадцяти точок, які будуть представляти графіки. В результаті експерименту було визначено, що десять коефіцієнтів – це оптимальна кількість коефіцієнтів полінома, які можуть ефективно представляти всі потрібні зміни в графіках. На рисунку 3.4 приведено графіки експериментальних досліджень, які були представлені у вигляді масиву з двадцяти точок та коефіцієнтів полінома. Можна зробити висновок, що кількість вихідних даних вдвічі менша. Це означає, що вихідних нейронів, при представленні експериментальних характеристик у вигляді коефіцієнтів полінома буде десять – порівняно з представленням у вигляді точок. Як буде показано в наступних розділах, за результатами навчання нейронної мережі, варіант з представлення графіків у вигляді коефіцієнтів полінома не є ефективним для передбачення, але показним, як представлення підходу.

Числовий експеримент проводився у двох варіантах – використання всіх даних, дослідження яких проводились в різних частотних діапазонах, або тестування окремих даних, для різних частотних діапазонів.

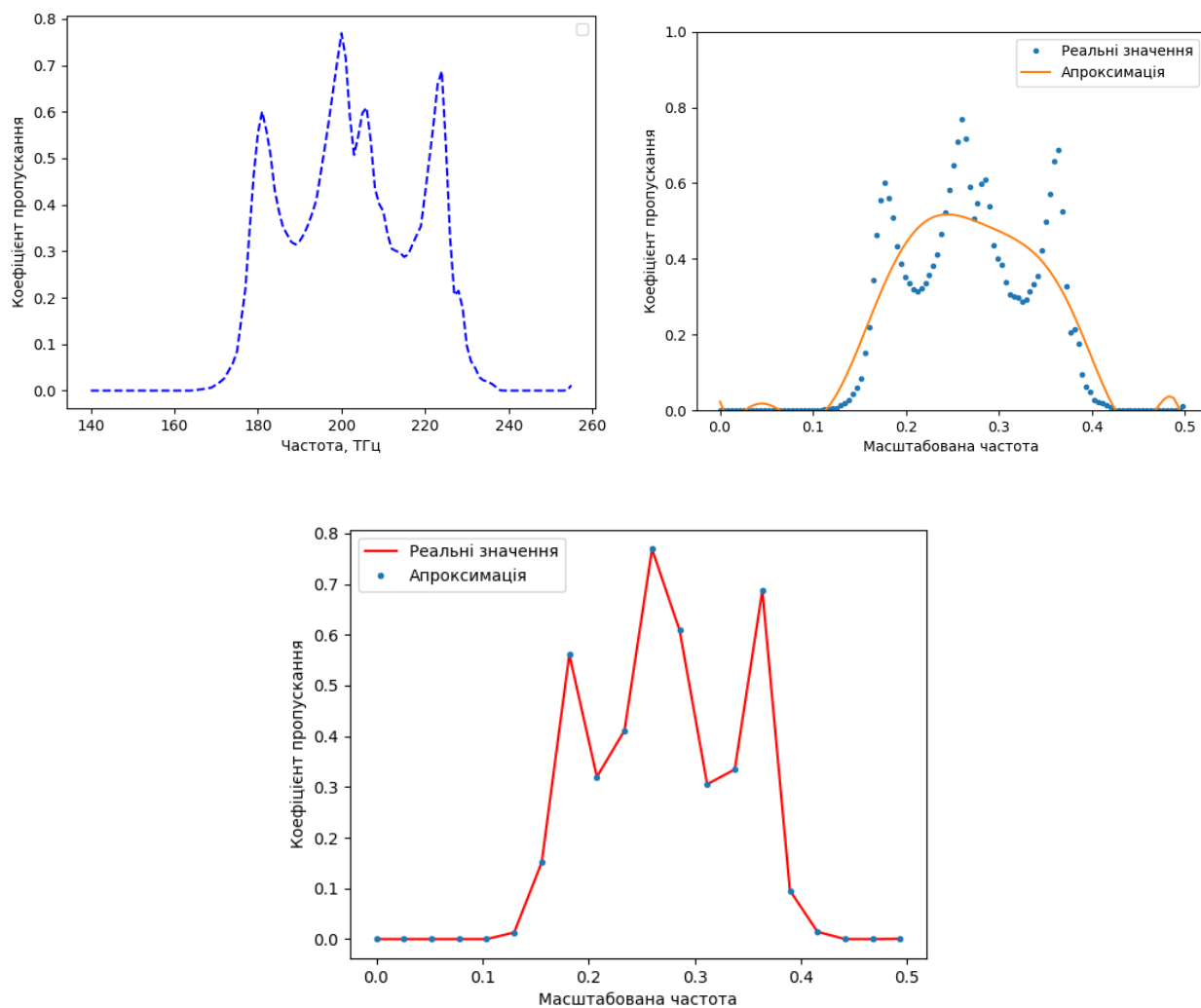
На рис. 3.12 показані графіки експериментальних досліджень структур, представлені на рис. 3.4 при $h = 100$ нм, $h = 180$ нм. Вони були використані для апробації роботи нейронної мережі.

Для представлення цих графіків було вибрано два методи – як координатні значення графіку та коефіцієнти полінома, отриманий лінійною апроксимацією

методом найменших квадратів. При представленні графіків у вигляді координатних значень, було вибрано двадцять точок. Більша кількість точок означає збільшення кількості вихідних нейронів в нейронній мережі, а менша означає менша інформативність про графік. При представленні графіків у вигляді коефіцієнтів полінома було вибрано поліном десятого порядку, таким чином інформативність зберігається і час обчислень буде оптимальним.



(a)



(б)

Рисунок 3.12 – Графіки експериментальних досліджень, які були використані для тестування нейронної мережі та їхні апроксимації. (а) Для структури, яка представлена на рис. 3.4 при $h = 100$ нм. (б) Для структури, яка представлена на рис. 3.4 при $h = 180$ нм.

Звичайно, для різних даних ці параметри є можливість корегувати. Значення у двадцять точок та поліном десятого степеня були встановлені числовим експериментом.

Проведено масштабування даних до чисел, які належать відрізьку $[0, 1]$:

$$\text{max min scale} = \frac{(x - \min(x))}{(\max(x) - \min(x))}, \quad (3.3)$$

де x – значення величини, max min scale – нове значення величини, $\min(x)$ – мінімальне значення величини x , $\max(x)$ – максимальне значення величини.

Числовий вигляд вхідних та вихідних даних, як правило, зводиться до цих масштабів, щоб математичні операції в нейронній мережі проводились швидше.

3.4 Особливості вибору моделей цифрових даних

Важливість у виборі правильних моделей цифрових даних полягає в тому, що, інформація, яка міститься в цих моделях, має коректно відображати поставлену задачу. Тобто, фізична причинність між вхідними та вихідними даними має бути відомою. При встановленні цієї причинності (на основі фізичних теорій, експериментальних даних, емпіричних спостереженнях і т. д.), потрібно встановити, яким чином відобразити таку інформацію про причинність.

В залежності від того, чи ця причинність складна по природі (якщо велика кількість параметрів, які залучені у формуванні причинності), можна ставити питання про те, яка має бути вибірка даних та величина цієї вибірки. Також, від цієї складності залежить кількість вхідних параметрів (нейронів) та кількість шарів і кількість нейронів в цих шарах. Балансуючи дві величини – кількість вхідних нейронів (та шарів з їх нейронами) та величину вибірки, є можливість, в залежності від існування тих чи інших даних, зменшувати величину одних параметрів та збільшувати величину інших. Наприклад, візьмемо задачу з прогнозування рукописних літер MNIST (Modified National Institute of Standards and Technology) [91]. Якщо взяти деяку частину елементів із всієї бази, і навчити нейронну мережу. Нехай точність прогнозування буде A %. Якщо тепер взяти більше елементів, але використовувати для навчання тільки частину зображення (обрізані якимось чином),

то точність прогнозування може буде близькою до A при деякій кількості елементів взятих з бази (рис. 3.6).



Рисунок 3.6 – Процес створення трьох окремих вибірок даних, які несуть різну інформацію про ті самі елементи.

Якщо у дослідників немає можливості збирати велику кількість даних (висока складність в збиранні тощо), але в той самий час існує можливість підготувати високу інформативність про елемент вибірки (достатню кількість параметрів, які впливають на причинність) тоді якість навчання нейронної мережі може бути достатньою для поставлених задач. Складність цього питання полягає в тому, що поки не існує якісної (і взагалі визначеної) теорії з приводу зв'язку кількості нейронів та шарів в залежності від величини вибірки. Надалі передбачається з'ясувати причину таких залежностей.

Як зазначалось у вступі, за останній час область дослідження метаматеріалів отримала великий виплиск нових робіт – дизайн метаструктур за допомогою глибокого навчання та експериментальні дослідження метаструктур. Частіше всього в таких роботах існують всі параметри метаструктур – топологічні розміри, склад, обставини експерименту та виміряні характеристики. На роботи, які відбирались для цього дослідження, накладались умови, які були вказані раніше – однорідність

даних: однаковість експериментальних характеристик, всі параметри метаструктур та обставини, при яких були виміряні характеристики.

Як буде показано в наступному розділі, кількість вхідних нейронів – це по суті кількість параметрів елементу вибірки. Збільшуючи кількість вхідних нейронів (інформації про метаструктуру та експеримент), можна навіть при малій вибірці отримати якісний результат навчання нейронної мережі.

Висновки до розділу 3

На початку розділу було представлено основні методи створення цифрових 3D-моделей. Описано, що такими методами є векторне та полігональне моделювання. В цій роботі був використаний полігональний метод створення 3D-моделей, тому що він простий у використанні, а також інформація, яка створена цим методом, є зручною. Його точності вистачає, щоб зібрати всю інформацію про структуру метаматеріалів.

Також були описано більшість форматів зберігання цифрової інформації про 3D-об'єкти. Ці формати мають можливість зберігати додаткову інформацію про 3D-об'єкти – текстуру, колір, взаємне розташування і т. д. Існує можливість використовувати додаткові дані, щоб зберігати інформацію про властивості реальних 3D-об'єктів. В цій роботі, саме цю можливість було використано для представлення складу метаматеріалів, як додаткова інформація про пікселі. Всі дані, були збережені у форматі «.ply», який має перевагу в тому, що несе інформацію про координати всіх пікселів та колір (*RGB*), який закріплений до кожного пікселю. Дані збережені в цьому форматі, не підходять, як вхідні дані для нейронної мережі. Тому був використаний програмний пакет *Open3D* на базі *Python*, для конвертації початкового формату у формат «.xyzrgb», який становить собою масив, де кожен рядок це вектор з шести чисел – перші три це координати, три інших – це *RGB*

числа. В такому вигляді інформацію можна використовувати в процесах згортки та надалі для навчання нейронної мережі.

Приведено всі розміри, які були використані при побудові 3D об'єктів, а також всі зіставлення між кодами HEX та параметрами елементів 3D об'єктів. Заміна чисел *RGB* у властивості матеріалів були проведені за допомогою розробленої програми за допомогою мови програмування *Python*.

В цій роботі були використані експериментальні характеристики коефіцієнта пропускання в залежності від частоти випромінювання в різних діапазонах. Вони були представлені у вигляді графіків залежності. Для представлення даних, відповідних для нейронної мережі, графіки можуть бути представлені різними методами. Найбільш очевидним є метод представлення точок на графіку, де в такій ситуації ручним методом було визначено, що двадцять точок на графіку зможуть представляти всі залежності, які присутні на графіках (як правило, в точках, де похідна близька до нуля). Також, графіки можна представляти у вигляді коефіцієнтів полінома (є можливість апроксимувати будь-який графік з деякою точністю за допомогою поліному). Методом підбору було визначено, що поліном в десятій степені ефективно апроксимує всі графіки, які були взяті з експериментальних досліджень метаматеріалів.

РОЗДІЛ 4

ПРОГНОЗУВАННЯ ВЛАСТИВОСТЕЙ МЕТАМАТЕРІАЛІВ НА ОСНОВІ ЗГОРТКОВОЇ НЕЙРОННОЇ МЕРЕЖІ

В цьому розділі представлено результати прогнозування властивостей метаматеріалів на основі інформації про їх структуру та склад. Для прогнозування були використані структури представлені на рис. 3.3 та на рис 3.2.

4.1 Варіативність у виборі параметрів системи та аргументація у їх виборі

Необхідна кількість прихованих шарів залежить від внутрішньої складності набору даних, аналізуючи вихідний результат кожного шару.

Мережі без прихованих шарів можуть якісно моделювати лише лінійну функцію. Цього недостатньо для більшості завдань розпізнавання ознак різного виду зображення, об'єктів і т. д.

Один прихований рівень дозволяє мережі моделювати будь-яку складну функцію. Цього достатньо для багатьох завдань розпізнавання зображень, але не завжди є оптимальним для об'єктів більшої розмірності.

Теоретично два прихованих шари мало використовуються порівняно з одним шаром, однак на практиці для деяких завдань додаткові приховані шари можуть бути необхідними. Треба слідкувати, щоб додатковий шар не призводив до перенавчання нейронної мережі.

В цілому для того, щоб дізнатись який найкращий варіант, потрібно для різного числа шарів. Це може зайняти час, особливо, якщо задача складна, або ж кількість даних достатньо велика.

Якщо аналізувати кількість згорткових шарів, то чим більше згорткових шарів, тим краще (в розумних межах, оскільки кожен згортковий шар зменшує кількість вхідних функцій до повністю зв'язаних шарів). З досвіду потрібно враховувати, що приблизно після двох-трьох шарів приріст точності стає досить малим. тож потрібно вирішити, що є основною метою — загальна точність чи час навчання. Проте, усі завдання розпізнавання зображень – об'єктів різні, тому найкращим методом є просто спробувати збільшувати кількість згорткових шарів по одному, доки результат не буде прийнятним.

Кількість нейронів на даний час не є точним числом, оскільки розкид задач, який потрібно вирішувати дуже великий, і тому для кожного окремого завдання потрібно емпірично досліджувати оптимальну кількість нейронів. Приблизний вибір кількості нейронів полягає в тому, щоб використовувати кількість нейронів розміром $2/3$ розміру попереднього шару. Однак це лише приблизний параметр, який знову залежить від набору даних. Інший часто використовуваний варіант — почати з надмірної кількості нейронів, а потім крок за кроком вилучати зайву кількість нейронів.

