

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Кваліфікаційна наукова
праця на правах рукопису

НОВОСЕЛЬЦЕВ ІГОР ВАЛЕРІЙОВИЧ

УДК 004.93'1: 004.032.26

МЕТОДИ ТА ЗАСОБИ РОЗПІЗНАВАННЯ ЗМІН ВЛАСТИВОСТЕЙ ОБ'ЄКТА
ЗА ЗОБРАЖЕННЯМ НА ОСНОВІ ШТУЧНИХ
НЕЙРОННИХ МЕРЕЖ

Спеціальність 05.13.23 – системи та засоби штучного інтелекту

Подається на здобуття наукового ступеня кандидата технічних наук

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

_____ І. В. Новосельцев

Науковий керівник:

Аксак Наталія Георгіївна, кандидат технічних наук, доцент

Харків – 2019

АНОТАЦІЯ

Новосельцев І.В. Методи та засоби розпізнавання змін властивостей об'єкта за зображенням на основі штучних нейронних мереж. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.23 – системи та засоби штучного інтелекту. – Харківський національний університет радіоелектроніки, Міністерство освіти і науки України, Харків, 2019.

У дисертаційній роботі на основі отриманих теоретичних і експериментальних досліджень вирішена актуальна науково-практична задача побудови нейромережових методів розпізнавання і класифікації зміни властивостей зображень, які на відміну від існуючих, зменшують помилку розпізнавання і збільшують точність класифікації зображень в умовах апіорної і поточної невизначеності і наявності завад.

Об'єктом дослідження є процеси розпізнавання змін властивостей об'єктів за зображенням. Предметом дослідження є методи та моделі розпізнавання змін властивостей об'єкта за зображенням на основі штучних нейронних мереж.

Дисертаційне дослідження базується на системному аналізі результатів сучасних теоретичних і прикладних розробок вітчизняних і зарубіжних учених в ІТ галузі. Для вирішення поставлених завдань використано: теорію обчислювального інтелекту, а саме, методи теорії штучних нейронних мереж, яка дозволила синтезувати нейромережові моделі та отримати процедури їх навчання; методи теорії оптимальності, за допомогою якої були синтезовані швидкодіючі процедури навчання; методи теорії розпізнавання образів, на основі яких були синтезовані моделі аналізованих об'єктів, що настроюються; методи імітаційного моделювання, що дозволили підтвердити ефективність отриманих результатів та розробити рекомендації щодо їх практичного використання. Експериментальні дослідження проводилися в лабораторних умовах і на реальних об'єктах.

У дисертаційній роботі проведено аналіз стану проблем розпізнавання, класифікації, кластеризації та обробки зображень. На основі системного підходу проаналізовано існуючі підходи до застосування нейронних мереж для рішення цих завдань і описані основні типи ШНМ, сформульована мета і задачі дослідження.

Пропонується комплекс взаємопов'язаних завдань, вирішення яких полягає у розробці методів та засобів розпізнавання змін властивостей об'єкта за зображенням на основі штучних нейронних мереж. Запропоновано:

- метод контролю зміни розмірів об'єкта, сутність якого полягає в використанні допомогою еталона та використанні властивостей перетворення подібності об'єктів, що дозволяє підвищити точність вимірювань.

- процедуру навчання БШП, що являє собою матричні варіанти процедури Качмажа (Уідроу-Гоффа), запропонована її модифікація, яка містить зону нечутливості, і розглянуті процедури налаштування цієї зони, що забезпечило отримання робастних оцінок шуканих параметрів мережі.

Удосконалено метод адаптації багат шарового персептрона на багатопроцесорні системи, що дозволило отримати оцінку потенційно можливого виграшу в продуктивності в залежності від використовуваних ресурсів, а також розмірності досліджуваного зображення.

Набув подальшого розвитку нейромережевий метод класифікації зображень на основі згортальних нейронних мереж з використанням в різних шарах різних функцій активації і робастного навчання параметрів мережі.

Набув подальшого розвитку нейромережевий метод розпізнавання змін параметрів зображень на прикладі меланом з використанням мереж PNN і CNN, на основі яких розроблені нейромережевий метод контролю зміни розмірів шкірного утворення і розпізнавання неоднорідності забарвлення шкірних змін, що дозволяють збільшити точність контролю зміни розмірів спостережуваних пігментних новоутворень шкіри для ранньої діагностики меланом і підвищити ефективність класифікації поліхромії пігментних новоутворень шкіри.

Практичне значення отриманих результатів полягає у розробці:

- методу контролю розмірів об'єкта за допомогою еталона, який може бути використаний для дистанційного визначення зміни розмірів об'єкта, що спостерігається (патент на винахід № 84380 А по МПК (2006) G01С 11/00 від 10.10.2008. Бюл. №19.);

структури та програмної реалізації нейромережевої системи визначення неоднорідності забарвлення складається з ансамблю PNN і CNN, що дозволяє підвищити точність діагностування, зменшити час навчання, знизити ризик надмірного навчання та полегшує лікареві постановку діагноза.

Запропоновані методи та засоби розпізнавання змін властивостей об'єкта за зображенням доведені до рівня практичної реалізації і дозволяють:

- виявляти важке захворювання шкіри на ранній стадії завдяки контролю зміни розмірів шкірного новоутворення, на підставі чого можна призначити й провести своєчасне лікування;

- накопичувати досвід для більш точної діагностики хворих за рахунок використання штучних нейронних мереж, що є актуальним для даної предметної області;

- використати розроблені методи в телемедицині, що дозволить своєчасно контролювати зміни дефектів шкіри.

Результати дисертаційної роботи впроваджені: в інституті дерматології та венерології АМН України (акт про впровадження від 05.11.2007); у практичну діяльність товариства з обмеженою відповідальністю «Сана-мед» (акт про впровадження від 01.07.2009); у практичну діяльність товариства з обмеженою відповідальністю «Іпра-софт» (акт про впровадження від 17.05.2018).

Дисертаційна робота виконана відповідно до плану науково-дослідних робіт Харківського національного університету радіоелектроніки в рамках таких держбюджетних тем: НДР «Синтез методів обробки інформації в умовах невизначеності на основі самонавчання і м'яких обчислень»; розділ «Гібридні

моделі, що самонавчаються, в задачах обробки нечіткої інформації» (№ ДР 0107U003028); НДР «Еволюційні гібридні системи обчислювального інтелекту зі змінною структурою для інтелектуального аналізу даних», розділ «Еволюційні гібридні методи і моделі інтелектуальної обробки інформації зі змінною структурою в умовах невизначеності» (№ ДР 0110U000458); НДР «Нейро-фаззи системи для поточної кластеризації і класифікації послідовностей даних за умов їх викривленості відсутніми та аномальними спостереженнями», розділ «Адаптивні методи та моделі класифікації даних і прогнозування часових рядів за умов їх викривленості відсутніми та аномальними спостереженнями на основі штучних імунних систем» (№ ДР 0113U000361)., затвердженими Міністерством освіти і науки України. Автор був одним з виконавців робіт за даними темами.

Результати дисертаційної роботи впроваджені: в інституті дерматології та венерології АМН України (акт про впровадження від 05.11.2007); у практичну діяльність товариства з обмеженою відповідальністю «Сана-мед» (акт про впровадження від 01.07.2009); у практичну діяльність товариства з обмеженою відповідальністю «Іпра-софт» (акт про впровадження від 17.05.2018).

Матеріали дисертації повною мірою викладені у 25 публікаціях, з них – 12 у фахових періодичних виданнях України з технічних наук; 7 статей включено у міжнародні наукометричні бази, 13 тез доповідей у матеріалах міжнародних наукових конференцій, одну з яких включено у базу Scopus.

Ключові слова: зображення, класифікація, критерій втрат, активаційна функція, навчання мережі, робастність, згорткова нейронна мережа.

СПИСОК ОПУБЛІКОВАНИХ РОБІТ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

в яких опубліковані основні наукові результати дисертації:

1. Аксак Н. Г. Распознавание изменения размера и цвета изображения на основе сверточной нейронной сети / Н.Г. Аксак, А.А. Бессонов, И.В. Новосельцев, О.Г. Руденко // Бионика интеллекта: Научн.-техн. журнал. – 2018. – №2 (91). – С.114-

119. (Входить до міжнародних наукометричних баз Google Scholar, Index Copernicus, BASE, Research Gate).

2. Новосельцев И. В. Система первичной диагностики меланомы кожи / И.В. Новосельцев, Н.Г. Аксак // Бионика интеллекта: Научн.-техн. журнал. – 2010. – №3(74). – С.94-98. (Входить до міжнародної наукометричної бази Google Scholar).

3. Новосельцев И.В. Применение многопроцессорных систем для реализации многослойного персептрона / И.В. Новосельцев, Н.Г. Аксак // Радиоэлектроника. Информатика. Управление. – №1(19). – 2008. – С.108-110. (Входить до міжнародних наукометричних баз РІНЦ, Google Scholar).

4. Новосельцев И.В. Вероятностные нейронные сети для решения задач медицинской диагностики / И.В. Новосельцев, Н.Г. Аксак, О.Ю.Барковская // Бионика интеллекта: Научн.-техн. журнал. – 2006. – №2(65). – С. 62-65.

5. Барковская О.Ю. Сравнительный анализ алгоритмов бинаризации полутонового изображения / О.Ю.Барковская, Н.Г. Аксак, И.В. Новосельцев // Автоматизированные системы управления и приборы автоматики, 2005. – Вып. 133. – С. 40-46.

6. Аксак Н. Г. Система предоставления медицинских услуг на основе архитектуры cloud-fog-dew / Н.Г. Аксак, И.В. Новосельцев, Д.Н. Росинский // Наука та виробництво: міжвуз. темат. зб. наук. пр. Вип. 19 / ДВНЗ «ПДТУ». –Маріуполь, ПДТУ, 2018. – С. 150-159.

7. Новосельцев І.В. Система віддаленого нейромережного діагностування меланоми / І.В. Новосельцев, Н.Г. Аксак // Збірник наукових праць, ХНУРЕ.– Харків: «Друкарня Мадрид», 2018. – 116-118.

8. Аксак Н.Г. Мультиагентная модель адаптации медицинского Интернет-ресурса на основе информационного портрета пользователя /Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Системы обработки інформації. Збірник наукових праць. – Х.: ХУПС, 2013. – Вип.2(109). – С. 242-246. (Входить до міжнародних наукометричних баз Index Copernicus, BASE).

9. Новоосельцев И.В. Показатели эффективности нейросетевой системы медицинской диагностики / И.В. Новосельцев, Н.Г. Аксак // Системи управління, навігації та зв'язку. Збірник наукових праць. Випуск 4(16), Київ, 2010. - С.142-149.

10. Аксак Н.Г. Адаптация нейросетевых моделей на высокопроизводительные системы с общей памятью / Н.Г. Аксак, И.В. Новосельцев, А.С. Солдатов // Системні технології. Збірник наукових праць. – Випуск 3 (56).- Том 2.-Дніпропетровськ, 2008.- С. 117-122.

11. Новосельцев И.В. Гибридные нейронные сети для распознавания образов / И.В. Новосельцев, Н.Г. Аксак, О.Ю. Барковская // Системи обробки інформації. Збірник наукових праць.-2007.- Вып. 3(61).-С.60-64. (Входить до міжнародних наукометричних баз BASE, WorldCat).

12. Барковская О.Ю. Обобщенная модель распознавания антропоморфных объектов / О.Ю. Барковская, И.В. Новосельцев, Н.Г. Аксак // Системні технології. Збірник наукових праць.- Випуск 6 (53).- Дніпропетровськ, 2007.- С. 11-16.

Публікації, які засвідчують апробацію матеріалів дисертації:

13. Axak N. Cloud-fog-dew Architecture for Personalized Service-oriented Systems / N. Axak, D.Rosinskiy, O. Barkovska, I. Novoseltsev // The 9th IEEE International Conference on Dependable Systems, Services and Technologies, DESSERT'2018, Kyiv, Ukraine – 2018. – P.80 -84. (Входить до міжнародної наукометричної бази **Scopus**, РІНЦ).

14. Аксак Н.Г. Спосіб контролю зміни розмірів об'єкта за допомогою шаблону / Н.Г. Аксак, М.В. Кушнар'ов, І.В. Новосельцев, А.Ю. Тихун //Патент на винахід № 84380 А по МПК (2006) G01C 11/00 від 10.10.2008. Бюл. №19.

15. Аксак Н.Г. Система интеллектуального анализа медицинских данных / Н.Г. Аксак, И.В. Новосельцев, Д.Н. Росинский, В.О. Лебедев // Комп'ютерні та інформаційні системи і технології: Матеріали другої міжнародної науково-технічної конференції. Харків. – 2018. – С.128-129.

16. Аксак Н.Г. Агентно-ориентированный медицинский Интернет-ресурс / Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Вычислительный интеллект (результаты, проблемы, перспективы): Материалы 2-й Международной научно-технической конференции. – Черкассы: Маклаут, 2013. – С.321.

17. Аксак Н.Г. Анализ способов ускорения нейроалгоритма на основе технологий параллельного программирования / Н.Г. Аксак, А.Ю. Тыхун, И.В. Новосельцев // Матеріали Х Міжнародної науково-технічної конференції "Системний аналіз та інформаційні технології". – К., 2008. – С.284.

18. Аксак Н.Г., Модели производительности синтеза нейроалгоритма на высокопроизводительных архитектурах / Н.Г. Аксак, А.Ю. Тыхун, И.В. Новосельцев // Труды 8-й Международной конференции. Высокопроизводительные параллельные вычисления на кластерных системах (HPC-2008), изд-во КГТУ, Казань. – 2008. – С.199-202.

Публікації, які додатково відображають наукові результати дисертації:

19. Аксак Н.Г. Распознавание образов на основе нейронных сетей с использованием технологии MPI / Н.Г. Аксак, Р.С. Верчиков, О.Р. Лола, И.В. Новосельцев, С.А. Олищук // Высокопроизводительные параллельные вычисления на кластерных системах. Материалы шестого Международного научно-практического семинара. Том 1. Санкт-Петербург, 2007. –С.21-25.

20. Новосельцев, И.В. Модель обработки больших данных для систем здравоохранения / И.В. Новосельцев, А.Н. Тарасенко //Материалы 22-го международного молодежного форума «Радиоэлектроника и молодежь в 21 столетии». – 17-19 апреля 2018. – Харьков. – С. 57-58.

21. Аксак Н.Г. Нейросетевая медицинская система принятия решений / Н.Г. Аксак, И.В. Новосельцев, А.П. Лучко // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: Матеріали восьмої міжнародної науково-технічної конференції Полтава–Баку – Харків - Жиліна. – 2018.- С.13.

22. Новосельцев И.В. Реализация многослойного персептрона с помощью стандарта OpenMP / И.В. Новосельцев, Н.Г. Аксак // 8-я международная научно-практическая конференция "Современные информационные и электронные технологии": Сб. трудов конференции - Одесса, 2007.-С.79.

23. Новосельцев И.В. Применение нейронных сетей для построения медицинской экспертной системы / И.В. Новосельцев, Н.Г. Аксак // Международная научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов по материалам Междун. Научн. конференции.- Харьков: ХНУРЭ, 2003,- С.341-342.

24. Новосельцев И.В. Применение нейронной сети при фильтрации спам сообщений / И.В. Новосельцев, Н.Г. Аксак //10-я Юбилейная международная научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов.Ч.2. - Харьков: ХНУРЭ, 2004.- С.295-296.

25.И.В.Новосельцев, Н.Г. Аксак. Использование параллельных технологий для распознавания образов. // Материалы 11-го юбилейного международного форума «Радиоэлектроника и молодежь в XXI веке». – Харьков: ХНУРЕ, 2007. – с.306.

ABSTRACT

Novoseltsev I.V. Methods and means for recognizing the changes of the object properties through images based on artificial neural networks.– Qualifying scientific work as a manuscripts.

Thesis for the degree of candidate of technical sciences in specialty 05.13.23 – the systems and means of artificial intelligence. - Kharkiv National University of Radio Electronics, Ministry of Education and Science of Ukraine, Kharkiv, 2019

In the dissertation work on the basis of the received theoretical and experimental researches the actual scientific and practical task of constructing neural network methods

of recognition and classification of changes in the properties of images is solved, which, in contrast to existing, reduce recognition error and increase the accuracy of the classification of images in conditions of a priori and current uncertainty and the presence of obstacles.

The object of the study is the processes of recognizing changes in the properties of objects in the image. The subject of the study is the methods and models for recognizing changes in the properties of an object in an image based on artificial neural networks.

The dissertation is based on a systematic analysis of the results of modern theoretical and applied developments of domestic and foreign scientists in the IT industry. In order to solve the problems, we used: the theory of computational intelligence, namely methods of the theory of artificial neural networks, which allowed to synthesize neural network models and obtain procedures for their training; methods of the theory of optimality, through which were synthesized high-speed training procedures; methods of the theory of pattern recognition, on the basis of which the models of the analyzed customizable objects were synthesized; Methods of simulation, which allowed to confirm the effectiveness of the results obtained and develop recommendations for their practical use. Experimental studies were conducted in laboratory conditions and on real objects.

In the dissertation work the analysis of the status of problems of recognition, classification, clustering and image processing is carried out. On the basis of the systematic approach, existing approaches to the application of neural networks for the solution of these problems are analyzed and the main types of SNM are described, the purpose and objectives of the research are formulated.

A set of interrelated tasks is proposed, the solution of which is to develop methods and means for recognizing changes in the properties of an object in an image based on artificial neural networks. Proposed:

- a method for controlling the change in the size of an object, the essence of which is to use the standard using the properties of converting the similarity of objects, which increases the accuracy of measurements.