В роботі [148] якісно показано основні формули та початкові значення, тобто потрібно підбирати початкові гіперпараметри (кількість шарів, кількість нейронів, швидкість навчання та ін.).

Потрібно привести тільки основні формули, які були використані в цій роботі для обчислення згорток в нейронній мережі. Вони виконувались автоматично за допомогою бібліотеки *pytorch* - [149].

Загальна формула для розрахунку вихідного значення $[N, C_{out}, D_{out}, H_{out}, W_{out}]$ при вхідних значеннях $[N, C_{in}, D_{in}, H_{in}, W_{in}]$ має такий вигляд:

$$out(N_i, C_{out_j}) = bias(C_{out_j}) + \sum_{k=0}^{C_{in}-1} weight(C_{out_j}, k) * input(N_i, k), \quad (4.1)$$

де $\star$ - це операція 3D згортки, N – величина вибірки, C – число каналів, H – висота в пікселях, W – ширина в пікселях, D – глибина в пікселях, i, j – число шару та позиція нейрона.

Для розрахунку вихідного масиву використовуються формули:

$$D_{out} = \left\lfloor \frac{D_{in} + 2 \times padding[0] - dilation[0] \times (kernel_size[0] - 1) - 1}{stride[0]} \right\rfloor, \quad (4.2)$$

$$H_{out} = \left\lfloor \frac{H_{in} + 2 \times padding[1] - dilation[1] \times (kernel_size[1] - 1) - 1}{stride[1]} \right\rfloor, \quad (4.3)$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2 \times padding[2] - dilation[2] \times (kernel_size[2] - 1) - 1}{stride[2]} \right\rfloor, \quad (4.4)$$

де $padding$ – відступ (в нашому випадку за умовчанням – 0), $dilation$ – відстань між елементами ядра, $kernel_size$ – розмір згорнутого ядра, $stride$ – величина зсуву ядра згортки, детальніше в документації pytorch для 3D згортки - [149].

4.2 Застосування техніки навчання згорткових мереж для прогнозування властивостей метаматеріалів

Враховуючи розглянуту вище техніку побудови згорткових нейронних мереж та їх застосування, є можливість використати згорткову мережу для прогнозування коефіцієнта пропускання метаматеріалів на основі їх структури та складу. Вхідні дані представляють топологічну структуру метаматеріалу та його склад, як було описано в підрозділі 3.2. Вихідними даними є експериментальна залежність коефіцієнта проходження в залежності від частоти, які представлені у двох варіантах – у вигляді масиву координат та у вигляді коефіцієнтів полінома десятого степеня – підрозділ 2.3. Далі будуть показані практичні застосування такого підходу у вигляді архітектури нейромереж (в програмному варіанті та як граф навчання) для

двох випадків представлення експериментальних характеристик та результати тестування згорткової нейронної мережі.

Як представлено на рис. 4.1, рис. 4.2, архітектура нейронної мережі, яка була використана для задачі, яка була поставлена в цій роботі, є доволі громіздкою. Причиною цього є складність патернів, які можуть виникнути у процесі дослідження спільних ознак у різних метаструктур і їх вплив на експериментальні дослідження коефіцієнта пропускання в залежності від частоти.

Layer (type)	Output Shape	Param #
InstanceNorm3d-1	[-1, 3, 256, 256, 6]	0
Conv3d-2	[-1, 6, 256, 256, 6]	33,756
Conv3d-3	[-1, 16, 256, 256, 6]	83,248
InstanceNorm3d-4	[-1, 16, 256, 256, 6]	0
Conv3d-5	[-1, 32, 25, 25, 6]	98,336
Conv3d-6	[-1, 64, 25, 25, 6]	153,664
InstanceNorm3d-7	[-1, 64, 25, 25, 6]	0
Dropout-8	[-1, 1536]	0
Linear-9	[-1, 512]	786,944
Linear-10	[-1, 128]	65,664
Linear-11	[-1, 40]	5,160
Total params: 1,226,772		
Trainable params: 1,226,772		
Non-trainable params: 0		
Input size (MB): 4.50		
Forward/backward pass size (MB): 127.59		
Params size (MB): 4.68		
Estimated Total Size (MB): 136.77		

Рисунок 4.1 - Програмний вигляд архітектури нейронної мережі у випадку представлення вихідних значень, як координатних точок.

Як, видно, з рис. 4.1, останній шар нейронної мережі має сорок нейронів. Через те, що координатних точок двадцять, то потрібно прогнозувати числові значення по осі X та значення по осі Y. Таким чином перші двадцять нейронів представляли значення частоти, а наступні значення коефіцієнта проходження для конкретної частоти.

Layer (type)	Output Shape	Param #
InstanceNorm3d-1	[-1, 3, 256, 256, 6]	0
Conv3d-2	[-1, 6, 256, 256, 6]	33,756
Conv3d-3	[-1, 16, 256, 256, 6]	83,248
InstanceNorm3d-4	[-1, 16, 256, 256, 6]	0
Conv3d-5	[-1, 32, 25, 25, 6]	98,336
Conv3d-6	[-1, 64, 25, 25, 6]	153,664
InstanceNorm3d-7	[-1, 64, 25, 25, 6]	0
Dropout-8	[-1, 1536]	0
Linear-9	[-1, 512]	786,944
Linear-10	[-1, 128]	65,664
Linear-11	[-1, 11]	1,419
Total params: 1,223,031		
Trainable params: 1,223,031		
Non-trainable params: 0		
Input size (MB): 4.50		
Forward/backward pass size (MB): 127.59		
Params size (MB): 4.67		
Estimated Total Size (MB): 136.76		

Рисунок 4.2 - Програмний вигляд архітектури нейронної мережі у випадку представлення вихідних значень, як коефіцієнтів полінома.

Як, видно з рис. 4.2, останній шар нейронної мережі має одинадцять нейронів. Через те, що потрібно прогнозувати коефіцієнти полінома десятого степеня, то додатковий член додається до десяти нейронів.

На рис 4.1, рис. 4.2 використовуються наступні терміни: *InstanceNorm* – шар нормування, який потрібен для того, щоб привести дані у зручний для нейронної мережі вигляд. *DropOut* – шар, який зменшує кількість зв'язків між нейронами, які «майже» не беруть участь (слабкий вплив) в подальшому перенесенні інформації. *Conv3d* – шар 3D згортки. *Linear* – повно зв'язний шар нейронної мережі. *Output Shape* – розмірність вихідних даних після обробки шару. *Param* – кількість параметрів (ваги, додаткові члени).

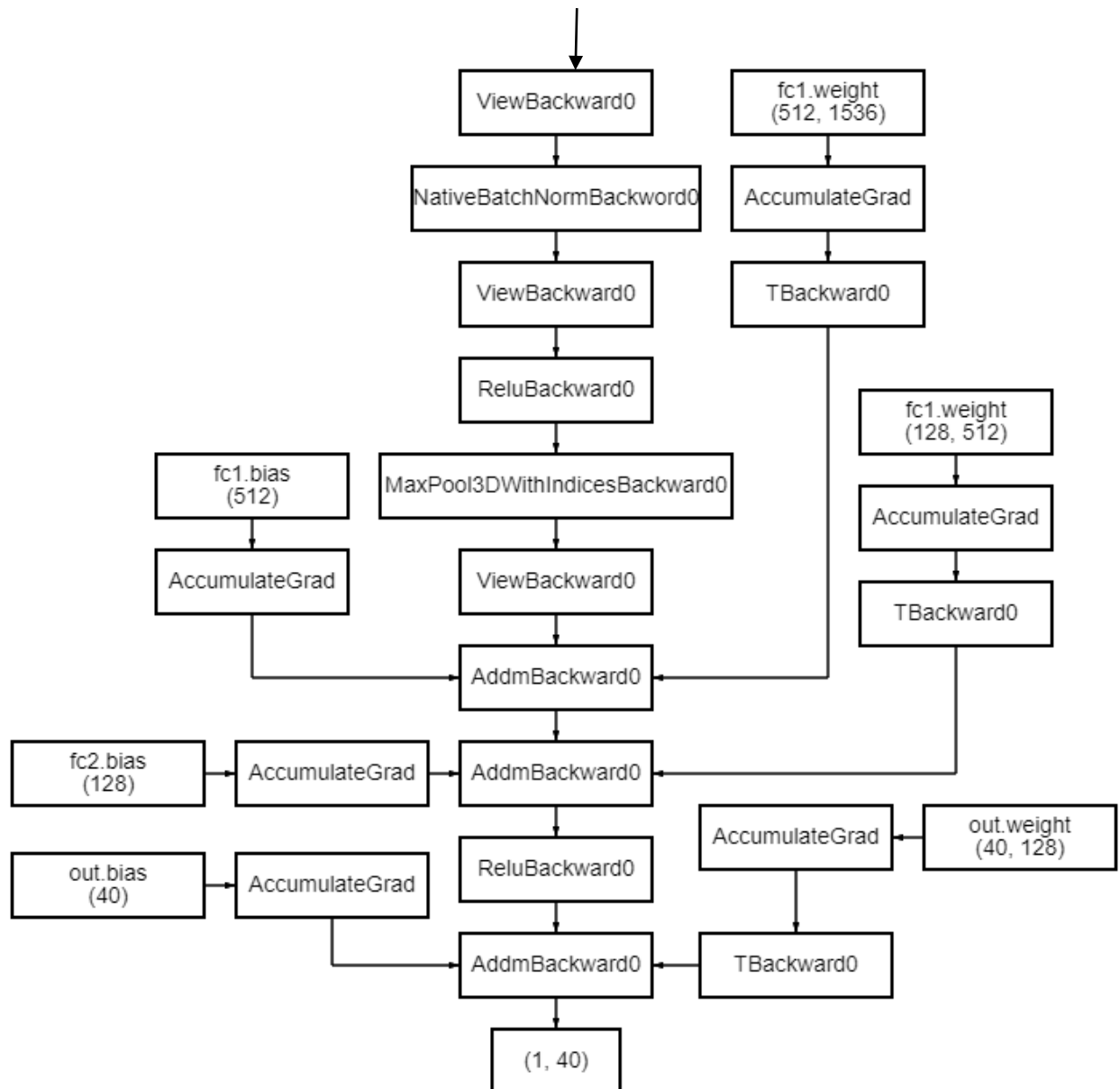
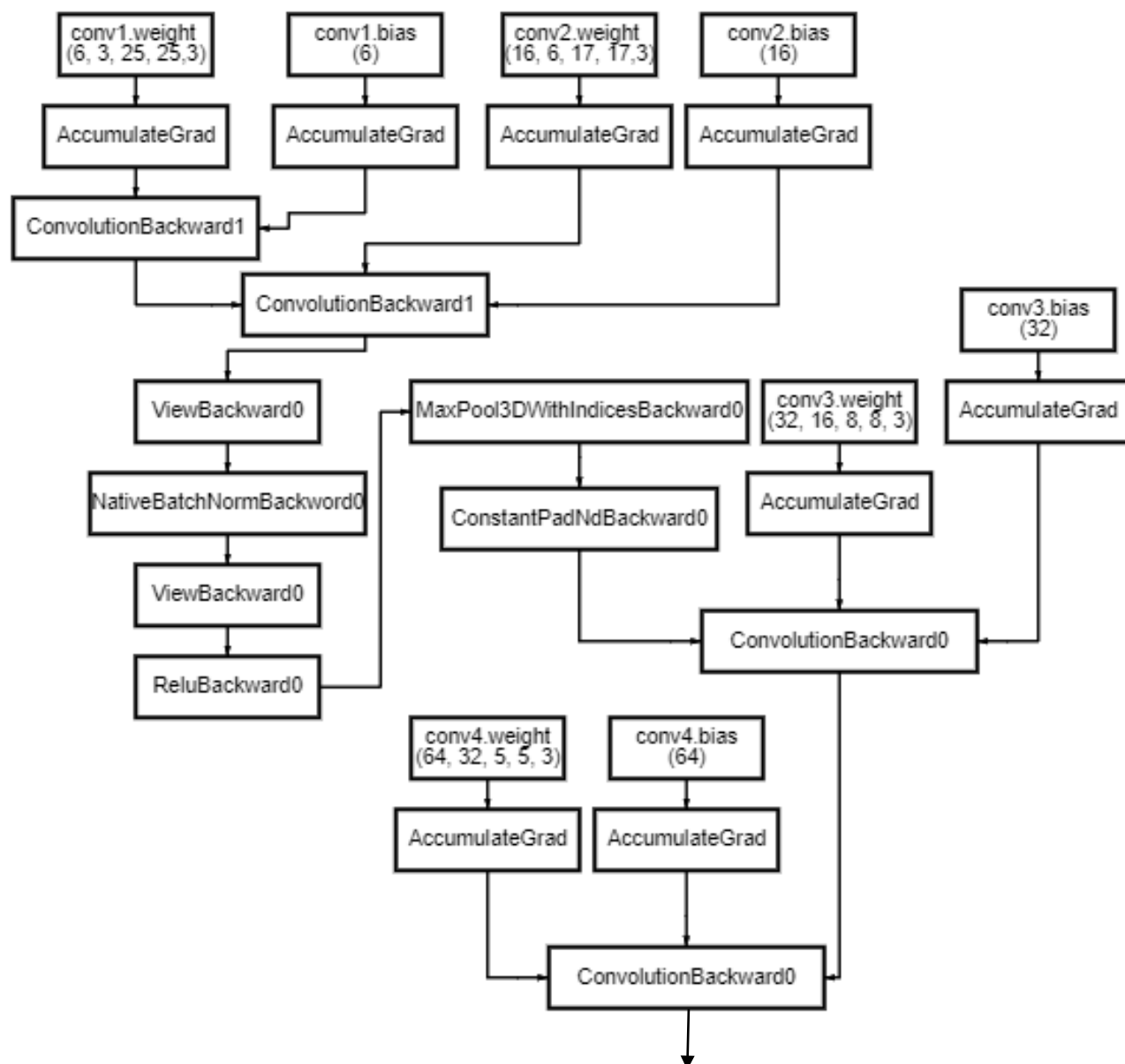


Рисунок 4.3 – Граф навчання нейронної мережі у випадку представлення графіків, як координатних точок. Вихідний шар має сорок нейронів.