- training procedure for BHP, which is the matrix variants of the Kaczmarz's procedure (Widrow-Hoff), its modification is proposed, which contains a zone of insensitivity, and the procedures for adjusting this zone were considered, which provided robust estimations of the required network parameters.

The method of adaptation of multilayer perceptron to multiprocessor systems has been improved, which allowed obtaining an estimation of the potential gain in productivity depending on the resources used, as well as the dimension of the investigated image.

He further developed the neural network method of image classification on the basis of cryptographic neural networks using various functions of activation and robust training of network parameters in different layers.

The neural network method of recognizing the change of image parameters on the example of melanoma using the networks PNN and CNN, on the basis of which the neural network method of controlling the change in the size of skin formation and recognition of the heterogeneity of coloring of skin changes, allows to increase the accuracy of the control of the change in the size of the observed pigmentary neoplasms of the skin for the early to diagnose melanoma and to improve the classification of pigment skin pigmentation polychromy.

The practical value of the results obtained is to develop:

- a method for controlling the size of an object using a standard that can be used to remotely determine the change in the size of an object being observed (patent for invention No. 84380 A according to the IPC (2006) G01C 11/00 of 10.10.2008. 19);
- the structure and program implementation of the neural network system for determining the color heterogeneity consists of the PNN and CNN ensemble, which allows to increase the accuracy of diagnosis, reduce the time of training, reduce the risk of excessive training and facilitate the diagnosis of a doctor.

Proposed methods and means for recognizing changes in the properties of an object by image are brought to the level of practical implementation and allow:

- to detect severe skin disease at an early stage by controlling changes in the size of the skin, based on which you can appoint and conduct timely treatment;
- to accumulate experience for more accurate diagnostics of patients by using artificial neural networks, which is relevant for a given subject area;
- to use the developed methods in telemedicine, which will allow timely control of changes in skin defects.

The results of the dissertation work are introduced: at the Institute of Dermatology and Venereology of the Academy of Medical Sciences of Ukraine (Act on the implementation of November 5, 2007); in the practical activity of the limited liability company "Sana-honey" (implementation act dated 01.07.2009); in the practical activity of the Limited Liability Company "Ipra-soft" (implementation act dated 05/17/2018).

The thesis is executed in accordance with the plan of research works of the Kharkiv National University of Radio Electronics in the framework of the following state budget topics: Research project "Synthesis of methods of information processing in conditions of uncertainty on the basis of self-study and soft computing"; The section "Self-learning hybrid models in the processing of fuzzy information" (No. DR 0107U003028); GDR "Evolutionary hybrid systems of computing intelligence with variable structure for the intellectual data analysis", section "Evolutionary hybrid methods and models of intelligent processing of information with variable structure under uncertainty" (No. DR 0110U000458); Research "Neurophase of the system for the current clusterization and classification of data sequences in the absence of their distortion and abnormal observations", section "Adaptive methods and models of data classification and prediction of time series in the absence of their distortion and abnormal observations based on artificial immune systems" (No. DR 0113U000361), Approved by the Ministry of Education and Science of Ukraine. The author was one of the performers of works on these topics.

The results of the dissertation work are introduced: at the Institute of Dermatology and Venereology of the Academy of Medical Sciences of Ukraine (Act on the

implementation of November 5, 2007); in the practical activity of the limited liability company "Sana-honey" (implementation act dated 01.07.2009); in the practical activity of the Limited Liability Company "Ipra-soft" (implementation act dated 05/17/2018).

The materials of the thesis are fully described in 25 publications, of which 12 are in professional periodicals of Ukraine on technical sciences; 7 articles are included in the international science and technology base, 13 abstracts in the materials of the international scientific conferences, one of which is included in the base of Scopus.

Keywords: image, classification, loss criterion, activation function, network learning, robustness, convolutional neural network.

LIST OF PUBLICATIONS

The list of publications, which reflect the main scientific results of the thesis:

1. Аксак Н. Г. Распознавание изменения размера и цвета изображения на основе сверточной нейронной сети / Н.Г. Аксак, А.А. Бессонов, И.В. Новосельцев, О.Г. Руденко // Бионика интеллекта: Научн.-техн. журнал. – 2018. – №2 (91). – С.114-119. (Входить до міжнародних наукометричних баз Google Scholar, Index Copernicus, BASE, Research Gate).

2. Новосельцев И. В. Система первичной диагностики меланомы кожи / И.В. Новосельцев, Н.Г. Аксак // Бионика интеллекта: Научн.-техн. журнал. – 2010. – №3(74). – С.94-98. (Входить до міжнародної наукометричної бази Google Scholar).

3. Новосельцев И.В. Применение многопроцессорных систем для реализации многослойного персептрона / И.В. Новосельцев, Н.Г. Аксак // Радиоэлектроника. Информатика. Управление. – №1(19). – 2008. – С.108-110. (Входить до міжнародних наукометричних баз РІНЦ, Google Scholar).

4. Новосельцев И.В. Вероятностные нейронные сети для решения задач медицинской диагностики / И.В. Новосельцев, Н.Г. Аксак, О.Ю.Барковская // Бионика интеллекта: Научн.-техн. журнал. – 2006. – №2(65). – С. 62-65.

5. Барковская О.Ю. Сравнительный анализ алгоритмов бинаризации полутонового изображения / О.Ю.Барковская, Н.Г. Аксак, И.В. Новосельцев // Автоматизированные системы управления и приборы автоматики, 2005. – Вып. 133. – С. 40-46.

6. Аксак Н. Г. Система предоставления медицинских услуг на основе архитектуры cloud-fog-dew / Н.Г. Аксак, И.В. Новосельцев, Д.Н. Росинский // Наука та виробництво: міжвуз. темат. зб. наук. пр. Вип. 19 / ДВНЗ «ПДТУ». – Маріуполь, ПДТУ, 2018. – С. 150-159.

7. Новосельцев І.В. Система віддаленого нейромережного діагностування меланоми / І.В. Новосельцев, Н.Г. Аксак // Збірник наукових праць, ХНУРЕ.– Харків: «Друкарня Мадрид», 2018. – 116-118.

8. Аксак Н.Г. Мультиагентная модель адаптации медицинского Интернет-ресурса на основе информационного портрета пользователя /Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Системи обробки інформації. Збірник наукових праць. – Х.: ХУПС, 2013. – Вип.2(109). – С. 242-246. (Входить до міжнародних наукометричних баз Index Copernicus, BASE).

9. Новосельцев И.В. Показатели эффективности нейросетевой системы медицинской диагностики / И.В. Новосельцев, Н.Г. Аксак // Системи управління, навігації та зв'язку. Збірник наукових праць. Випуск 4(16), Київ, 2010. - С.142-149.

10. Аксак Н.Г. Адаптация нейросетевых моделей на высокопроизводительные системы с общей памятью / Н.Г. Аксак, И.В. Новосельцев, А.С. Солдатов // Системні технології. Збірник наукових праць. – Випуск 3 (56).- Том 2.-Дніпропетровськ, 2008.- С. 117-122.

11. Новосельцев И.В. Гибридные нейронные сети для распознавания образов / И.В. Новосельцев, Н.Г. Аксак, О.Ю. Барковская // Системи обробки інформації. Збірник наукових праць.-2007.- Вип. 3(61).-С.60-64. (Входить до міжнародних наукометричних баз BASE, WorldCat).

12. Барковская О.Ю. Обобщенная модель распознавания антропоморфных объектов / О.Ю. Барковская, И.В. Новосельцев, Н.Г. Аксак // Системні технології. Збірник наукових праць. - Випуск 6 (53).- Дніпропетровськ, 2007.- С. 11-16.

Results that confirm the approbation of the thesis:

13. AxakN. Cloud-fog-dew Architecture for Personalized Service-oriented Systems / N. Axak, D.Rosinskiy, O. Barkovska, I. Novoseltsev // The 9th IEEE International Conference on Dependable Systems, Services and Technologies, DESSERT'2018, Kyiv, Ukraine – 2018. – P.80 -84. (Входить до міжнародної наукометричної бази **Scopus**, РІНЦ).

14. Аксак Н.Г. Спосіб контролю зміни розмірів об'єкта за допомогою шаблону / Н.Г. Аксак, М.В. Кушнар'ов, І.В. Новосельцев, А.Ю. Тихун // Патент на винахід № 84380 А по МПК (2006) G01C 11/00 від 10.10.2008. Бюл. №19.

15. Аксак Н.Г. Система интеллектуального анализа медицинских данных/ Н.Г. Аксак, И.В. Новосельцев, Д.Н. Росинский, В.О. Лебедев // Комп'ютерні та інформаційні системи і технології: Матеріали другої міжнародної науково-технічної конференції. Харків. – 2018. – С.128-129.

16. Аксак Н.Г. Агентно-ориентированный медицинский Интернет-ресурс / Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Вычислительный интеллект (результаты, проблемы, перспективы): Материалы 2-й Международной научно-технической конференции. – Черкассы: Маклаут, 2013. – С.321.

17. Аксак Н.Г. Анализ способов ускорения нейроалгоритма на основе технологий параллельного программирования / Н.Г. Аксак, А.Ю. Тихун, И.В. Новосельцев // Матеріали X Міжнародної науково-технічної конференції "Системний аналіз та інформаційні технології". – К., 2008. – С.284.

18. Аксак Н.Г., Модели производительности синтеза нейроалгоритма на высокопроизводительных архитектурах / Н.Г. Аксак, А.Ю. Тихун, І.В. Новосельцев // Труды 8-й Международной конференции. Высокопроизводительные параллельные

вычисления на кластерных системах (НРС-2008), изд-во КГТУ, Казань. – 2008. – С.199-202.

Publications that additionally reflect the scientific results of the thesis:

19. Аксак Н.Г. Распознавание образов на основе нейронных сетей с использованием технологии MPI / Н.Г. Аксак, Р.С. Верчиков, О.Р. Лола, И.В. Новосельцев, С.А. Олищук // Высокопроизводительные параллельные вычисления на кластерных системах. Материалы шестого Международного научно-практического семинара. Том 1. Санкт-Петербург, 2007. –С.21-25.

20. Новосельцев, И.В. Модель обработки больших данных для систем здравоохранения / И.В. Новосельцев, А.Н. Тарасенко //Материалы 22-го международного молодежного форума «Радиоэлектроника и молодежь в 21 столетии». – 17-19 апреля 2018. – Харьков. – С. 57-58.

21. Аксак Н.Г. Нейросетевая медицинская система принятия решений / Н.Г. Аксак, И.В. Новосельцев, А.П. Лучко // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: Матеріали восьмої міжнародної науково-технічної конференції Полтава -Баку – Харків - Жиліна. – 2018.- С.13.

22. Новосельцев И.В. Реализация многослойного персептрона с помощью стандарта OpenMP / И.В. Новосельцев, Н.Г. Аксак // 8-я международная научно-практическая конференция "Современные информационные и электронные технологии": Сб. трудов конференции - Одесса, 2007.-С.79.

23. Новосельцев И.В. Применение нейронных сетей для построения медицинской экспертной системы / И.В. Новосельцев, Н.Г. Аксак // Международная научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов по материалам Междун. Научн. конференции.- Харьков: ХНУРЭ, 2003,- С.341-342.

24. Новосельцев И.В. Применение нейронной сети при фильтрации спам сообщений / И.В. Новосельцев, Н.Г. Аксак //10-я Юбилейная международная

научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов. Ч.2. - Харьков: ХНУРЭ, 2004.- С.295-296.

25. И.В.Новосельцев, Н.Г. Аксак. Использование параллельных технологий для распознавания образов. // Материалы 11-го юбилейного международного форума «Радиоэлектроника и молодежь в XXI веке». – Харьков: ХНУРЕ, 2007. – с.306.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	21
ВСТУП	22
1 АНАЛІЗ СТАНУ ПИТАННЯ І ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ	30
1.1. Аналіз особливостей завдань розпізнавання, класифікації та кластеризації	30
1.1.1. Деякі особливості завдань розпізнавання, класифікації та кластеризації	30
1.1.2. Застосування нейронних мереж для задач розпізнавання, класифікації та кластеризації	33
1.1.3. Класифікація зображення за допомогою ШНМ	36
1.2. Штучні нейронні мережі, що застосовуються в задачах класифікації та розпізнавання образів	39
1.2.1. Багатошаровий персептрон (БП)	39
1.2.2. Радіально-базисна мережа (РБМ)	41
1.2.3. Мережа Кохонена	42
1.2.4. Когнітрон та неокогнітрон	42
1.2.5. Автоенкодер	43
1.2.6. Ймовірнісні мережі	44
1.2.7. Згорткова нейронна мережа	46
1.3. Методи визначення зміни характеристик зображення	47
1.3.1. Метод контролю зміни розмірів об'єкта за допомогою шаблону	47
1.3.2. Метод визначення неоднорідності забарвлення об'єкта	53
1.4. Висновки по розділу 1 і постановка задач дослідження	55
2 РОЗПІЗНАВАННЯ ОБРАЗІВ І КЛАСИФІКАЦІЯ НА ОСНОВІ БАГАТОШАРОВОГО ПЕРСЕПТРОНА	59
2.1. Визначення структури персептрона	59
2.2. Вибір критерію якості навчання БП	60
2.2.1. Використання в якості критерію функції помилки	60

2.2.2. Використання функції перехресної ентропії як функції втрат	62
2.3. Вибір функції активації нейронів	63
2.3.1. Сигмоїда	63
2.3.2. Гіперболічний тангенс	65
2.3.3. ReLU	66
2.3.4. SoftMax	69
2.4. Навчання перцептрона	70
2.4.1. Алгоритм зворотного поширення помилки	70
2.4.2. Метод Ньютона	73
2.4.3. Метод Левенберга-Марквардта	73
2.4.4. Квазіньютонівські методи	74
2.4.5. Методи сполучених градієнтів	75
2.4.6. Матрична процедура навчання БП	76
2.4.7. Робастне навчання багат шарового перцептрона	80
2.5. Адаптація багат шарового перцептрона на багатопроцесорні системи	84
2.5.1. Реалізація БП на багатопроцесорній системі	84
2.5.2. Аналіз продуктивності і результати тестування розробленої системи	91
2.6. Висновки по розділу 2	96
3 ОБРОБКА ЗОБРАЖЕНЬ НА ОСНОВІ ЗНМ	98
3.1. Архітектура ЗНМ	100
3.1.1 Шар підвибірки (пулінгу, MAX - pooling)	103
3.1.2. Нелінійний шар активації	104
3.1.3. Дропаут шар.	104
3.1.4. Нормалізуючий шар.	104
3.1.5. Повнозв'язний шар	105
3.2. Вибір функції активації і функції втрат	106
3.2.1. Вибір функції активації	106
3.2.2. Вибір функції втрат	107

	20
3.3. Навчання мережі	107
3.3.1. Навчання нейронів вихідного (повнозв'язного) шару	109
3.3.2. Навчання вихідного (повнозв'язного) шару мережі при виборі функції перехресної ентропії як функції втрат	111
3.3.3. Навчання нейронів конволюційного шару	112
3.4. Навчання на основі стохастичного градієнта	114
3.5. Робастне навчання	118
3.6. Висновки по розділу 3	122
4 ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ І РІШЕННЯ ПРАКТИЧНИХ ЗАВДАНЬ	124
4.1. Моделювання процесів розпізнавання і класифікації ШНМ прямого поширення	124
4.2. Моделювання процесу розпізнавання і класифікації загортальними ШНМ	131
4.3. Нейромережева експертна система медичної діагностики первинної меланоми шкіри	135
4.3.1. Завдання діагностування меланоми шкіри	135
4.3.2. Архітектура ЕС діагностики первинної меланоми шкіри	137
4.3.3. Функціонування експертної система медичної діагностики первинної меланоми шкіри	140
4.4. Висновки по розділу 4	144
ВИСНОВКИ	146
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	148
ДОДАТОК А	162
ДОДАТОК Б	163
ДОДАТОК С	170
ДОДАТОК Д	174

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

CNN – convolutional neural network (згортальна нейронна мережа)

GPU – graphics processing unit

CPU – central processing unit

ReLU – rectified linear unit

MIMD – multiple instruction multiple data

MLP – multilayer perceptron (багатошаровий персептрон)

MPI – message passing interface

NAG – nesterov accelerated gradient

PNN – probabilistic neural network (імовірнісна нейронна мережа)

БНМ – багатошарова нейронна мережа

БП – багатошаровий персептрон

ЗНМ – згортальна нейронна мережа

ЗП – алгоритм зворотного поширення помилки

РБМ – радіально-базисна мережа

МНМ – модульна нейронна мережа

ГНМ – гібридна нейронна мережа

ШНМ – штучна нейронна мережа

ЕС – експертна система

ВСТУП

Актуальність теми. На сьогодні одним з наукових і технологічних напрямків, що найбільш інтенсивно розвивається, є обробка і аналіз зображень.

Серед найважливіших завдань, пов'язаних з обробкою зображень і які відносять до завдань комп'ютерного зору, слід зазначити:

Classification — класифікація зображення по типу об'єкта, яке воно містить.

Semantic segmentation — визначення всіх пікселів об'єктів певного класу або фону на зображенні. Якщо кілька об'єктів одного класу перекриваються, їх пікселі ніяк не відокремлюються один від одного.

Object detection — виявлення всіх об'єктів зазначених класів і визначення рамки, що охоплює, для кожного з них.

Instance segmentation — визначення пікселів, що належать кожному об'єкту кожного класу окремо.

За останні роки було представлено ряд методів та алгоритмів, що застосовуються для вирішення цих завдань, серед одними з найбільш ефективних є штучні нейронні мережі (ШНМ) [1-10]. Використання персептрона, радіально-базисних мереж, автоенкодера, неокогнітрона, імовірнісних мереж виявилось досить ефективним при вирішенні широкого кола завдань. Однак новий поштовх до інтересу досліджень в даній області дала поява в 1998 році нового типу мереж – згортальних ШНМ (ЗНМ) [11-14].