На графі такі позначення: *AccumulateGrad* – процес розрахунку градієнта похибки; *Backward* – процес зміни коефіцієнтів нейронної мережі; *bias/weight* – коефіцієнти нейронної мережі.

На рисунку 4.4 представлена архітектура нейромережі у вигляді графу навчання для випадку з представленням графіків, коефіцієнтів полінома.



Продовження графу навчання на наступній сторінці. Всі блоки в даному графі є програмними процесами в алгоритмі згорткової нейронної мережі. Блоки без вхідних значень є параметрами згорткової нейронної мережі в окремих її шарах. Кількість параметрів нейронної мережі, а також кількість нейронів і шарів є гіперпараметрами нейронної мережі. Вони встановлюються або автоматично або вручну, і визначаються числовим експериментом. Ці значення корелюються в залежності від якості прогнозування нейронної мережі.

у зв'язку зі складністю задачі, та неявного алгоритму підбору параметрів, процес знаходження оптимальних гіперпараметрів був зупинений на числах показаних на схемах навчання та архітектури мережі.

4.3 Тестування нейронної мережі

Успішність навчання нейронної мережі майже завжди ґрунтується на величині похибки, яку має нейронна мережа. Не для всіх задач це є вірним твердженням. Задача, яка вирішувалась в даній роботі, є саме таким прикладом.

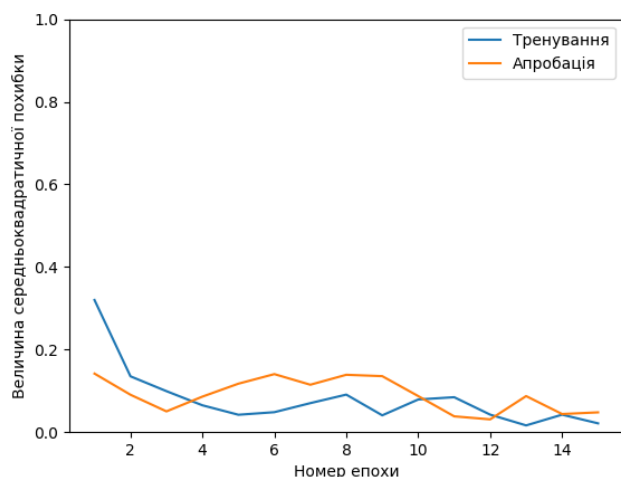
Апробація приведених даних відбувалась за правилом 80/20, що означає, що 20 % від початкової вибірки використовувалось для тестування і 80% даних використовувалось для навчання.

В другому випадку, коли експериментальні дані були представлені у вигляді коефіцієнтів полінома, після навчання нейронної мережі згенеровані графіки залежностей величини похибки від кількості епох – рис. 4.5.

Як видно з графіків, структура дійсно працює та навчається і спостерігається зменшення похибки із кожною наступною епохою навчання. А отже, з кожною наступною епохою, прогнозування вихідного графіку все більше буде якіснішим та правдоподібним.

На рис. 4.6 представлені графіки залежності середньоквадратичної похибки в залежності від кількості епох навчання при прогнозуванні координатних точок. Видно, як після восьмої епохи графік апробації на рис. 4.6 (а) зріс. Це процес перенавчання, що означає, що нейронна мережа не зберігає тенденції та властивості даних, а просто записує окремі залежності. Після цього, якщо потрібно передбачити властивості даних, які не були записані в нейронну мережу, то результат буде негативним.

(a)



(б)

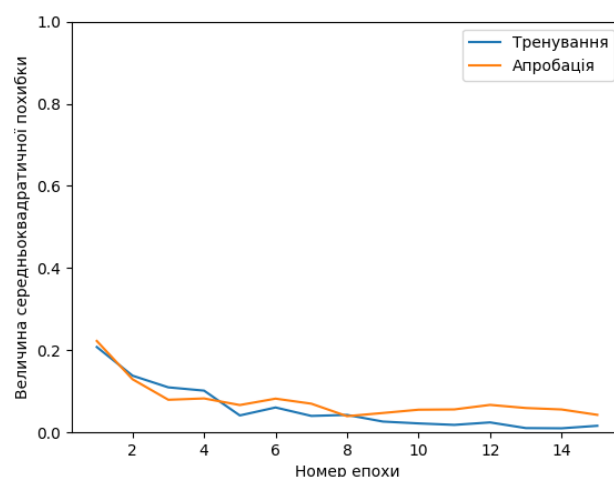
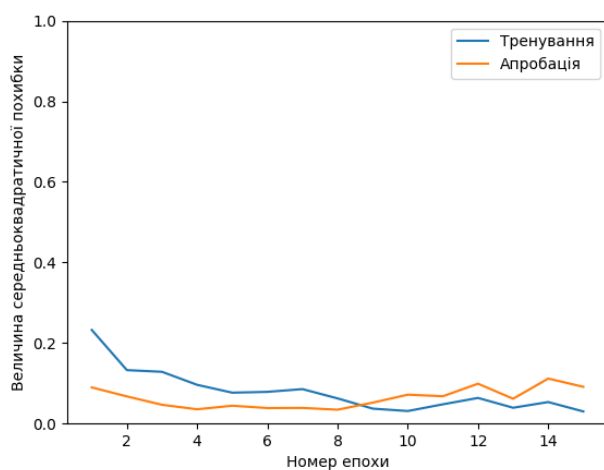


Рисунок 4.5 – Графіки залежності середньоквадратичної похибки в залежності від кількості епох навчання при прогнозуванні коефіцієнтів полінома. (а) Належить до рис. 3.12 (а). (б) Належить до рис. 3.12 (б).

На рис. 4.7 показані значення прогнозування коефіцієнтів полінома для графіків, які представлені на рис. 3.12.

(а)



(б)

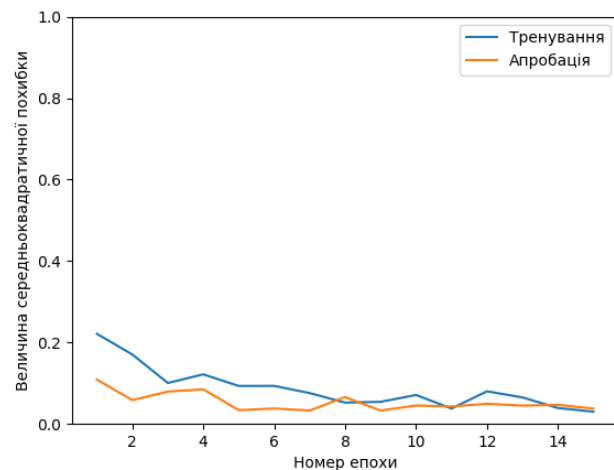


Рисунок 4.6 – Графіки залежності середньоквадратичної похибки в залежності від кількості епох навчання при прогнозуванні координатних точок. (а) Належить до рис. 3.12 (а). (б) Належить до рис. 3.12 (б).

На рис. 4.6 видно, як залежність похибки від кількості епох відрізняються для різних структур. Це вплив малої кількості даних для навчання, що означає, що деякі графіки прогнозувати простіше, ніж інші. Тобто, деякі структури мають більш унікальну структуру та відповідно електромагнітні властивості, що складно прогнозувати при такому об'ємі даних. Основна мета отримати зменшення похибки та для тренування і для апробації.

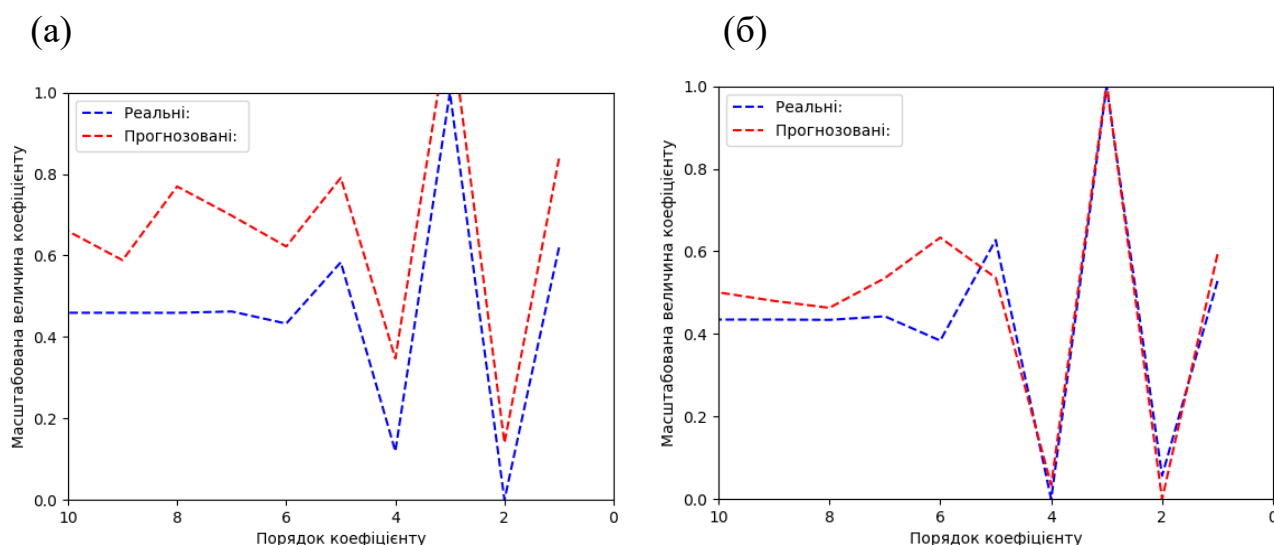


Рисунок 4.7 – Графіки залежності масштабованих коефіцієнтів полінома в залежності від порядку коефіцієнту. (а) Належить до рис. 3.12 (а). (б) Належить до рис. 3.12 (б).

Значення для графіків представлені нижче:

Для рис. 4.7 (а).

Реальні: [0.6197104, 0.0, 1.0, 0.11930583, 0.58333427, 0.43281317, 0.46254474, 0.45914727, 0.45934752, 0.45934284, 0.45934284]

Прогнозовані: [0.8380994, 0.14015469, 1.1809906, 0.34716496, 0.79055053, 0.62198794, 0.6978662, 0.7693689, 0.5883604, 0.65969193, 0.6330386]

Для рис. 4.7 (б).

Реальні: [0.52842945, 0.0565926, 1.0, 0.0, 0.6275758, 0.3838324, 0.44265613, 0.4341173, 0.4348082, 0.43478072, 0.4347811]

Прогнозовані: [0.59306294, 0, 1.007374, 0.034103062, 0.5359896, 0.6335097, 0.53516674, 0.46325043, 0.48007572, 0.5006119, 0.46437696]

Масштабована величина коефіцієнтів означає, що після того, як була проведена апроксимація, всі коефіцієнти були масштабовані за формулою (3.3).

На рис. 4.7 показані значення прогнозування координатних точок для графіків, які представлені на рис. 3.12.

Як видно з графіків, точність прогнозування низька, і не підходить для практичного прогнозування точних властивостей структур. Попри це, нейронна мережа прогнозує характер реальних залежностей коефіцієнта пропускання від частоти. Потрібно враховувати, також, що після отримання прогнозованих значень масштабованої частоти є можливість масштабувати в оригінальний діапазон значень. Для цього потрібно, щоб всі значення, які прогнозувались, а також, які були використані для навчання були масштабовані із самого початку в однаковому діапазоні. Також важливо, щоб всі значення були в однакових одиницях вимірювання.

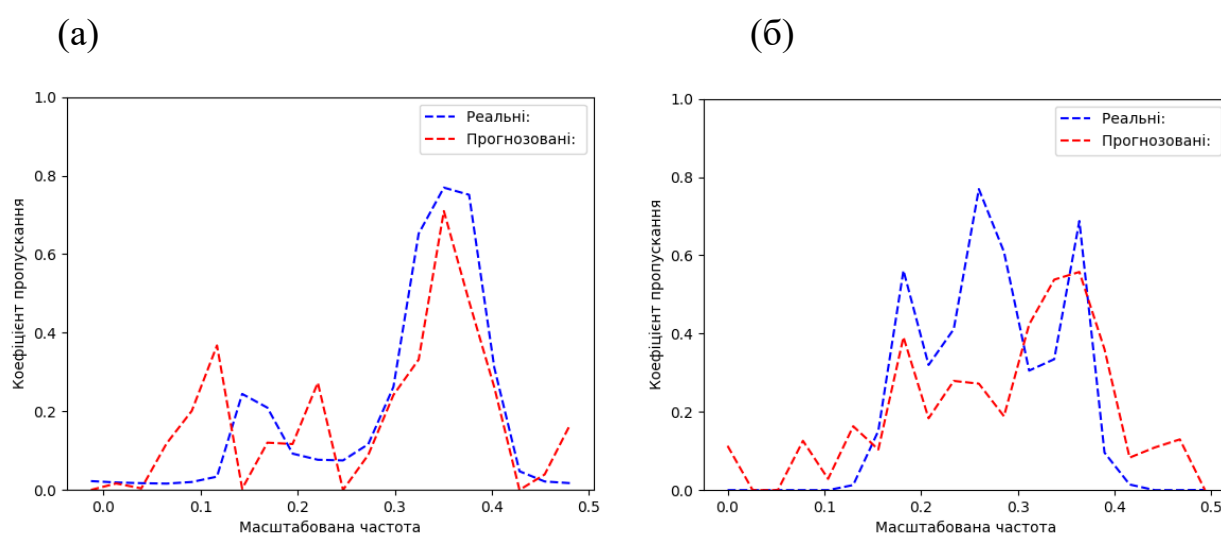


Рисунок 4.8 - Графіки залежності коефіцієнта пропускання в залежності від масштабованої частоти. (а) Належить до рис. 3.12 (а). (б) Належить до рис. 3.12 (б).

Як зазначалось, в цьому випадку прогнозувалось сорок числових значень. Перші двадцять це масштабована частота (за формулою (3.3)), а наступні двадцять, коефіцієнт пропускання. На рис 4.8 графіки по осі X були приведені до однакових значень для зручності аналізу графіків. Після рис. 4.8 приведені реальні та прогнозовані значення масштабованої частоти та коефіцієнту пропускання.