Один з найбільш важливих критеріїв оцінки роботи нейронної мережі в разі розпізнавання зображення – це якість розпізнавання зображень. Варто зазначити, що для кількісної оцінки якості розпізнавання зображення за допомогою функціонування нейронної мережі найчастіше застосовується алгоритм мінімізації середньоквадратичної помилки. Навчання ШНМ прямого поширення полягає в корекції її вагових параметрів на основі мінімізації цієї помилки. Реалізація застосовуваних для цього градієнтних алгоритмів труднощів не викликає, тим

більше, що для обчислення градієнтів можна використовувати спеціальні бібліотеки, такі як Theano і TensorFlow.

Слід також зазначити, що на сьогоднішній день існує велика кількість стандартних архітектур побудови нейронних мереж, що істотно полегшує завдання побудови нейронної мережі з нуля і зводить її до підбору потрібного конкретного завдання структурі мережі.

У дисертаційній роботі розглядаються можливі шляхи вирішення актуальної проблеми розпізнавання зміни властивостей досліджуваного зображення з використанням нейромережевих технологій і побудови нейромережевої діагностичної системи, що допомагає лікареві поставити своєчасний діагноз на ранній стадії, коли людину ще можна врятувати.

Час навчання нейронних мереж, особливо при використанні стандартних персональних комп'ютерів і великих обсягів вихідних даних, може бути дуже великий. Проблема високої обчислювальної складності навчання нейромереж найчастіше вирішують, використовуючи більш потужний комп'ютер або спеціалізований апаратний нейроприскорювач. Однак такий шлях доступний далеко не всім.

В даний час, з переходом комп'ютерної індустрії на багатоядерні архітектури, є можливість збільшити продуктивність нейронних мереж на багатоядерних комп'ютерах. Однак існуючі програмні реалізації нейронних мереж не розраховані на використання переваг багатопроцесорних систем. Тому, для ефективного використання обчислювальних можливостей високопродуктивних систем потрібна розробка методик розпаралелювання, на основі яких може здійснюватися реалізація нейронних мереж. Застосування даних методик дозволить значно підвищити продуктивність при вирішенні багатьох практичних завдань.

У зв'язку з цим підвищення точності і швидкості обробки зображень, що дозволяють виявляти зміну їх характеристик, є актуальною науковою задачею.

Зв'язок роботи з науковими програмами. Дисертаційна робота виконана відповідно до плану науково-дослідних робіт Харківського національного університету радіоелектроніки в рамках таких держбюджетних тем: НДР «Синтез методів обробки інформації в умовах невизначеності на основі самонавчання і м'яких обчислень»; розділ «Гібридні моделі, що самонавчаються, в задачах обробки нечіткої інформації» (№ ДР 0107U003028); НДР «Еволюційні гібридні системи обчислювального інтелекту зі змінною структурою для інтелектуального аналізу даних», розділ «Еволюційні гібридні методи і моделі інтелектуальної обробки інформації зі змінною структурою в умовах невизначеності» (№ ДР 0110U000458); НДР «Нейро-фаззі системи для поточної кластеризації і класифікації послідовностей даних за умов їх викривленості відсутніми та аномальними спостереженнями», розділ «Адаптивні методи та моделі класифікації даних і прогнозування часових рядів за умов їх викривленості відсутніми та аномальними спостереженнями на основі штучних імунних систем» (№ ДР 0113U000361), затвердженими Міністерством освіти і науки України. Автор був одним з виконавців робіт за даними темами.

Мета та задачі дослідження. Метою дисертаційної роботи є розробка нейромережових методів і засобів обробки та оцінювання змін властивостей об'єкта за зображенням для підвищення точності контролю зміни розмірів і неоднорідності забарвлення спостережуваних об'єктів та ефективності їх класифікації.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- проаналізувати стан проблеми і особливості завдань класифікації, кластеризації та розпізнавання зображень;
- розглянути методи визначення зміни характеристик зображення і розробити метод контролю зміни розмірів об'єкта на основі еталона;
- розвинути нейромережовий метод визначення неоднорідності забарвлення досліджуваного об'єкта;

- розробити робастний метод прискореного навчання нейронних мереж прямого поширення;
- удосконалити метод адаптації ШНМ прямого поширення під високопродуктивні обчислювальні системи із загальною та розподіленою пам'яттю;
- розвинути нейромережевий метод класифікації, кластеризації та розпізнавання зображень на основі згортальних нейронних мереж;
- розвинути нейромережевий метод розпізнавання зміни параметрів зображень на прикладі меланоми з використанням мереж PNN і CNN;
- програмно реалізувати інтелектуальну систему діагностики первинної меланоми шкіри, що складається з ансамблю мереж PNN и CNN, яка на підставі опитування пацієнта та обробки ділянок шкірних новоутворень дозволить лікареві полегшити постановку діагнозу.

Об'єкт дослідження – процеси розпізнавання змін властивостей об'єктів за зображенням.

Предмет дослідження – методи та моделі розпізнавання змін властивостей об'єкта за зображенням на основі штучних нейронних мереж.

Методи дослідження: ґрунтуються на теорії обчислювального інтелекту, а саме, на методах теорії штучних нейронних мереж, яка дозволила синтезувати нейромережеві моделі та отримати процедури їх навчання; методи теорії оптимальності, за допомогою якої були синтезовані швидкодіючі процедури навчання; методи теорії розпізнавання образів, на основі яких були синтезовані моделі аналізованих об'єктів, що настроюються; методи імітаційного моделювання, що дозволили підтвердити ефективність отриманих результатів та розробити рекомендації щодо їх практичного використання. Експериментальні дослідження проводилися в лабораторних умовах і на реальних об'єктах.

Наукова новизна результатів дисертаційної роботи:

- вперше запропоновано метод контролю зміни розмірів об'єкта за допомогою шаблону, який характеризується використанням властивостей перетворення подібності об'єктів, що дозволяє підвищити точність вимірювань;

- вперше запропоновано процедуру навчання багат шарового персептрона, що являє собою матричний варіант процедури Качмажа (Уїдроу-Хоффа), яка характеризується використанням зони нечутливості, налаштування якої забезпечує отримання робастних оцінок шуканих параметрів мережі;

- удосконалено нейромережевий метод розпізнавання зміни параметрів зображень, який відрізняється використанням мереж PNN і CNN, що дозволяє підвищити точність контролю зміни розмірів спостережуваних об'єктів і ефективність їх класифікації;

- отримав подальший розвиток нейромережевий метод класифікації зображень на основі згорткових нейронних мереж шляхом використання в різних шарах різних функцій активації і робастного навчання параметрів мережі, що дозволяє підвищити точність класифікації.

Практичне значення одержаних результатів. Запропоновані методи та засоби розпізнавання змін властивостей об'єкта за зображенням доведені до рівня практичної реалізації і дозволяють:

- контролювати розміри об'єкта за допомогою еталона, який може бути використаний для дистанційного визначення зміни розмірів об'єкта, що спостерігається (патент на винахід № 84380 А по МПК (2006) G01C 11/00 від 10.10.2008. Бюл. №19.);

- виявляти важке захворювання шкіри на ранній стадії завдяки контролю зміни розмірів шкірного новоутворення, на підставі чого можна призначити й провести своєчасне лікування;

- накопичувати досвід для більш точної діагностики хворих за рахунок використання штучних нейронних мереж, що є актуальним для даної предметної області;

– використати розроблені методи в телемедицині, що дозволить своєчасно контролювати зміни дефектів шкіри.

На основі розроблених методів та алгоритмів програмно реалізовано інтелектуальну систему діагностування неоднорідності забарвлення, що складається з ансамблю мереж PNN і CNN цитологічних змін (первинної меланоми шкіри).

Результати дисертаційної роботи впроваджені: в інституті дерматології та венерології АМН України (акт про впровадження від 05.11.2007); у практичну діяльність товариства з обмеженою відповідальністю «Сана-мед» (акт про впровадження від 01.07.2009); у практичну діяльність товариства з обмеженою відповідальністю «Іпра-софт» (акт про впровадження від 17.05.2018).