Як видно з рис. 4.8, нейронна мережа здатна прогнозувати поведінку відгуку метаструктури на зовнішнє електромагнітне випромінювання на основі інформації про структуру метаматеріалу. Враховуючи малу кількість даних даний результат є, що підтверджує ефективність та можливість розвитку підходу до збереження інформації про метаструктури для прогнозування їх властивостей.

Для рис. 4.8 (а).

Реальні: [-0.012987014, 0.012987014, 0.038961038, 0.064935066, 0.09090909, 0.116883114, 0.14285715, 0.16883117, 0.19480518, 0.22077923, 0.24675323, 0.27272728, 0.2987013, 0.32467532, 0.35064936, 0.37662336, 0.4025974, 0.42857143, 0.45454544, 0.48051947, 0.0225, 0.019, 0.0172, 0.0161, 0.0202, 0.0338, 0.24429998, 0.2093, 0.0924, 0.0768, 0.0751, 0.11630001, 0.2623, 0.6522, 0.7693, 0.75109994, 0.3114, 0.047500003, 0.0215, 0.0173]

Прогнозовані: [0.060140304, 0.3850649, 0.00027713738, 0.11589991, 0.3016426, 0.048162125, 0.14014784, 0, 0.07744114, 0.43345103, 0.20851138, 0.38179973, 0.25860208, 0.44152153, 0.19651465, 0.25396004, 0.40143672, 0.2832771, 0.61151034, 0.44971025, 0, 0.01640812, 0.0039747655, 0.11816128, 0.20108522, 0.3675576, 0.0032999516, 0.120338, 0.11690984, 0.2727404, 0, 0.08904219, 0.24159427, 0.33176902, 0.709703, 0.47976306, 0.2613104, 0, 0.039521642, 0.16513]

Для рис. 4.8 (б).

Реальні: [0.0, 0.025974028, 0.051948056, 0.077922076, 0.10389611, 0.12987013, 0.15584415, 0.18181819, 0.20779222, 0.23376623, 0.25974026, 0.2857143, 0.3116883,

0.33766234, 0.36363637, 0.38961035, 0.41558444, 0.44155845, 0.46753246, 0.49350646, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.013, 0.1515, 0.5612, 0.3199, 0.4112, 0.7692, 0.6089, 0.3054, 0.3346, 0.6876, 0.0959, 0.0144, 0.0, 0.0, 0.0006]

Прогнозовані: [0.09919541, 0.0053084195, 0.16427211, 0.049280398, 0.09444884, 0.11724385, 0.14485542, 0.31153142, 0.093267225, 0.23106304, 0.31671315, 0.33969098, 0.31029278, 0.34858927, 0.3756469, 0.37643573, 0.35723937, 0.503126, 0.46162617, 0.5404491, 0.1131073, 0, 0, 0.12604204, 0.028855713, 0.16321337, 0.10385544, 0.39093223, 0.18348555, 0.27907413, 0.27215573, 0.18887599, 0.42324144, 0.53791517, 0.5574385, 0.3601104, 0.08303199, 0.10904175, 0.12965156, 0]

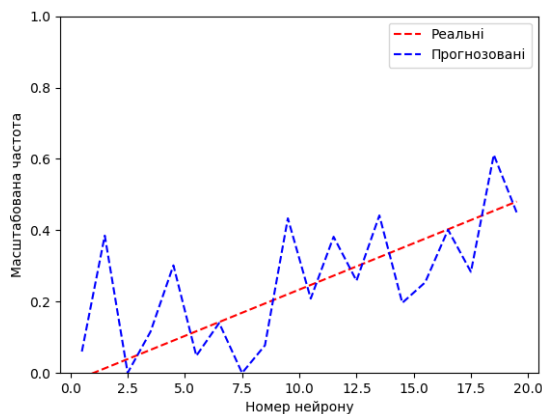


Рисунок 4.9 – Графік відхилення значень масштабованої частоти для структури рис. 3.12 (а).

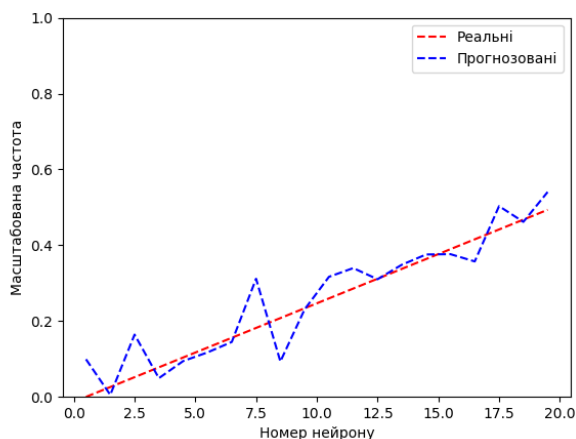


Рисунок 4.10 – Графік відхилення значень масштабованої частоти для структури рис. 3.12 (б).

Як видно з рисунка 4.10, похибка для прогнозування масштабованої частоти є суттєвою, що означає, що нейронна мережа немає інформації про структури такого типу в такому частотному діапазоні. Таким чином, для того, щоб прогнозувати властивості структур у потрібних діапазонах, нейронна мережа має спочатку бути навчена на даних, які описують структури саме в тих частотних діапазонах.

Звичайно, графіки не будуть повністю сходиться по ряду причин – мала кількість даних; недостатня інформація про реальний вплив інших обставин, при яких знімалися експериментальні дані; не ідеальна конвертація програмного 3D пакету з програмного формату (*ply, obj, fbx ...*) в *xyzrgb*; перетворення *rgb* каналів в канали з властивостями; масштабування графіків до сегмента $[0, 1]$; не ідеальність архітектури нейронної мережі, тобто значення гіперпараметрів: кількість шарів, кількість нейронів та ін.. Не зважаючи на це, маємо нейронну мережу, яку можна покращувати, та алгоритм, який дає можливість аналізувати і прогнозувати властивості та структуру метаматеріалів.

У зв'язку з тим, щоб при збільшенні похибки навіть на маленьку долю, коефіцієнти змінюються у незрівнянно більшому степені, тому даний варіант не є дуже ефективним в порівнянні з варіантом представлення графіку у вигляді точок.

Також, були проведені числові експерименти із 3D структурами іншого діапазону. Аналіз показав, що при такому малому об'єму даних нейронна мережа, не дає змогу прогнозувати властивості структур в різних діапазонах. Числовий експеримент для структур показав середньоквадратичну похибку більше 0.4, що є незадовільним результатом.

Аналізуючи отримані результати, можна підсумувати наступне – сам характер прогнозованих характеристик зберігається, що означає, що була отримана система, яка розроблена правильним шляхом; звичайно, кількість точок не дає змогу повністю отримувати бажаний результат, що пов'язано зі наростальною складністю обчислень при збільшенні точок; присутня похибка пов'язана з дуже малою

кількістю даних та складністю алгоритму підготування даних для нейромережі, що означає, що велика кількість неточностей в сукупності може дати велику похибку.

Враховуючи все вищесказане, потрібно розуміти, що для такої малої кількості даних, отриманий результат є більш ніж прийнятним. Збільшуючи кількість даних, є можливість привести результати до дійсно ефективного результату, що дасть змогу дослідникам передбачувати структуру метаматеріалів на основі встановлення потрібних властивостей. Проблеми полягають тільки в наборі даних, які відповідають умовам (описані в 1-2 розділах), а також складністю набору даних, а в деяких випадках і взагалі їх відсутністю.

Для порівняння, було взято таблицю порівняння часу виконання, переваг та недоліків різних методів обчислення розсіювання електромагнітного випромінювання для наночастинки золота [155] та підходу описаного в дисертаційному дослідженні для метаматеріалів. Об'єктивно час витрачений на обчислення розсіювання електромагнітного випромінювання для наночастинки має бути меншим ніж час витрачений на обчислення розсіювання для метаструктур різних розмірів.

Таблиця 4.1. Час обчислення, недоліки та переваги різних методів симуляції у порівнянні з підходом, який був застосований в дисертаційному дослідженні

Метод	Час обчислення	Переваги	Недоліки
Теорія Мі	Кілька мілісекунд на одну частоту.	Швидкий час обчислень. Може також використовуватися для обчислення оптичного відгуку покритих сфер.	Застосовується лише до сферично симетричних частинок. Неможливо включити взаємодію субстрату, тому важко відтворити

			багато експериментів.
Метод матриць переходу	Кілька мілісекунд на одну частоту.	Швидкий час обчислень. Підтримується широкий діапазон геометричних об'єктів. Також можна включити взаємодію із субстратом.	Обчислення є чисельно нестабільними для витягнутих або сплюснених об'єктів (матриці скорочуються під час обчислень – помилки округлення стають значними та швидко накопичуються).
Дискретне дипольне наближення	Залежить від кількості диполів і розриву. Зазвичай 50 с на окрему частоту.	Є можливість використовувати для оцінки частинки довільної форми, вказавши табличний список розташування диполів.	Критерій збіжності: $ n kd < 1$. n = комплексний показник заломлення. k = хвильовий вектор. d = відстань між диполями. (Неможливо розв'язати для високого співвідношення сторін/видовжених частинок або тих, що мають великий показник

			заломлення)
Метод скінченн их елементі в	Зазвичай 150 с на окрему частоту при використанні елемента довжиною 3 нм. Знаходиться компроміс між часом обчислення та довжиною елемента.	Може використовуватися для оцінки розподілу розсіяного поля частинки довільної форми. Використання нерегулярної тетраедральної адаптивної сітки для моделювання FEM дозволяє отримувати наближення криволінійних поверхонь більш точно.	Час обчислення тривалий.
Метод скінченн их різниць у часовій області	Широкосмуговий відгук обчислюється в широкому діапазоні частот, зазвичай потрібно ~ 3 години. Знаходиться компроміс між часом обчислення та довжиною елемента.	Може використовуватися для оцінки параметрів розсіювання частинки довільної форми.	Час обчислення тривалий. Значення діелектричної проникності повинні бути визначені в набагато ширшому діапазоні частот, ніж просто діапазон, який цікавить. Модель Друде-Лоренца може бути неточним відображенням експериментальних даних.
Метод запропон	Залежить від параметрів	Дуже швидкий час прогнозування.	Необхідність в великій кількості

ований в дисертації йному дослідженні	нейромережі: гіперпараметри, кількості вихідних нейронів. Для метаматеріалів розмірністю 25 х 25 мм, час навчання 10 хвилин, а час прогнозування 5 с.	Може симулювати, як всю характеристику, так і для однієї одиниці параметра. Може використовуватись для прогнозування будь-яких метаматеріалів/матеріалів та структур. Проста у використанні система. Майже не залежить від складності метаматеріалів/матеріалів та структур.	даних або інформативності про ці дані. Тривалий час навчання. Залежність від роздільної здатності 3D моделей.
---------------------------------------	---	--	---

Потрібно зауважити, що порівняння відбувались на різних обчислювальних станціях. Так для всіх обчислювальних методів окрім методу глибокого навчання було використано персональний комп'ютер Dell Vostro 200 (2.19 GHz, 2 Gb RAM), а для методу глибокого навчання – відеокарта NVIDIA GeForce RTX 3050 Ti Laptop GPU.

4.4 Оптимізація якості прогнозування властивостей метаматеріалів

Підходів до розгляду шляхів щодо покращенню якості прогнозування характеристик метаматеріалів існує два – покращення архітектури нейромережі, або представлення даних у більш ефективному вигляді, або збільшення кількості та якості даних, які можуть впливати на електромагнітні характеристики метаматеріалів.

Оптимізувати постановку задачі не завжди вдається. Потрібно враховувати, що на стадії розгляду набору даних для нейронної мережі, не розроблена специфічна чітка архітектура та підхід до розв'язання конкретної задачі програмним методом. Звичайно, більшу частину часу розв'язок задачі є саме розгляд задачі та підготовка правильних даних у зручному для нейронної мережі вигляді. Попри це, час, який потребує процес формування необхідної архітектури, може бути великим. Тому найкраще вирішити всі завдання та установити чітко задачу на початкових стадіях розв'язку задач.

Потрібно зауважити, що найбільш важливим фактором є правильний підбір даних. Збільшення даних в цій задачі дисертаційної роботи не є найкращим підходом, через обмеженість доступу до них. Звичайно, є можливість розширювати спектр параметрів, які впливають на експериментальні дослідження метаструктур, тобто поляризація, температура, показник заломлення діелектрика та ін. Робіт, які мають потрібні дані (експериментальні графіки) з врахованими параметрами, набагато більше і тому є можливість збільшити дані для навчання нейронної мережі. Звичайно, після цього час, який буде займати процес навчання нейронної мережі, може значно збільшитися.

Підхід до оптимізації архітектури нейромережі був розглянутий в попередньому підрозділі. Варто зазначити, що цей підхід є найбільш популярним в ситуаціях, коли гнучкість задачі не дозволяє змінити підхід до розв'язання задачі, або немає достатньо даних, які відповідають постановці задачі. На даний час, в області глибокого навчання, рівень підходів та технік до оптимізації гіперпараметрів та архітектури нейромережі, достатньо широко. Існує велика кількість інформації та технік, які дозволяють правильно та ефективно розв'язувати ті чи інші задачі. Також існує величезна кількість програмних бібліотек, які в першу чергу призначені для реалізації процесів оптимізації гіперпараметрів та зміни архітектури в разі необхідності.

Найбільш ефективні є такі підходи:

Налаштування гіперпараметрів допомагає знайти значення, що забезпечують найкращі здатність до навчання та продуктивність моделі. Використовуються алгоритми автоматичного налаштування оптимальних гіперпараметрів глибокого навчання.

Використання попередньо обрахованих ваг на іншому наборі даних. Тут використовуються переваги вивчення попередніх даних, які можуть бути корисними під час навчання або навіть прямого прогнозування в другому налаштуванні.

Балансування даних: дані для навчання повинні точно відображати фактичний стан об'єкту вивчення. Якщо виникає проблема існування роздільних кількох класів і є лише кілька точок даних для одного з цих класів, тоді потрібно балансування даних.