Особистий внесок здобувача. Всі основні результати, що виносяться на захист, отримано автором самостійно. Внесок автора в публікаціях, написаних у співавторстві такий: в [1] запропоновано алгоритм розпізнавання розміру та кольору об'єкта за зображенням з використанням загорткової нейронної мережі; в [2] запропоновано систему медичної діагностики меланоми для діагностики злоякісних шкірних новоутворень; в [3] запропоновано модель реалізації багатошарового персептрона на багатопроцесорній системі для розпізнавання об'єкта за зображенням; в [4] запропонована гібридна модель на базі імовірнісної нейронної мережі з підсумовуванням результатів експертів для діагностики раку шкіри; в [5] запропоновано використання бінарних нейронів для обробки напівтонових зображень; в [6] запропоновано використання архітектури cloud-fog-dew для надання медичних послуг; в [7] запропоновано систему віддаленого нейромережевого розпізнавання меланоми; в [8] запропоновано модель адаптації медичного Інтернет-ресурсу; в [9] розроблено моделі продуктивності синтезу нейроалгоритмів на високопродуктивних архітектурах; в [10] запропоновано метод адаптації моделей нейронної мережі персептронного типу для розпізнавання меланоми на багатоядерні обчислювальні системи на основі стандарту OpenMP; в [11] розроблено гібридні нейронні моделі для розпізнавання зображень; в [12] запропоновано узагальнену

модель розпізнавання антропоморфних об'єктів; в [13] запропоновано Cloud-Fog-Dew архітектуру для сервіс-орієнтованих систем; в [14] запропоновано метод контролю зміни розмірів шкірного новоутворення за допомогою шаблону; в [15] запропоновано систему інтелектуального аналізу медичних даних; в [16] запропоновано агентно-орієнтований медичний Інтернет-ресурс; в [17] проведено аналіз способів прискорення нейроалгоритмів для розпізнавання кожних захворювань на основі технологій паралельного програмування; в [18] розроблено моделі продуктивності нейроалгоритмів на високопродуктивних архітектурах; в [19] реалізовано розпізнавання зображень об'єктів на основі багат шарового персептрона за допомогою стандарту MPI; в [20] розроблено моделі обробки великих даних для медичних закладів; в [21] запропоновано нейромережеву медичну експертну систему підтримки прийняття рішень; в [22] реалізовано багат шаровий персептрон на багатопроцесорних системах на основі стандарту OpenMP; в [23] проведено аналіз застосування нейронних мереж для створення медичних систем розпізнавання шкірних новоутворень; в [24] запропоновано використання нейронних мереж для фільтрації спам повідомлень при організації віддаленого моніторингу пацієнтів; в [25] запропоновано використання паралельних технологій для розпізнавання образів.

Апробація результатів дисертації. Основні положення дисертаційної роботи апробовані на таких міжнародних конференціях і форумах: на 9th IEEE Міжнародній конференції «International Conference on Dependable Systems, Services and Technologies» (DESSERT'2018, Kyiv, Ukraine); на 2-й Міжнародній науково-технічній конференції «Комп'ютерні та інформаційні системи і технології», (Харків, 2018); на Міжнародній науково-технічній конференції «Інформаційні системи та технології в медицині» (Харків: ISM-2018); на 2-й Міжнародній науково-практичній конференції «Обчислювальний інтелект (результати, проблеми, перспективи)» (Черкаси, 2013); на 10-й Міжнародній науково-технічній конференції «Системний аналіз та інформаційні технології» (Київ, 2008); на 7-й і 8-й Міжнародних конференціях «Високопродуктивні паралельні обчислення на кластерних системах»

(Росія, м. Санкт-Петербург, 2007, м. Казань, 2008); на 22-у Міжнародному молодіжному форумі «Радіoeлектроніка і молодь у ХХІ сторіччі» (Харків, 2018); на 8-й Міжнародній науково-технічній конференції «Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління» (Полтава–Баку–Харків–Жиліна, 2018); на 8-й Міжнародній науково-практичній конференції «Сучасні інформаційні та електронні технології» (Одеса, 2007); на 9-й і 10-й Міжнародних науково-технічних конференціях «Теорія і техніка передачі, прийому та обробки інформації» (Харків, 2003, 2004).

Публікації. За матеріалами дисертації опубліковано 25 друкованих праць, з них – 12 у фахових періодичних виданнях України з технічних наук; 7 статей включено у міжнародні наукометричні бази, 13 тез доповідей у матеріалах міжнародних наукових конференцій, одну з яких включено у базу Scopus.

Структура та обсяг роботи. Дисертація складається зі вступу, 4 розділів, висновків, списку використаних джерел, додатків. Загальний обсяг роботи складає 177 сторінок тексту, що містять 148 сторінок основного тексту, анотації на 16 сторінках, 32 рисунки, 7 таблиць, список використаних джерел з 141 найменування на 14 сторінках, 4 додатка на 15 сторінках.

1 АНАЛІЗ СТАНУ ПИТАННЯ І ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ

1.1. Аналіз особливостей завдань розпізнавання, класифікації та кластеризації

1.1.1. Деякі особливості завдань розпізнавання, класифікації та кластеризації

В задачах розпізнавання образів можна виділити два основних напрямки:

1. Вивчення здібностей до розпізнавання, якими володіють людські істоти і інші живі організми.

2. Розвиток теорії та методів побудови пристроїв, призначених для вирішення окремих завдань розпізнавання образів в певних прикладних областях.

Розпізнавання образів можна визначити як віднесення вихідних даних до певного класу за допомогою виділення істотних ознак або властивостей, які характеризують ці дані, із загальної маси несуттєвих деталей. У табл. 1.1 перерахований ряд завдань класифікації, а також відповідні вихідні дані і відповіді, які видаються системою[15].

Таблиця 1.1 - Завдання класифікації. Вихідні дані для цих завдань і види відповідей систем розпізнавання.

Змістовний характер завдання класифікації	Вид вихідних даних	Вид відповіді системи розпізнавання
Розпізнавання символів	Оптичні сигнали або елементи розгортки	Назва символу
Розпізнавання мови	Акустичні сигнали	«Ім'я» слова
Ідентифікація людини, що говорить	Голос	Ім'я людини, що говорить
Прогноз погоди	Синоптичні карти	Прогноз погоди
Встановлення медичного діагнозу	Симптоми захворювання	Вид захворювання
Прогноз стану фондової біржі	Фінансові новини та зведення	Прогноз підвищення або зниження цін на ринку

Проблема розпізнавання суттєво ускладнюється, якщо завдання, що вирішується, важко піддається формалізації або не формалізовано взагалі[16-18].

Головними особливостями важко формалізованих завдань є:

- відсутність можливості отримання точного рішення, оскільки алгоритм рішення, як правило, ніхто не знає або не може бути виконаний за розумний час через високу обчислювальну складність його реалізації навіть на швидкодіючих комп'ютерах;

- знання про предметну область і завдання, що вирішується, а також вихідні дані є в багатьох випадках неоднозначними, неповними і суперечливими, носять суб'єктивний характер.

З цих причин для вирішення важко формалізованих завдань застосовуються наближені методи, засновані на широкому використанні емпіричних даних, експертного оцінювання, нечітких і некласичних логік, моделювання, спеціально розроблених методик і т.д.

В цілому проблема розпізнавання образів складається з двох частин: навчання та розпізнавання. Навчання здійснюється шляхом показу окремих об'єктів із зазначенням їх приналежності до того чи іншого образу. В результаті навчання система, що розпізнає, повинна придбати здатність реагувати однаково на всі об'єкти одного образу і по-різному – на всі об'єкти різних образів. Дуже важливо, що процес навчання повинен завершитися тільки шляхом показів кінцевого числа об'єктів без будь-яких інших підказок. Об'єктами навчання можуть бути або картинки, або інші візуальні зображення (літери), або різні явища зовнішнього світу, наприклад звуки, стани організму при медичному діагнозі, стан технічного об'єкта в системах управління і ін. Важливо, що в процесі навчання вказуються тільки самі об'єкти і їх приналежність образу. За навчанням йде процес розпізнавання нових об'єктів, який характеризує дії вже навченої системи. Автоматизація цих процедур і становить проблему навчання розпізнаванню образів.

Процес рішення тісно пов'язаної з розпізнаванням завдання класифікації, незалежно від природи вихідних даних, можна розбити на наступні етапи:

- інсталяційний – формується постановка завдання на змістовному рівні;
- постановочний – визначається тип прикладної задачі в термінах теорії класифікації;
- інформаційний – полягає у виробленні плану збору вихідної інформації, її попередньому аналізі і редагуванні;
- апіорний математико-постановочний – полягає у виборі на підставі висновків, отриманих в результаті реалізації попередніх етапів, базових математичних моделей для математичної постановки конкретного завдання класифікації;
- розвідувальний аналіз – складають спеціальні методи попередньої статистичної обробки та аналізу вихідних даних з метою виявлення їх імовірнісної і геометричної природи;
- апостеріорний математико-постановочний – уточнюється вибір базової математичної моделі з урахуванням висновків, отриманих на етапі реалізації розвідувального аналізу;
- обчислювальний – проводиться програмна реалізація обраного математичного апарату для вирішення конкретного завдання;
- підсумковий – проводиться аналіз і інтерпретація результатів проведеного дослідження [18].

Таким чином, вид завдання класифікації визначається в результаті реалізації перших трьох етапів процесу дослідження; наприклад, якщо попередня вибіркова інформація відсутня, а апіорні відомості про класи об'єктів являють собою лише деякі припущення самого загального характеру, то задача відноситься до класу задач розпізнавання образів з самонавчанням [19].

Разом з тим, на практиці часто виявляється, що завданню властива нечіткість, яка значно ускладнює або взагалі робить неможливим отримання рішення, так що на

перший план виходить проблема усунення нечіткості, яка притаманна задачі класифікації [19,20].

За останні роки було представлено ряд методів та алгоритмів, застосовуваних для вирішення цих завдань, серед яких одними з найбільш ефективних є штучні нейронні мережі (ШНМ) [1-10]. Використання персептрона, радіально-базисних мереж, автоенкодера, неокогнітрона, імовірнісних мереж виявилося досить ефективним при вирішенні широкого кола завдань. Однак новий поштовх до інтересу досліджень в даній області дала поява в 1998 р нового типу мереж – згортальних ШНМ (ЗНМ).

1.1.2. Застосування нейронних мереж для задач розпізнавання, класифікації та кластеризації

Питання про необхідні і достатні властивості мережі для вирішення того чи іншого роду завдань являє собою цілий напрям нейрокомп'ютерної науки. Так як проблема синтезу ШНМ сильно залежить від задачі, що розв'язується, дати загальні докладні рекомендації важко. У більшості випадків оптимальний варіант виходить на основі інтуїтивного підбору, хоча в літературі наведено докази того, що для будь-якого алгоритму існує нейронна мережа, яка може його реалізувати. Зупинимось на цьому докладніше [2,21].

Багато задач: розпізнавання образів (зорових, мовних і т.д.), виконання функціональних перетворень при обробці сигналів, управління, прогнозування, ідентифікації складних систем і т.д., зводяться до наступної математичної постановки. Необхідно побудувати відображення $X \rightarrow Y$ таке, щоб на кожен можливий вхідний сигнал X формувалася правильний вихідний сигнал Y . Відображення задається кінцевим набором пар ((вхід), (відомий вихід)). Число таких пар (навчальних прикладів) істотно менше загального числа можливих поєднань значень вхідних і вихідних сигналів.

В задачах розпізнавання образів X – деяке представлення образу (зображення, вектор чисел і т.д.), Y – номер класу, до якого належить вхідний образ.

В результаті побудови такого відображення (тобто $X \rightarrow Y$) необхідно досягти того, щоб:

- забезпечувалося формування правильних вихідних сигналів у відповідності з усіма прикладами навчальної вибірки;
- забезпечувалося формування правильних вихідних сигналів у відповідності з усіма можливими вхідними сигналами, які не ввійшли в навчальну вибірку [48]

Для того, щоб для деякого вхідного впливу ШНМ давала бажану відповідь, мережу навчають шляхом послідовного пред'явлення вхідних векторів з одночасним підстроюванням ваг відповідно до певної процедури. В процесі навчання ваги мережі поступово стають такими, щоб кожен вхідний вектор виробляв вихідний вектор.

Розрізняють алгоритми навчання з вчителем і без вчителя [1-3]. Навчання з вчителем передбачає, що для кожного вхідного вектору існує цільовий вектор, що представляє собою необхідний вихід. Разом вони утворюють навчальну пару. Зазвичай мережа навчається на деякому числі таких навчальних пар. Пред'являється вихідний вектор, обчислюється вихід мережі і порівнюється з відповідним цільовим вектором, різниця (помилка) за допомогою зворотного зв'язку подається в мережу і ваги змінюються відповідно до алгоритму, які прагнуть мінімізувати помилку. Вектори навчальної множини пред'являються послідовно, обчислюються помилки і ваги підлаштовуються для кожного вектору до тих пір, поки помилка по всьому навчальному масиві не досягне прийнятного рівня.

Найбільш поширеним алгоритмом навчання з учителем є алгоритм зворотного поширення помилки (ЗП).

Навчання без вчителя є набагато більш правдоподібною моделлю навчання в біологічній системі [1]. Завдання класифікація без вчителя зводиться до розбиття

деякого набору об'єктів, що характеризуються векторами ознак, на класи еквівалентності[11].

Головна риса, що робить навчання без учителя привабливим, - це його "самостійність". Процес навчання, як і в разі навчання з учителем, полягає в підстроювання ваг синапсів. Підстроювання синапсів може проводитися тільки на підставі інформації, доступної в нейроні, тобто його стану і вже наявних вагових коефіцієнтів.

Більшість сучасних алгоритмів навчання НМ без вчителя зросла з концепцій Д.О. Хебба, який запропонував модель навчання без вчителя, в якій синаптична вага зростає, якщо активовані обидва нейрони, джерело і приймач[1,3].

Інший алгоритм навчання без учителя - алгоритм Кохонена - передбачає підстроювання синапсів на підставі їх значень від попередньої ітерації.

Однак досить часто визначити кількість класів, на які повинні бути класифіковані вхідні образи, неможливо. У цьому випадку число класів визначається експериментально. Розроблено безліч методів для визначення кількості класів в процесі навчання.

Одним з представників даних методів є метод простого перебору. Ідея методу полягає в тому, щоб почавши з малого числа класів поступово збільшувати його до тих пір, поки не буде отримана «хороша» класифікація. Метод простого перебору добре працює при невеликому числі класів. При досить великому числі класів і великому обсязі вхідних образів, які необхідно розбити на класи, така процедура підбору стає занадто повільною, так як число пробних класифікацій можна порівняти за порядком величини з числом образів, які класифікуються. В результаті це призводить до великих обчислювальних витрат, які часто витрачаються даремно.

Альтернативою методу перебору служить метод відпалу, ідея якого полягає в тому, що на основі критерію якості класу приймається рішення про видалення цього класу, розбивці класу на два або про злиття цього класу з іншим [22]. Два класи зливаються, якщо їх ядра ближче, ніж середня відстань від елемента класу до ядра в

одному з них. (Можливі варіанти: використання середньої відстані по обох класах, використання порогового коефіцієнта, що показує, у скільки разів має відстань між ядрами перевершувати середня відстань від елемента до ядра і ін.) [1]. Якщо ж клас «хороший», то він залишається без змін.

Існує багато евристичних алгоритмів класифікації без вчителя, заснованих на використанні заходів близькості між об'єктами. Кожен з них має свою область застосування, а найбільш поширеним недоліком є відсутність чіткої формалізації завдання: здійснюється перехід від ідеї кластеризації прямо до алгоритму, в результаті невідомо, що шукається (але щось в будь-якому випадку знаходиться, іноді - непогано) [11].

1.1.3. Класифікація зображення за допомогою ШНМ

При вирішенні задач класифікації необхідно віднести наявні статичні зразки (характеристики ситуації на ринку, дані медогляду, інформація про клієнта) до певних класів. Можливо кілька способів подання даних. Найбільш поширеним є спосіб, при якому зразок видається вектором. Компоненти цього вектора є різні характеристики зразка, які впливають на прийняття рішення про те, до якого класу можна віднести даний зразок. Наприклад, для медичних завдань в якості компонентів цього вектора можуть бути дані з медичної карти хворого. Таким чином, на підставі деякої інформації про приклад, необхідно визначити, до якого класу його можна віднести. Класифікатор таким чином відносить об'єкт до одного з класів відповідно до визначеного розбиттям N -мірного простору, який називається простором входів, і розмірність цього простору є кількістю компонент вектора. Перш за все, потрібно визначити рівень складності системи. В дійсних задачах виникає ситуація, коли кількість зразків обмежена, що ускладнює визначення складності завдання. Можливо виділити три основні рівні складності. Перший (найпростіший) - коли класи можна розділити прямими лініями (або гіперплощинами, якщо простір входів має розмірність більше двох) - так звана

лінійна роздільність. У другому випадку класи неможливо розділити лініями (площинами), але їх можна виділити за допомогою більш складного поділу - нелінійна роздільність. У третьому випадку класи перетинаються і можна говорити тільки про ймовірнісну подільність. В ідеальному варіанті після попередньої обробки повинні отримати лінійно роздільне завдання, що значно спрощує побудову класифікатора. На жаль, при вирішенні реальних завдань, як правило, існує обмежена кількість зразків, на підставі яких і проводиться побудова класифікатора. При цьому не можна провести таку передобробку даних, яка дозволить досягти лінійної роздільності прикладів. Мережі з прямим зв'язком є універсальним засобом апроксимації функцій, що дозволяє їх використовувати в рішенні задач класифікації. Як правило, нейронні мережі виявляються найбільш ефективним способом класифікації, тому що генерують фактично велике число регресійних моделей (які використовуються в рішенні задач класифікації статистичними методами).

На жаль, в застосуванні ШНМ в практичних завданнях виникає ряд проблем, пов'язаних в першу чергу з тим, що заздалегідь не відомо, якої складності (розміру) може знадобитися мережа для досить точної реалізації відображення. Ця складність може виявитися надмірно високою, що потребує складної архітектури мереж.