Налаштування гіперпараметрів

Нижче наведено гіперпараметри, які можна налаштувати для покращення продуктивності нейронної мережі.

Швидкість навчання контролює, наскільки потрібно коригувати ваги щодо градієнта функцій втрат. Якщо швидкість навчання низька, це повільно змінюватиме ваги. Це призведе до уповільнення конвергенції до глобальних мінімумів функції втрат і призведе до перенавчання. Якщо швидкість навчання висока, це не дозволить реалізувати збіжність градієнтного спуску до мінімуму функції втрат.

Функція активації: механізм, за допомогою якого нейрони обробляють і передають інформацію через нейронну мережу. Функція активації визначає степінь впливу певного нейрона. Найбільш популярні функції активації: *binary*, *sigmoid*, *tanh*, *softmax*, *ReLU*, *Leaky ReLU* [150].

Порції: розмір порції визначає кількість зразків, які необхідно обробити перед оновленням параметрів моделі. Ділиться набір даних на кілька порцій. У порційному градієнтному спуску розмір порції такий самий, як розмір навчального набору даних. У стохастичному градієнтному спуску розмір порції завжди дорівнює

одній навчальній точці даних. У мінісерії розмір порції зазвичай кратний степеню двійки: 32, 64, 128 тощо.

Епохи: скільки разів модель проходить весь навчальний набір протягом процесу навчання. Це включає як пряме, так і зворотне поширення.

Регуляризація: техніка запобігання складності моделі. Це робиться шляхом штрафування функції втрат. Штрафуються всі ваги, роблячи їх маленькими. Це перетворює модель на простішу, ефективну та менш схильну до перенавчання.

Оптимізація: оптимізатор оновлює параметри ваги, щоб мінімізувати функцію втрат. Поширеними оптимізаторами є *Adam*, *RMSPprop*, *Momentum*, *Adagrad*, *Adadelta*.

Набір алгоритмів

Якщо окремі нейронні мережі не такі точні, як хотілося б, є можливість створити ансамбль нейронних мереж і об'єднати їх передбачувани можливості. Є можливість вибрати різні архітектури нейронних мереж і навчити їх на різних частинах даних, об'єднати їх і використати їхню сукупну передбачувану можливість, щоб отримати високу точність тестових даних. Припустімо, створюється класифікатор котів проти собак, 0 «кіт» і 1 – «собака». При поєднанні різних класифікаторів котів проти собак точність алгоритму ансамблю підвищується на основі кореляції Пірсона [152] між окремими класифікаторами [151].

Перевірка на перенавчання

Перший крок у забезпеченні належної роботи нейронної мережі на даних тестування — це переконатися, що нейронна мережа не перенавчається. Перенавчання відбувається, коли модель починає запам'ятовувати значення з навчальних даних замість того, щоб вчитися на них. Таким чином, коли модель стикається з даними, яких вона раніше не бачила, вона не може добре працювати з ними.

Щоб визначити, чи перенавчилася модель, потрібно порівняти точність навчання та точність тестування. Якщо точність навчання набагато вища за

точність тестування, можна стверджувати, що модель перенавчена. Є також можливість нанести прогнозовані точки на графік для перевірки. Існують деякі способи, щоб уникнути перенавчання – рис. 4.11:



Рисунок 4.11 – Графічний (функціональний) вигляд процесу перенавчання, не до навчання та нормального навчання нейронної мережі [151].

- Регуляризація даних.
- Видалення — випадкове скидання зв'язків між нейронами, змушуючи мережу знаходити нові шляхи та узагальнювати внутрішні зв'язки.
- Рання зупинка — прискорює навчання нейронної мережі, що призводить до зменшення помилок у тестовому наборі.

При глибокому навчанні нейронної мережі, вагові коефіцієнти стають більш спеціалізованими для навчальних даних, що спричиняє їх перенавчання. Під час ранньої зупинки відстежується продуктивність моделі порівняно з наборами даних навчання та перевірки. Коли є погіршення продуктивності набору даних перевірки, навчання негайно припиняється.

Якщо після виконання всіх наведених вище методів модель все ще не дає очікуваних результатів у конкретному тестовому наборі даних, це можна пояснити відсутністю навчальних даних, або їх низькою якістю. Існує багато випадків використання, коли кількість доступних навчальних даних обмежена. Якщо немає можливості зібрати більше даних, є можливість вдаватися до методів збільшення даних [154].

Для прикладу, якщо йде робота над набором зображень, є можливість додати нові зображення до навчальних даних, зсуваючи зображення, повертаючи зображення, довільно обрізаючи зображення тощо. Це може надати різні приклади для збільшення даних навчання нейронної мережі.

Висновки до розділу 4

Встановлено, що архітектури для двох випадків представлення вихідних параметрів – в цифровому-табличному вигляді та у вигляді графу навчання нейронної мережі дають можливість прогнозувати властивості 3D структур для одного частотного діапазону. Визначено всі шари, а також кількість вхідних та вихідних параметрів в кожному шарі.

Було виявлено, що одержані результати є задовільними для оцінки загальної похибки, але не є задовільними для випадку представлення експериментальних характеристик у вигляді коефіцієнтів полінома, тому що навіть мала похибка дає зміни у коефіцієнтах, що призводить до великої зміни у вигляді графіків поліномів. Метод представлення графіків, як координатних є кращим, але довшим в обробці нейронною мережею.

Виявлено, що метод представлення інформації про структуру та склад метаматеріалів, у форматі 3D об'єктів, де кожен піксель несе інформацію про властивості матеріалу, є зручним для навчання нейронних мереж. Показано, можливість використання такого способу представлення даних про топологічні 3D структури та склад для прогнозування їх властивостей.

ОСНОВНІ НАУКОВІ ТА ПРАКТИЧНІ РЕЗУЛЬТАТИ І ВИСНОВКИ

1) В результаті прогнозування властивостей метаматеріалів було згенеровано залежності коефіцієнта пропускання від частоти. Залежності були згенеровані у двох представленнях – як коефіцієнти поліномів та у вигляді координатних точок графіків. Для випадку з представлення як координатних точок графіків, було згенеровано окремі графіки коефіцієнта пропускання від масштабованої частоти та графіки частоти від номера нейрона. Таким чином відносна середня похибка склала 0,21. Для практичного використання даного представлення такий метод є ефективним, при більшій кількості навчальних даних та близькості діапазонів графіків властивостей. Для випадку з представлення, як коефіцієнтів полінома відносна середня похибка склала 0,09. Обернена конвертація коефіцієнтів полінома до вигляду графіків залежностей коефіцієнтів пропускання від частоти не дало позитивні результати з точки зору практичного застосування методу представлення у вигляді коефіцієнтів, але корисно з точки зору отримання інформації про ефективність та можливість навчання згорткової нейронної мережі з даною архітектурою, параметрами та навчальними даними. Переваги такого підходу полягають в універсальності – метаматеріали будь-якої складності можуть бути використані, швидкості генерації – залежить від кількості та роздільності даних, простота оптимізації – налаштування всіх параметрів архітектури.

2) Встановлено, що дані збережені в «.ply» форматі, не підходять, як вхідні дані для нейронної мережі та тому був використаний програмний пакет *Open3D* на базі *Python*, для конвертації початкового формату у формат «.xyzrgb», який становить собою масив, де кожен рядок – це вектор з шести чисел; перші три – це координати, три інших – це *RGB* числа. Показана можливість використання такого вигляду інформації в процесах згортки та для навчання нейронної мережі. Представлено умови, які накладаються на дані: однорідність даних, що означає, що всі параметри, які впливають на результат прогнозування, мають або бути однаковими, або бути внесені в дані, як інформація про властивості; масштабованість, що означає, що дані мають бути зведені до однакових діапазонів значень та масштабів.

3) Встановлено, що архітектури для двох випадків представлення вихідних параметрів – в цифровому-табличному вигляді для навчання нейронної мережі дають можливість прогнозувати властивості 3D структур для одного частотного діапазону. Визначено всі шари нейронної мережі, а також кількість вхідних та вихідних параметрів в кожному шарі.

4) Встановлено формати, які мають можливість зберігати додаткову інформацію про 3D-об'єкти – текстуру, колір, взаємне розташування і т. д. Описано можливості використання додаткових даних, щоб зберігати інформацію про властивості реальних 3D-структур. Встановлено, що формат «.ply», має перевагу в тому, що несе інформацію про координати всіх пікселів та колір (*RGB*), який закріплений до кожного пікселю. Проаналізовано інші формати збереження інформації про 3D структури. Досліджено, що такі формати, як «.fbx», «.obj», «.stl», «.3ds» не підходять, через такі причини: відсутність ефективного програмного функціонала (для конвертування в потрібний для нейронної мережі формат); відсутність збереження інформації про додаткові канали інформації; присутність зайвої інформації, яку немає можливості відділити від необхідної.

5) Проаналізовано, що використання 3D структур та їх властивостей у різних частотних діапазонах можливо тільки при достатній кількості різних 3D структур. Методом глибокого навчання можна аналізувати метаматеріали, тільки якщо присутня достатня кількість даних для різних частотних діапазонів. Оцінити точне співвідношення потрібної кількості даних для аналізу різних частотних діапазонів немає можливості, через те, що відсутні конструктивні дослідження в цьому напрямі.

СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

- [1] X. Yu, J. Zhou, H. Liang, Z. Jiang and L. Wu, “Mechanical metamaterials associated with stiffness, rigidity and compressibility: A brief review,” *Progress in Materials Science*, vol. 94, pp. 114-173, May. 2018. doi:10.1016/j.pmatsci.2017.12.003.
- [2] M. Maasch, “Tunable microwave metamaterial structures,” Ph.D. dissertation, Electron. Eng., Technische Universität Darmstadt, 2016. [Online]. Available: Proquest. Accessed on: 08.02.2016.
- [3] G. Goussetis, A. Feresidis and A. Harvey, “Experimental realisation of electromagnetic metamaterials,” *Journal of Modern Optics - J MOD OPTIC*. vol. 57, no. 1, pp. 1-16. Jan. 2010. doi:10.1080/09500340903497515.
- [4] X. Cai, R. Zhu and G. Hu. “Experimental study for metamaterials based on dielectric resonators and wire frame,” *Metamaterials*, vol. 2, pp. 220-226. Dec. 2008. doi:10.1016/j.metmat.2008.08.001.
- [5] I. Shadrivov, D. Powell, S. Morrison, G. Milford and Y. Kivshar, *Experimental studies of binary metamaterials: CLEO/Europe and IQEC 2007 Conference Digest*, June 17-22, 2007 Munich, Germany. Optica Publishing Group, 2007.
- [6] A. Arjunan, A. Baroutaji and J. Robinson, “Advances in Acoustic Metamaterials,” *Encyclopedia of Smart Materials*, vol. 3, pp. 1-10. Jan. 2021. doi:10.1016/B978-0-12-815732-9.00091-7.
- [7] M. Zoziuk, O. Yurikov, D. Koroliouk and Y. Yakymenko, “The Principle of Creating Quasiperiodic Surfaces under the Action of a Vibrating Dielectric Matrix,” *Microsystems, Electronics and Acoustics*, vol. 25, no. 1, 2020, pp. 5-10, Dec. 2020. doi:10.20535/2523-4455.me.202632.
- [8] P. Krysenko, M. Zoziuk, O. Yurikov, D. Koroliuk and Yu. Yakymenko, “Chladni Figures Simulation on a Rectangular Plate,” *Microsystems, Electronics and Acoustics*, vol. 26, no. 1, 2021, pp. 241698.1-241698.6, Dec. 2021. doi:10.20535/2523-4455.me.241698.

- [9] J. Jiang, M. Chen and J. Fan, “Deep neural networks for the evaluation and design of photonic devices,” *Nature Reviews Materials*, vol. 6, pp. 679–700. Dec. 2021. doi:10.1038/s41578-020-00260-1.
- [10] Wu. Lingling, L. Lei, W. Yong, Z. Zirui, Z. Houlong, K. Deepakshyam, W. Qianxuan and J. Hanqing, “A machine learning-based method to design modular metamaterials,” *Extreme Mechanics Letters*, vol. 36, no. 100657, Apr. 2020. doi:10.1016/j.eml.2020.100657.
- [11] O. Khatib, S. Ren, J. Malof and W. Padilla, “Deep Learning the Electromagnetic Properties of Metamaterials—A Comprehensive Review,” *Adv. Funct. Mater.*, vol. 31, no. 2101748, May. 2021. doi:10.1002/adfm.202101748.
- [12] J. Masson, J. Biggins and E. Ringe, “Machine learning for nanoplasmonics,” *Nature Nanotechnology*, vol. 18, pp. 111–123, Jan. 2023, doi:10.1038/s41565-022-01284-0.
- [13] F. Demore, G. Rizzi, M. Collet, P. Neff and A. Madeo, “Unfolding engineering metamaterials design: Relaxed micromorphic modeling of large-scale acoustic meta-structures,” *Journal of the Mechanics and Physics of Solids*, vol. 168, no. 104995, Nov. 2022. doi:10.1016/j.jmps.2022.104995.
- [14] L. Yanglong and W. Yan, “Structural optimization of metamaterials based on periodic surface modeling,” *Computer Methods in Applied Mechanics and Engineering*, vol. 395, no. 115057, May 2022. doi:10.1016/j.cma.2022.115057.
- [15] W. Pabst, S. Hříbalová and T. Uhlířová. “Quasi-laminate and quasi-columnate modeling of dielectric and piezoelectric properties of cubic-cell metamaterials,” *Journal of the European Ceramic Society*, vol. 42, no. 3, pp. 1396–1406, Apr. 2022. doi:10.1016/j.jeurceramsoc.2021.12.013.
- [16] Z. Shaoyu, Z. Yingyan, Z. Yihe, Z. Wei, Y. Jie and K. Sritawat, “Genetic programming-assisted micromechanical models of graphene origami-enabled metal metamaterials,” *Acta Materialia*, vol. 228, no. 117791, Apr. 2022. doi:10.1016/j.actamat.2022.117791.

- [17] T. Wei, Z. Tian, Y. Zhichun, “Theoretical modelling and design of metamaterial stiffened plate for vibration suppression and supersonic flutter,” *Composite Structures*, vol. 282, no. 115010. doi:0.1016/j.compstruct.2021.115010.
- [18] Z. Chen, G. Wang, Y. Mao and C. Lim, “New metamaterial mathematical modeling of acoustic topological insulators via tunable underwater local resonance,” *Applied Mathematical Modelling*, vol. 108, pp. 258-274, Aug. 2022. doi:10.1016/j.apm.2022.03.023.
- [19] Y. Chen, L. Ye, Y. Zhang and K. Fu, “Compression behaviours of 3D-printed CF/PA metamaterials: Experiment and modelling,” *International Journal of Mechanical Sciences*, vol. 206, no. 106634, Sep. 2021. doi:10.1016/j.ijmecsci.2021.106634.
- [20] S. Hříbalová, T. Uhlířová and W. Pabst, “Computer modeling of systematic processing defects on the thermal and elastic properties of open Kelvin-cell metamaterials,” *Journal of the European Ceramic Society*. vol. 41, no. 14, pp. 7130-7140, Nov. 2021. doi:10.1016/j.jeurceramsoc.2021.07.031.
- [21] P. Vogiatzis, S. Chen, X. Wang, T. Li and L. Wang, “Topology optimization of multi-material negative Poisson’s ratio metamaterials using a reconciled level set method,” *Computer-Aided Design*, vol. 83, pp. 15-32, Feb. 2017, doi:10.1016/j.cad.2016.09.009.
- [22] A. Radman, X. Huang and Y. Xie, “Topological optimization for the design of microstructures of isotropic cellular materials. Engineering Optimization,” *Computational Materials Design*, vol. 50, no. 6, pp. 1861-1870, Apr. 2011, doi:10.1080/0305215X.2012.737781.
- [23] B. Bickel, M. Bächer, M. Otaduy, H. Lee, H. Pfister, M. Gross, and W. Matusik, “Design and fabrication of materials with desired deformation behavior,” *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1-10, Jul. 2010, doi:10.1145/1778765.1778800.
- [24] P. Kumar, T. Ali and M. Pai, “Electromagnetic Metamaterials: A New Paradigm of Antenna Design,” *IEEE Access*, vol. 9, pp. 18722-18751, Jan. 2021, doi:10.1109/ACCESS.2021.3053100.

- [25] Y. Jin, L. He, Z. Wen, B. Mortazavi, H. Guo, D. Torrent, B. Djafari-Rouhani, T. Rabczuk, X. Zhuang, and Y. Li, "Intelligent on-demand design of phononic metamaterials," *Nanophotonics*, vol. 11, no.3, pp. 439-460, Jan. 2022, doi:10.1515/nanoph-2021-0639.
- [26] R. Trivedi, N. Rai and K. Bharadwaj, "Metamaterial: Materials with Exceptional Properties," *INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT)*. *ETRASC*, vol. 2, no. 3, pp. 367-373, Jul. 2018, doi:10.17577/IJERTCONV2IS03058.
- [27] M. Bessa, P. Glowacki and M. Houlder, "Bayesian Machine Learning in Metamaterial Design: Fragile Becomes Supercompressible," *Advanced Materials*, vol. 31, no. 48, pp. 1904845, Oct. 2019, doi: 10.1002/adma.201904845.
- [28] G. Freitas, S. Rêgo and C. Vasconcelos, "Design of metamaterials using artificial neural networks," presented at the *SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, 2012, pp. 541-545. Available: doi: 10.1109/IMOC.2011.6169403. Accessed on: March 15, 2012.
- [29] Z. Fang and J. Zhan, "Deep Physical Informed Neural Networks for Metamaterial Design," *IEEE Access*, vol. 8, pp. 24506-24513, Dec. 2019, doi:10.1109/ACCESS.2019.2963375.
- [30] L. Wang, Y. Chan, F. Ahmed, Z. Liu, P. Zhu and W. Chen, "Deep generative modeling for mechanistic-based learning and design of metamaterial systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 372, pp. 113377, Dec. 2020, doi:10.1016/j.cma.2020.113377.
- [31] R. Cang, H. Li, H. Yao, Y. Jiao and Y Ren, "Improving direct physical properties prediction of heterogeneous materials from imaging data via convolutional neural network and a morphology-aware generative model," *Computational Materials Science*, vol. 150, pp. 212-221, Jul. 2018, doi:10.1016/j.commatsci.2018.03.074.

- [32] B. Cunha, C. Droz, A. Zine, S. Foulard and M. Ichchou, “A review of machine learning methods applied to structural dynamics and vibroacoustic,” *Mechanical Systems and Signal Processing*, vol. 200, pp. 110535, Jun. 2023, doi:10.1016/j.ymssp.2023.110535.
- [33] J. Wang, W. Chen, D. Da, M. Fuge and R. Rai, “IH-GAN: A conditional generative model for implicit surface-based inverse design of cellular structures,” *Computer Methods in Applied Mechanics and Engineering*, vol. 396, pp. 115060, Jun. 2022, doi:10.1016/j.cma.2022.115060.
- [34] J. Montañez-Barrera, J. Barroso-Maldonado, A. Bedoya-Santacruz and A. Mota-Babiloni, “Correlated-informed neural networks: A new machine learning framework to predict pressure drop in micro-channels,” *International Journal of Heat and Mass Transfer*, vol. 194, pp. 123017, Sep. 2022, doi:10.1016/j.ijheatmasstransfer.2022.123017.
- [35] L. Wang, J. Boddapati, K. Liu, P. Zhu, C. Daraio and W. Chen, “Mechanical cloak via data-driven aperiodic metamaterial design,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 119, no. 13, Mar. 2022, doi:10.1073/pnas.2122185119.
- [36] R. Tan, N. Zhang and W. Ye, “A deep learning–based method for the design of microstructural materials,” *Struct. Multidisc. Optim.*, vol. 61, pp. 1417–1438. Nov. 2019, doi: 10.1007/s00158-019-02424-2.
- [37] C. Liu and G. Yu, “A deep learning model for the topological design of 2D periodic wave barriers,” *Construction and Building Materials*, vol. 4, no. 4, pp. 451-469, Jul. 2022, doi:10.1111/mice.12743.
- [38] C. Liu and G. Yu, “Intelligent design of engineered metabarrier based on deep learning,” *Composite Structures*, vol. 280, pp. 114911, Jan. 2022, doi: 10.1016/j.compstruct.2021.114911.

- [39] J. Bastek, S. Kumar, B. Telgen, R. Glaesener and D. Kochmann, “Inverting the structure–property map of truss metamaterials by deep learning,” *Proceedings of the National Academy of Sciences*, vol. 119, no. 1, Dec. 2021, doi:10.1073/pnas.2111505119.
- [40] C. Liu and G. Yu, “Deep learning-based topology design of periodic barrier for full-mode waves,” *Construction and Building Materials*, vol. 314, pp. 125579, Jan. 2022, doi:10.1016/j.conbuildmat.2021.125579.
- [41] S. Kadulkar, Z. Sherman, V. Ganesan and T. Truskett, “Machine Learning-Assisted Design of Material Properties”, *Annual review of chemical and biomolecular engineering*, vol. 13, pp. 235-254, Jun. 2022, doi:10.1146/annurev-chembioeng-092220-024340.
- [42] Z. Nie, H. Jiang and L. Kara, “Stress Field Prediction in Cantilevered Structures Using Convolutional Neural Networks,” *Journal of Computing and Information Science in Engineering*, vol. 20, no. 1, pp. 1, Jun. 2019, doi:10.1115/1.4044097.
- [43] D. White, W. Arrighi, J. Kudo and S. Watts, “Multiscale topology optimization using neural network surrogate models,” *Computer Methods in Applied Mechanics and Engineering*, vol. 346, pp. 1118-1135, Apr. 2019, doi:10.1016/j.cma.2018.09.007.
- [44] R. Cang, Y. Xu, S. Chen, Y. Liu, Y. Jiao and M. Ren, “Microstructure Representation and Reconstruction of Heterogeneous Materials Via Deep Belief Network for Computational Material Design”, *J. Mech. Des.*, vol. 139, no. 7, pp. 071404, Jul. 2017, doi:10.1115/1.4036649.
- [45] X. Li, S. Ning, Z. Liu, Z. Yan, C. Luo and Z. Zhuang, “Designing phononic crystal with anticipated band gap through a deep learning based data-driven method,” *Computer Methods in Applied Mechanics and Engineering*, vol. 361, pp. 112737, Apr. 2020, doi:10.1016/j.cma.2019.112737.
- [46] P. Tahmasebi and A. Hezarkhani, “A fast and independent architecture of artificial neural network for permeability prediction,” *Journal of Petroleum Science and Engineering*, vol. 86-87, pp. 118-126, May. 2012, doi:10.1016/j.petrol.2012.03.019.

- [47] C. Qian, R. Kai Tan and W. Ye, “An adaptive artificial neural network-based generative design method for layout designs,” *International Journal of Heat and Mass Transfer*, vol. 184, pp. 122313, Mar. 2022, doi: 10.1016/j.ijheatmasstransfer.2021.122313.
- [48] C. Duan, A. Nandy and H. Kulik, “Machine Learning for the Discovery, Design, and Engineering of Materials,” *Annual review of chemical and biomolecular engineering*, vol. 13, no. 1, pp. 405-429, Jun. 2022, doi:10.1146/annurev-chembioeng-092320-120230.
- [49] Z. Liu, L. Raju, D. Zhu and W. Cai, “A Hybrid Strategy for the Discovery and Design of Photonic Structures,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 10, no. 1, pp. 126-135, Mar. 2020, doi:10.1109/JETCAS.2020.2970080.
- [50] W. Ma, F. Cheng, Y. Xu, Q. Wen and Y. Liu, “Probabilistic Representation and Inverse Design of Metamaterials Based on a Deep Generative Model with Semi-Supervised Learning Strategy,” *Advanced Materials*, vol. 31, no. 35, pp. 1901111, Aug. 2019, doi:10.1002/adma.201901111.
- [51] Y. Tang, K. Kojima, T. Koike-Akino, Y. Wang, P. Wu, M. Tahersima, D. Jha, K. Parsons, and M. Qi, “Generative Deep Learning Model for a Multi-level Nano-Optic Broadband Power Splitter,” presented at the IEEE. Optical Fiber Communication Conference (OFC), 2020, pp. 1-3. Available: ieeexplore.ieee.org/document/9083040. Accessed on: May 04, 2020.
- [52] R. Gómez-Bombarelli, J. Wei, D. Duvenaud, J. Miguel Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. Hirzel, R. Adams and A. Aspuru-Guzik, “Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules,” *ACS Central Science*, vol. 4, no. 2, pp. 268-276, Jan. 2018, doi:10.1021/acscentsci.7b00572.
- [53] M. Segler, M. Preuss and M. Waller, “Planning chemical syntheses with deep neural networks and symbolic,” *Nature*, vol. 555, pp. 604-610, Mar. 2018, doi:10.1038/nature25978.

- [54] J. Granda, L. Donina and V. Dragone, "Controlling an organic synthesis robot with machine learning to search for new reactivity," *Nature*, vol. 559, pp. 377-381, Jul. 2018, doi:10.1038/s41586-018-0307-8.
- [55] P. Jorgensen, M. Schmidt and O. Winther, "Deep Generative Models for Molecular Science," *Molecular informatics*, vol. 37, no. 1-2, pp. 1700133, Jan. 2018, doi: 10.1002/minf.201700133.
- [56] I. Malkiel, M. Mrejen and A. Nagler, "Plasmonic nanostructure design and characterization via Deep Learning," *Light: Science & Applications*, vol. 7, no. 60, Sep. 2018, doi:10.1038/s41377-018-0060-7.
- [57] T. James, and D. Hristozov, "Deep Learning and Computational Chemistry," *Methods in molecular biology (Clifton, N.J.)*, vol. 2390, pp. 125-151, Nov. 2021, doi: 10.1007/978-1-0716-1787-8_5.
- [58] Z. Liu, D. Zhu, K. Lee, A. Kim, L. Raju and W. Cai, "Compounding Meta-Atoms into Metamolecules with Hybrid Artificial Intelligence Techniques," *Advanced Materials*, vol. 32, no. 6, pp. 1904790, Feb. 2020, doi:10.1002/adma.201904790.
- [59] M. Nielsen, *Neural Networks and Deep Learning*, Dec. 2018. [Online]. Available: <http://neuralnetworksanddeeplearning.com/> Accessed on: 2015.
- [60] A. Bomba, M. Nazaruk, N. Kunanets and V. Pasichnyk, "Modeling the Dynamics of Knowledge Potential of Agents in the Educational Social and Communication Environment," presented at *Advances in Intelligent Systems and Computing IV*, 2019 vol. 1080, pp. 17-24. Available: doi:10.1007/978-3-030-33695-0_2. Accessed on: November 02, 2019.
- [61] I. Krak, O. Barmak and E. Manziuk, "Using visual analytics to develop human and machine-centric models: A review of approaches and proposed information technology," *Computational Intelligence*, vol. 38, no. 3, pp. 921-946, Jun. 2022, doi:10.1111/coin.12289.

- [62] В. Грицик, Ю. Крак, А. Куляс, А. Шевчунко, Е. Кісельова, С. Яковлєв, П. Бидюк, Н. Гіль, Т. Романова та П. Стецюк, *Математичні методи оптимізації та інтелектуальні комп'ютерні технології моделювання складних процесів і систем з урахуванням просторових форм об'єктів*. Донецьк: ІПП "Наука і освіта", 2011.
- [63] V. Pasichnyk, A. Bomba, M. Nazaruk and N. Kunanets, "The dynamics simulation of knowledge potentials of agents including the feedback," presented at the XII International Conference on Mathematics, Science and Technology Education (ICon-MaSTEd 2020). 2020, pp. 012020. Available: doi:10.1088/1742-6596/1840/1/012020. Accessed on: March 01, 2020.
- [64] R. Tan, C. Qian, M. Wang and W. Ye, "An efficient data generation method for ANN-based surrogate models," *Structural and Multidisciplinary Optimization*, vol. 65, no. 90, Feb. 2022, doi:10.1007/s00158-022-03180-6.
- [65] L. Wang, S. Yerramilli, A. Iyer, D. Apley, P. Zhu and W. Chen, "Scalable Gaussian Processes for Data-Driven Design Using Big Data with Categorical Factors," *Journal of Mechanical Design*, vol. 144, no. 2, pp. 1-36, Aug. 2021, doi:10.1115/1.4052221.
- [66] Y. Chen, Z. Lin, X. Zhao, G. Wang and Y. Gu, "Deep Learning-Based Classification of Hyperspectral Data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 6, pp. 2094-2107, Jun. 2014, doi:10.1109/JSTARS.2014.2329330.
- [67] L. Li, H. Ruan, C. Liu, Y. Li, Y. Shuang, A. Alu, C. Qiu and T. Cui, "Machine-learning reprogrammable metasurface imager," *Nature*, vol. 10, no. 1082, Mar. 2019, doi:10.1038/s41467-019-09103-2.
- [68] W. Ma, Z. Liu, Z. Kudyshev, A. Boltasseva, W. Cai and Y. Liu, "Deep learning for the design of photonic structures," *Nature*, vol. 15, pp. 77-90, Oct. 2020, doi:10.1038/s41566-020-0685-y.

- [69] A. Hammond and R. Camacho, "Designing integrated photonic devices using artificial neural networks," *Optics Express*, vol. 27, no. 21, pp. 29620-29638, Sep. 2019, doi:10.1364/OE.27.029620.
- [70] O. Khatib, S. Ren, J. Malof and W. Padilla, "Deep Learning the Electromagnetic Properties of Metamaterials—A Comprehensive Review," *Advanced Functional Materials*, vol. 31, no. 31, pp. 2101748, Aug. 2021, doi:10.1002/adfm.202101748.
- [71] Y. Xu, X. Zhang, Y. Fu and Y. Liu, "Interfacing photonics with artificial intelligence: an innovative design strategy for photonic structures and devices based on artificial neural networks," *Photonics Research*, vol. 9, no. 4, pp. 135-152, Mar. 2021, doi:10.1364/PRJ.417693.
- [72] J. Jiang, D. Sell, S. Hoyer, J. Hickey, J. Yang and J. Fan, "Free-Form Diffractive Metagrating Design Based on Generative Adversarial Network," *ACS Nano*, vol. 13, no. 8, pp. 8872-8878, Jul. 2019, doi:10.1021/acsnano.9b02371.
- [73] W. Kong, J. Chen, Z. Huang and D. Kuang, "Bidirectional cascaded deep neural networks with a pretrained autoencoder for dielectric metasurfaces," *Photonics Research*, vol. 9, no. 8, pp. 1607-1615, Jul. 2021, doi:10.1364/PRJ.428425.
- [74] C. Nadell, B. Huang, J. Malof and W. Padilla, "Deep learning for accelerated all-dielectric metasurface design," *Optics Express*, vol. 27, no. 20, pp. 27523-27535, Sep. 2019. doi:10.1364/OE.27.027523.
- [75] J. Peurifoy, Y. Shen, L. Jing, Y. Yang, F. Cano-Renteria, B. Delacy, M. Tegmark, J. Joannopoulos and M. Soljačić, "Nanophotonic particle simulation and inverse design using artificial neural networks," *Physics and Simulation of Optoelectronic Devices XXVI*, vol. 4, no. 6, Jun. 2018, doi: 10.1117/12.2289195.
- [76] D. Liu, Y. Tan, E. Khoram and Z. Yu, "Training Deep Neural Networks for the Inverse Design of Nanophotonic Structures," *ACS Photonics*, vol. 5, no. 4, pp. 1365-1369, doi:10.1364/CLEO_AT.2019.JF2F.4.

- [77] Y. Chen, J. Zhu, Y. Xie, N. Feng and Q. Liu, "Smart inverse design of graphene-based photonic metamaterials by an adaptive artificial neural network," *Nanoscale*, vol. 11, no. 19, May 2019, doi:10.1039/C9NR01315F.
- [78] I. Sajedian, J. Kim and J. Rho, "Finding the optical properties of plasmonic structures by image processing using a combination of convolutional neural networks and recurrent neural networks," *Microsyst Nanoeng*, vol. 5, no. 27, Jun. 2019, doi:10.1038/s41378-019-0069-y.
- [79] Z. Liu, L. Raju, D. Zhu and W. Cai, "A Hybrid Strategy for the Discovery and Design of Photonic Structures," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 10, no. 1, pp. 126-135, Mar. 2020, doi:10.1109/JETCAS.2020.2970080.
- [80] S. So and J. Rho, "Designing nanophotonic structures using conditional deep convolutional generative adversarial networks," *Nanophotonics*, vol. 8, no. 7, Jun. 2019, pp. 1255-1261, doi:10.1515/nanoph-2019-0117.
- [81] Z. Liu, D. Zhu, S. Rodrigues, K. Lee and W. Cai, "A Generative Model for Inverse Design of Metamaterials," *Nano Letters (ACS Publications)*, vol. 18, no. 10, pp. 6570-6576, Sep. 2018, doi:10.1021/acs.nanolett.8b03171.
- [82] W. Ma and Y. Liu, "A data-efficient self-supervised deep learning model for design and characterization of nanophotonic structures," *Science China Physics, Mechanics and Astronomy*, vol. 63, no. 284212, Jun. 2020, doi:10.1007/s11433-020-1575-2.
- [83] A. Mall, A. Patil, D. Tamboli, A. Seth and A. Kumar, "Fast design of plasmonic metasurfaces enabled by deep learning," *Journal of Physics D: Applied Physics*, vol. 53, no. 49, Aug. 2020, doi:10.1088/1361-6463/abb33c.
- [84] M. Tahersima, K. Kojima, T. Koike-Akino, D. Jha, B. Wang, C. Lin and K. Parsons, "Deep Neural Network Inverse Design of Integrated Photonic Power Splitters," vol. 9, no. 1, pp. 1368, Feb. 2019, doi:10.1038/s41598-018-37952-2.

- [85] T. Qiu, X. Shi, J. Wang, Y. Li, S. Qu, Q. Cheng, T. Cui and S. Sui, "Deep Learning: A Rapid and Efficient Route to Automatic Metasurface Design," *Advanced Science*, vol. 6, no. 12, pp. 1900128, Jun. 2019, doi:10.1002/advs.201900128.
- [86] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems*, vol. 2, pp. 303-314, Dec. 1989, doi:10.1007/BF02551274.
- [87] C. Gurbuz, F. Kronowetter, C. Dietz, M. Eser, J. Schmid and S. Marburg, "Generative adversarial networks for the design of acoustic metamaterials," *The Journal of the Acoustical Society of America*, vol. 149, pp. 1169-1174, Feb. 2021, doi:10.1121/10.0003501.
- [88] B. Sanchez-Lengeling and A. Aspuru-Guzik, "Inverse molecular design using machine learning: Generative models for matter engineering," *Science*, vol. 361, no. 6400, pp. 360-365, Jul. 2018, doi:10.1126/science.aat2663.
- [89] T. Xie and J. Grossman, Jeffrey, "Crystal Graph Convolutional Neural Networks for Accurate and Interpretable Prediction of Material Properties," *Physical Review Letters*, vol. 120, no. 14, pp. 145301, Apr. 2018, doi:10.1103/PhysRevLett.120.145301.
- [90] X. Zhou, Q. Xiao and H. Wang, "Metamaterials Design Method based on Deep Learning Database," in *2021 International Conference on Advanced Technologies and Applications of Modern Industry (ATAMI 2021)*, pp. 12
- [91] S. Lim, S. Young and R. Patton, "An analysis of image storage systems for scalable training of deep neural networks," in *An analysis of image storage systems for scalable training of deep neural networks*.
- [92] L. Cong, Y. Srivastava and R. Singh, "Inter and intra-metamolecular interaction enabled broadband high-efficiency polarization control in metasurfaces," *Applied Physics Letters*, vol. 108, no. 1, pp. 011110, Jan. 2016, doi:10.1063/1.4939564.

- [93] F. Ding, Z. Wang, S. He, V. Shalaev and A. Kildishev, "Broadband High-Efficiency Half-Wave Plate: A Super-Cell Based Plasmonic Metasurface Approach," *ACS Nano*, vol. 9, no. 4, pp. 4111-4119, Mar. 2015, doi:10.1021/acsnano.5b00218.
- [94] C. Huang, "Efficient and broadband polarization conversion with the coupled metasurfaces," *Optics Express*, vol. 23, no. 25, pp. 32015-32024, Dec. 2015, doi:10.1364/OE.23.032015.
- [95] A. Shaltout, J. Liu, A. Kildishev and V. Shalaev, "Photonic spin Hall effect in gap-plasmon metasurfaces for on-chip chiroptical spectroscopy," *Optica*, vol. 2, no. 10, pp. 860-863, Sep. 2015, doi:10.1364/OPTICA.2.000860.
- [96] N. Grady, J. Heyes, C. Roy, Z. Yong, R. Matthew, A. Abul, T. Antoinette and H. Chen, "Terahertz Metamaterials for Linear Polarization Conversion and Anomalous Refraction," *Science*, vol. 340, no. 6138, pp. 1304-1307, May 2013, doi:10.1126/science.1235399.
- [97] *Foundations of 3D printing and CAD file formats used in the industry*, in *Multimaterial 3D Printing Technology*, pp. 17-42, J. Yang, N. Li, J. Shi, W. Tang, G. Zhang and F. Zhang, Elsevier Ltd, 2021.
- [98] M. Zozyuk, D. Koroliouk, P. Krysenko, A. Yurikov and Y. Yakymenko, "Prediction of characteristics using a convolutional neural network based on experimental data on the structure and composition of metamaterials," *STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING*, vol. 11, no. 3, pp. 777-787, Jun. 2023, doi:10.19139/soic-2310-5070-1707.
- [99] The 10 Most Popular 3D File Formats. All3DP. 2022. [Online]. Available: <https://all3dp.com/2/most-common-3d-file-formats-model>. Accessed on: January 27, 2022.
- [100] I. Goodfellow, Y. Bengio and A. Courville. *Deep Learning*, November 18, 2016. [Online]. Available: <https://www.deeplearningbook.org/> Accessed on: 18.11.2016.

- [101] Y. Boureau, J. Ponce and Y. Lecun, “A Theoretical Analysis of Feature Pooling in Visual Recognition,” in *ICML 2010 – Proceedings. 27th International Conference on Machine Learning*, pp. 111-118.
- [102] Y. Boureau, N. Roux, F. Bach, J. Ponce and Y. Lecun, “Ask the locals: multi-way local pooling for image recognition,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2651-2658.
- [103] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, “Backpropagation applied to handwritten zip code recognition,” *Neural Computation*, vol. 1, no. 4, pp. 541-551, Dec. 1989, doi:10.1162/neco.1989.1.4.541.
- [104] K. Gregor and Y. LeCun, “Learning fast approximations of sparse coding,” in *In Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML'10)*, pp. 399–406.
- [105] B. LE, N. DOVE, C. AGNEW, M. KORN and A. MUTSO, “Predicting nonmarital romantic relationship dissolution: A meta-analytic synthesis,” *Personal Relationships*, vol. 17, no. 3, pp. 377-390, Jul. 2020, doi:10.1111/j.1475-6811.2010.01285.x.
- [106] D. Sona, A. Sperduti and A. Starita, “Discriminant pattern recognition using transformation-invariant neurons,” *Neural computation*, vol. 12, no. 6, pp. 1255-1270, Jun. 2000, doi:10.1162/089976600300015402.
- [107] I. Goodfellow, “Multidimensional, Downsampled Convolution for Autoencoders,” Department of Computer Science and Operations Research, Montreal, 2010.
- [108] A. Jain and A. Kumar, “Hybrid Neural Network Models for Hydrologic Time Series Forecasting,” *Applied Soft Computing*, vol. 7, no. 2, pp. 585-592, Mar. 2007, doi:10.1016/j.asoc.2006.03.002.
- [109] P. Pinheiro and R. Collobert, “From image-level to pixel-level labeling with Convolutional Networks,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1713-1721.

- [110] K. Briggman, W. Denk, S. Seung, M. Helmstaedter and S. Turaga, “Maximin affinity learning of image segmentation,” in *NIPS'09: Proceedings of the 22nd International Conference on Neural Information Processing Systems*, pp. 1865-1873.
- [111] S. Turaga, J. Murray, V. Jain, F. Roth, M. Helmstaedter, K. Briggman, W. Denk and H. Seung, “Convolutional networks can learn to generate affinity graphs for image segmentation,” *Neural Computation*, vol. 22, no. 2, pp. 551-538, Feb. 2010, doi:10.1162/neco.2009.10-08-881.
- [112] C. Farabet, C. Couprie, L. Najman and Y. LeCun, “Learning Hierarchical Features for Scene Labeling,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915-1929, Aug. 2013, doi:10.1109/TPAMI.2012.231.
- [113] F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou and P. Barbano, “Toward automatic phenotyping of developing embryos from videos,” *IEEE Transactions Image Processing*, vol. 14, no. 9, pp. 1360-1371, Sep. 2005, doi:10.1109/TIP.2005.852470.
- [114] J. Tompson, A. Jain, Y. Lecun and C. Bregler, “Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation,” in *NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems*, pp. 1799-1807.
- [115] A. Coates, H. Lee and A. Ng, “An analysis of single-layer networks in unsupervised feature learning,” in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 215-223.
- [116] K. Jarrett, K. Kavukcuoglu, M. Ranzato and Y. LeCun, “What is the best multi-stage architecture for object recognition?,” in *IEEE 12th International Conference on Computer Vision*, pp. 2146-2153.
- [117] A. Saxe, P. Koh, Z. Chen, M. Bhand, B. Suresh and A. Ng, “On random weights and unsupervised feature learning,” in *In Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML'11)*, pp. 1089-1096.

- [118] N. Pinto, Z. Stone, T. Zickler and D. Cox, “Scaling up biologically-inspired computer vision: A case study in unconstrained face recognition on facebook,” in *CVPR 2011 WORKSHOPS*, pp. 35-42.
- [119] D. Cox and N. Pinto, “Beyond simple features: A large-scale feature search approach to unconstrained face recognition,” in *2011 IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, pp. 8-15.
- [120] H. Lee, R. Grosse, R. Ranganath and A. Ng, “Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations,” in *In Proceedings of the 26th Annual International Conference on Machine Learning (ICML '09)*, pp. 609-616.
- [121] M. Ranzato, F. J. Huang, Y. -L. Boureau and Y. LeCun, “Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8.
- [122] A. Orenchia, J. Coching, A. Matias, E. Dadios, R. Baldovino and R. Billones, “A Comparative Study on the Use of Raw and Filtered Images for Multi-class Image Classification,” in *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, pp. 1-6.
- [123] K. Kavukcuoglu, P. Sermanet, Y. Boureau, K. Gregor, M. Mathieu and Y. Lecun, “Learning Convolutional Feature Hierarchies for Visual Recognition,” in *Advances in Neural Information Processing Systems*, pp. 1090-1098.
- [124] A. Coates, B. Huval, T. Wang, D. Wu, B. Catanzaro and N. Andrew, “Deep learning with COTS HPC systems,” in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pp. 1337–1345.
- [125] K. Hornik, M. Stinchcombe and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359-366, 1989 doi:10.1016/0893-6080(89)90020-8.

- [126] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Math. Of Control, Signals and Systems*, vol. 2, pp. 303-314, Dec. 1989, doi:10.1007/BF02551274.
- [127] K. Hornik, M. Stinchcombe and H. White, "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks," *Neural Networks*, vol. 3, no. 5, pp. 551-560, Jan. 1990, doi:10.1016/0893-6080(90)90005-6.
- [128] M. Leshno, V. Lin, A. Pinkus and S. Schocken, "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural Networks*, vol. 6, no. 6, pp. 861-867, Sep. 1993, doi:10.1016/S0893-6080(05)80131-5.
- [129] A. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 930-945, May 1993, doi:10.1109/18.256500.
- [130] J. Hastad, "Almost optimal lower bounds for small depth circuits," in *Proceedings of the eighteenth annual ACM symposium on Theory of computing (STOC '86)*, pp. 6-20.
- [131] J. Hastad and M. Goldmann, "On the power of small-depth threshold circuits," *computational complexity*, vol. 1, pp. 113-129, Jun. 1991, doi:10.1007/BF01272517.
- [132] A. Hajnal, W. Maass, P. Pudlák, M. Szegedy and G. Turán, "Threshold circuits of bounded depth," *Journal of Computer and System Sciences*, vol. 46, no. 2, pp. 129-154, Apr. 1993, doi:10.1016/0022-0000(93)90001-D.
- [133] W. Maass, "Bounds for the computational power and learning complexity of analog neural nets," *In Proc. of the 25th ACM Symp. Theory of Computing*, vol. 26, no. 3, Jun. 1997, doi:10.1145/167088.167193.
- [134] W. Maass, G. Schnitger and E. Sontag, "A Comparison of the Computational Power of Sigmoid and Boolean Threshold Circuits," in *Theoretical Advances in Neural Computation and Learning*, V. Roychowdhury, K. Siu and A. Orlitsky, A, Eds. Boston, MA: Springer, 1994, pp. 127-151.

- [135] G. Montúfar, “Universal Approximation Depth and Errors of Narrow Belief Networks with Discrete Units,” *Neural Computation*, vol. 26, no. 7, pp. 1386-1407, Jul. 2014, doi:10.1162/NECO_a_00601.
- [136] Y. Bengio, P. Lamblin, D. Popovici and H. Larochelle, “Greedy layer-wise training of deep networks,” in *Proceedings of the 19th International Conference on Neural Information Processing Systems (NIPS'06)*, pp. 153–160.
- [137] D. Erhan, P. Manzagol, Y. Bengio, S. Bengio and P. Vincent, “The Difficulty of Training Deep Architectures and the Effect of Unsupervised Pre-Training,” *Journal of Machine Learning Research - Proceedings Track*, vol. 5, pp. 153-160, Jan. 2009. [Online]. Available: Academic OneFile, <http://proceedings.mlr.press/v5/erhan09a/erhan09a.pdf>. Accessed on: April 15, 2009.
- [138] Y. Bengio, J. Louradour, R. Collobert and J. Weston, “Curriculum learning,” in *Proceedings of the 26th annual international conference on machine learning*, pp. 41-48.
- [139] I. Guyon, G. Dror, V. Lemaire, G. Taylor and D. Aha, “Unsupervised and transfer learning challenge,” in *2011 International Joint Conference on Neural Networks*, pp. 793-800.
- [140] D. Cireşan, U. Meier, J. Masci and J. Schmidhuber, “Multi-column deep neural network for traffic sign classification,” *Neural Networks*, vol. 32, pp. 333-338, Aug. 2012, doi:10.1016/j.neunet.2012.02.023.
- [141] A. Krizhevsky, I. Sutskever and G. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. of the ACM*, vol. 60, no. 6, pp. 84-90, Jun. 2017, doi:10.1145/3065386.
- [142] P. Sermanet, K. Kavukcuoglu, S. Chintala and Y. Lecun, “Pedestrian Detection with Unsupervised Multi-stage Feature Learning,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3626-3633.
- [143] J. Li, Y. Liu, J. Liu, R. Song, W. Liu, K. Han and Q. Du, “Feature Guide Network with Context Aggregation Pyramid for Remote Sensing Image Segmentation,” *IEEE*

Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 15, pp. 9900-9912, Nov. 2022, 10.1109/JSTARS.2022.3221860.

[144] C. Couprie, C. Farabet, L. Najman and Y. LeCun, “Indoor semantic segmentation using depth information,” in *International Conference on Learning Representations (ICLR2013)*, pp. 1-8.

[145] S. Kahou, C. Pal, X. Bouthillier, P. Froumenty, Ç. Gülçehre, R. Memisevic, P. Vincent, A. Courville, Y. Bengio, R. Ferrari, M. Mirza, S. Jean, P. Carrier, Y. Dauphin, N. Boulanger-Lewandowski, A. Aggarwal, J. Zumer, P. Lamblin, J. Raymond, G. Desjardins, R. Pascanu, D. Warde-Farley, A. Torabi, A. Sharma, E. Bengio, M. Côté, K. Konda and Z. Wu, “Combining modality specific deep neural networks for emotion recognition in video,” in *Proceedings of the 15th ACM on International conference on multimodal interaction (ICMI '13)*, pp. 543-550.

[146] I. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud and V. Shet, “Multi-digit number recognition from Street View imagery using deep convolutional neural networks,” in *International Conference on Learning Representations*, pp. 1-13.

[147] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, “Going deeper with convolutions,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-9

[148] Convolution Neural Networks cheatsheet. Stanford University, 2022. [Online]. Available: <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks>. Accessed on: 2022.

[149] Conv3d. PyTorch Contributors, 2023. [Online]. Available: <https://pytorch.org/docs/stable/generated/torch.nn.Conv3d.html>. Accessed on: 2023.

[150] Techniques to improve the performance of a neural network. Analytics Vidhya, 2019. [Online]. Available: <https://medium.com/analytics-vidhya/improving-performance-of-a-neural-network-22776fde2089>. Accessed on: Dec. 18, 2019.

- [151] Improving the Performance of a Neural Network. Medium.com, 2018. [Online]. Available: <https://towardsdatascience.com/how-to-increase-the-accuracy-of-a-neural-network-9f5d1c6f407d>. Accessed on: May 17, 2018.
- [152] Correlation Coefficient: Simple Definition, Formula, Easy Steps. StatisticsHowTo.com, 2021. [Online]. Available: <https://www.statisticshowto.com/probability-and-statistics/correlation-coefficient-formula/>. Accessed on: May 26, 2022.
- [153] M Zozyuk and O. Yurikov, “Convolutional Neural Network to Predict the Penetration Coefficient of Metamaterials Based on Their Structure and Composition,” *Microsystems, Electronics and Acoustics*, vol. 28, no. 1, pp. 1-10, Jul. 2023, doi:10.20535/2523-4455.mea.271444.
- [154] S. Dovgyi, O. Yurikov and M. Zozyuk, “On One Statistical Model of Error Rate in the Stream of Packet Data Transmission through Communication Channels,” *Cybern Syst Anal*, vol. 56, no. 5, pp. 739–744, Oct. 2020. doi:10.1007/s10559-020-00294-x.
- [155] J. Parsons, C. Burrows, J. Sambles and W. Barnes, “A comparison of techniques used to simulate the scattering of electromagnetic radiation by metallic nanostructures,” *Journal of Modern Optics*, vol. 57, no. 5, pp. 356-365, Mar. 2012, doi:10.1080/09500341003628702.

ДОДАТКИ

ДОДАТОК А. Список публікацій здобувача за темою дисертації:

Праці, в яких опубліковано основні результати дисертації:

[1] M. O. Zozyuk, D. V. Koroliouk, P. I. Krysenko, A. I. Yurikov and Y. I. Yakymenko, “Prediction of characteristics using a convolutional neural network based on experimental data on the structure and composition of metamaterials,” *STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING*, vol. 11, no. 3, pp. 777–787, Jun. 2023, doi:10.19139/soic-2310-5070-1707. *(Стаття у закордонному виданні, проіндексованому у базі даних Web of Science Core Collection та Scopus та віднесеному до 3 квартилю (Q3) відповідно до класифікації SCImago Journal and Country Rank).*

Особистий внесок здобувача: здобувачем підготовлено текст статті, розроблено алгоритм прогнозування коефіцієнту пропускання метаматеріалів на основі структури та складу метаматеріалів, розроблено програмну частину з підготовки даних для проведення експерименту, проведено аналіз результатів прогнозування коефіцієнту пропускання метаматеріалів на основі структури та їх складу та проведено обговорення результатів із співавторами.

[2] М.О. Зозюк та О.І. Юріков. “Використання згорткової нейронної мережі для прогнозування коефіцієнту пропускання метаматеріалів від їх структури та складу,” *Мікросистеми, Електроніка та Акустика.*, vol. 28, no. 1, pp. 271444.1-271444.10, Jul. 2023, doi:10.20535/2523-4455.me.271444. *(Стаття у періодичному науковому фаховому виданні категорії Б).*

Особистий внесок здобувача: здобувачем підготовлено текст статті, підготовлено гіпотези про дані для нейромереж, проаналізовано результати прогнозування коефіцієнту пропускання метаматеріалів та проведено їх обговорення із співавторами.

[3] M. O. Zoziuk, O. I. Yurikov, D. V. Koroliouk and Y. I. Yakymenko, “The Principle of Creating Quasiperiodic Surfaces under the Action of a Vibrating Dielectric Matrix,” *Microsystems, Electronics and Acoustics*, vol. 25, no. 1, 2020, pp. 5-10, Dec. 2020. doi:10.20535/2523-4455.mea.202632. **(Стаття у періодичному науковому фаховому виданні категорії Б).**

Особистий внесок здобувача: здобувачем підготовлено текст статті, проведено моделювання фігур Хладні за допомогою приведених формул, проведено аналіз результатів моделювання та проведено їх обговорення із співавторами.

[4] P. I. Krysenko, M. O. Zoziuk, O. I. Yurikov, D. V. Koroliuk and Yu. I. Yakymenko, “Chladni Figures Simulation on a Rectangular Plate,” *Microsystems, Electronics and Acoustics*, vol. 26, no. 1, 2021, pp. 241698.1-241698.6, Dec. 2021. doi:10.20535/2523-4455.mea.241698. **(Стаття у періодичному науковому фаховому виданні категорії Б).**

Особистий внесок здобувача: здобувачем підготовлено текст статті, проведено моделювання фігур Хладні та графіків залежності коефіцієнту зміщення від частоти для хвиль різної природи за допомогою приведених формул, проведено аналіз результатів моделювання та проведено їх обговорення із співавторами.

[5] S.O. Dovgyi, O.I. Yurikov and M.O. Zozyuk, “On One Statistical Model of Error Rate in the Stream of Packet Data Transmission through Communication Channels,” *Cybern Syst Anal*, vol. 56, no. 5, pp. 739–744, Oct. 2020. doi:10.1007/s10559-020-00294-x. **(Стаття у закордонному виданні, проіндексованому у базі даних Web of Science Core Collection та Scopus та віднесеному до 2 квартилю (Q2) відповідно до класифікації SCImago Journal and Country Rank).**

Особистий внесок здобувача: здобувач приймав участь в підготовці тексту статті, проаналізовано результати моделювання та проведено їх обговорення із співавторами.

Праці, які засвідчують апробацію матеріалів дисертації:

[1] M. O. Zozyuk, D. V. Koroliouk, V. O. Moskaliuk, A. I. Yurikov and Y. I. Yakymenko, “Creation of quasiperiodic surfaces under the action of vibrating dielectric matrices,” presented at the *2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO)*, Kyiv, 2020. Available: doi:10.1109/ELNANO50318.2020.9088821. Accessed on: April 22, 2020.

Особистий внесок здобувача: здобувачем підготовлено текст статті, проведено моделювання фігур Хладні за допомогою приведених формул, проведено аналіз результатів моделювання та проведено обговорення отриманих результатів із співавторами.

[2] М.О. Зозюк, Д.В. Королюк, П.І. Крисенко, О.І. Юріков та Ю.І. Якименко, “Використання нейронних мереж для прогнозування властивостей метаматеріалів,” представлено на *IX Міжнародній науковій конференції імені І. І. Ляшка «Обчислювальна та прикладна математика»*, Київ, Україна, 2022.

Особистий внесок здобувача: здобувачем підготовлено текст виступу, проведено моделювання характеристик метаматеріалів на основі їх структури, проведено аналіз результатів моделювання та проведено обговорення отриманих результатів із співавторами.

[3] М.О. Зозюк та О.І. Юріков, “Підготовка даних для використання в системах прогнозування властивостей метаматеріалів для нейронної згорткової мережі,” представлено на *XXI Міжнародній науково – практичній конференції – Інформаційно-комунікаційні технології та сталий розвиток*, Київ, Україна, 2022.

Особистий внесок здобувача: здобувачем підготовлено текст тез, проведений набір даних для навчання глибокої нейронної мережі, перевірена якість даних та проведено обговорення отриманих результатів із співавторами.