У загальному випадку можна сказати, що в мережі з одним прихованим шаром вектор, відповідний вхідному зразку, перетворюється прихованим шаром в новий простір, який може мати іншу розмірність, а потім гіперплощини, відповідні нейронам вихідного шару, поділяють його на класи. Таким чином, мережа розпізнає не тільки характеристики вихідних даних, але і "характеристики характеристик", сформовані прихованим шаром. Для побудови класифікатора необхідно визначити, які параметри впливають на прийняття рішення про те, до якого класу належить зразок. При цьому можуть виникнути дві проблеми. По-перше, якщо кількість параметрів мала, то може виникнути ситуація, коли один і той же набір вихідних даних відповідає прикладам, які перебувають в різних класах. Тоді неможливо навчити ШНМ, і система не буде коректно працювати (неможливо знайти мінімум,

який відповідає такому набору вихідних даних). Вихідні дані обов'язково повинні бути несуперечливі. Для вирішення цієї проблеми необхідно збільшити розмірність простору ознак (кількість компонент вхідного вектора, відповідного зразку). Але при збільшенні розмірності простору ознак може виникнути ситуація, коли число прикладів може стати недостатнім для навчання мережі, і вона замість узагальнення просто запам'ятає приклади з навчальної вибірки і не зможе коректно функціонувати. Таким чином, при визначенні ознак необхідно знайти компроміс з їх кількістю. Далі необхідно визначити спосіб представлення вхідних даних для нейронної мережі, тобто визначити спосіб нормування, яке необхідне, оскільки нейронні мережі працюють з даними, представленими числами в діапазоні 0-1, а вихідні дані можуть мати довільний діапазон або взагалі бути нечисловими даними. Можливі різні способи нормування, починаючи від простого лінійного перетворення в необхідний діапазон і закінчуючи багатовимірним аналізом параметрів і нелінійним нормуванням в залежності від впливу параметрів один на одного [1,2].

Завдання класифікації при наявності двох класів може бути вирішене на мережі з одним нейроном у вихідному шарі, який може приймати одне з двох значень 0 або 1, залежно від того, до якого класу належить зразок. При наявності декількох класів виникає проблема, пов'язана з поданням цих даних для виходу мережі. Найбільш простим способом представлення вихідних даних в такому випадку є вектор, компоненти якого відповідають різним номерам класів. При цьому i -я компонента вектора відповідає i -му класу. Всі інші компоненти при цьому встановлюються в 0. Тоді, наприклад, другому класу буде відповідати 1 на 2 виході мережі і 0 на інших.

Під підзадачею в даному випадку розуміється те, що мережа визначає наявність однієї з компонент вектора. Тобто вихідний вектор розбивається на групи по два компонента в кожній таким чином, щоб в них увійшли всі можливі комбінації компонент вхідного вектора. Число цих груп можна визначити як кількість невпорядкованих вибірок по два з вихідних компонента. Правильний вибір обсягу

мережі має велике значення. Побудувати невелику і якісну модель часто буває неможливо, а велика модель буде просто запам'ятовувати приклади з навчальної вибірки і не виробляти апроксимацію, що, природньо, призведе до некоректної роботи класифікатора.

1.2. Штучні нейронні мережі, що застосовуються в задачах класифікації та розпізнавання образів

Серед існуючої в даний час великої кількості мережевих структур для задач класифікації і розпізнавання зазвичай використовують статичні ШНМ прямого поширення: БП, радіально-базисна мережа (РБМ), мережа Кохонена, автоенкодер, неокогнітрон, стохастичні та згорткові мережі.

1.2.1. Багатошаровий персептрон (БП)

БП являє собою нейронну мережу з декількома шарами, кожен з яких складається з обчислювальних вузлів (нейронів) [1-10].

Входи мережі під'єднані до кожного нейрона в першому шарі. Виходи нейронів першого шару потім стають входами для нейронів другого шару і так далі. Останній шар є вихідним, всі інші шари між вхідним і вихідним шарами називаються прихованими шарами. Архітектура багатошарового персептрона може бути зручно записана як $n_0 - n_1 - \dots - n_l$, де n_0 є розмірністю вхідного вектору мережі, а n_i , $1 \leq i \leq l$ позначає номери вузлів у відповідних шарах.

Таким чином, БП використовує наступну апроксимацію нелінійного оператора $f(\bullet)$:

$$y(k) = \hat{f}(k) = f^q \left[(W^q)^T f^{q-1} \left[(W^{q-1})^T f^{q-2} \left[\dots f^1 \left[(W^1 x(k) + b_1 \right)^T \right] \dots \right] \right] + b_q, \quad (1.2)$$

де W^i – вектор вагових параметрів нейронів i -го шару мережі; $f^i[\bullet]$ – активаційна функція (АФ) i -го шару; b_i – зміщення i -го нейрона.

Так як в практичних додатках доводиться проводити різні операції не тільки з самою активаційною функцією, але і з її першою похідною, необхідним є використання в якості активаційної монотонної, диференційованої і обмеженої функції. Особливо важливу роль відіграють такі функції при моделюванні нелінійних залежностей між вхідними та вихідними змінними. Це так звані логістичні або сигмоїдальні (S-образні) функції.

До числа таких функцій відносяться

– логістична (уніполярна)

$$f_{\log}(z) = \frac{1}{1 + e^{-\alpha z}}; \quad (1.3)$$

$$\frac{d}{dz} f_{\log}(z) = \alpha f_{\log}(z)(1 - f_{\log}(z)),$$

– гіперболічного тангенса (біполярна)

$$f_{th}(z) = \tanh(\alpha z) = \frac{e^{\alpha z} - e^{-\alpha z}}{e^{\alpha z} + e^{-\alpha z}}; \quad (1.4)$$

$$\frac{d}{dz} f_{th}(z) = 1 - \tanh^2(\alpha z).$$

Функції (1.3) і (1.4) можуть бути виражені одна через одну.

Слід також зазначити, що перевага функції $f_{th}(z)$ перед $f_{\log}(z)$ полягає в її симетричності щодо початку координат (в ряді випадків це суттєво полегшує обчислення).

Апроксимуючі можливості БП були досліджені і описані багатьма авторами [1-10,23]. При цьому не вирішеним залишається питання вибору кількості нейронів в прихованих шарах, які забезпечують задану точність рішення задачі, хоча існують деякі рекомендації щодо визначення необхідної кількості нейронів, які будуть розглянуті нижче.

Важливим також є те, що доказ можливостей апроксимації за допомогою БП передбачає, що ваги задані коректно. Однак залишається відкритим питання вибору алгоритму навчання.

Більш докладно БП буде розглянуто в розділі 2.

1.2.2. Радіально-базисна мережа (РБМ)

РБМ є альтернативною БП [1-10,24-33]. РБМ має двошарову обчислювальну структуру. Прихований шар складається з матриці вузлів, кожен з яких містить вектор параметрів, званий центром. Вузол обчислює деяку відстань між центром і вхідним вектором мережі і передає результат через нелінійну функцію. Вихідний шар є по суті набором лінійних суматорів.

РБМ використовує наступну апроксимацію $f(\bullet)$:

$$\hat{y}(k) = \hat{f}(k) = w_0 + \sum_{i=1}^N w_i \Phi_i(x) = w_0 + W^T \Phi(k), \quad (1.5)$$

де $\Phi(k)$ – вектор обраних базисних функцій (БФ).

У даних мережах в якості базисних вибираються деякі функції відстані між векторами $\phi_i(u) = f(\|u - c_i\|)$.

Вектори c_i називають центрами базисних функцій. Функції $\phi_i(u)$ обираються невід'ємними і зростаючими при збільшенні $\|u - c_i\|$. В якості міри близькості векторів u і c_i береться зазвичай евклідова або манхеттенська метрика, а в якості функцій $\phi(\cdot)$ найчастіше обирають гауссову функцію

$$\phi(z, \sigma) = \exp(-z^2 / \sigma^2), \quad (1.6)$$

де σ^2 - параметр відхилення (дисперсія).

Крім того, використовується також плоска сплайн-функція, мультіквадратична і зворотна мультіквадратична функція.

Топологія РБМ, очевидно, є схожою на архітектуру багатошарового персептрона, а відмінність полягає тільки в характеристиках (параметрах) прихованих нейронів.

Існуюча схожість між РБМ і двошаровим персептроном свідчить про те, що дана мережа має аналогічні з персептроном можливості апроксимації. Це, фактично, і було доведено.

1.2.3. Мережа Кохонена

ШНМ Кохонена [1,34] або карта ознак, що самоорганізується (Self organizing featured map, SOFM або просто SOM) була запропонована фінським дослідником Т. Кохоненом. Алгоритм Кохонена дає можливість будувати нейронну мережу для поділу векторів вхідних сигналів на підгрупи. Мережа складається з нейронів, що утворюють прямокутну решітку на площині.

Елементи вхідних сигналів подаються на входи всіх нейронів мережі. У процесі роботи алгоритму налаштовуються синаптичні ваги нейронів.

Вхідні сигнали - вектори дійсних чисел - послідовно пред'являються мережі. Бажані вихідні сигнали не визначаються. Після того, як було пред'явлено достатню кількість вхідних векторів, синаптичні ваги мережі визначають кластери.

В даній мережі нейрони вихідного шару називаються кластерними елементами, їх кількість визначає максимальну кількість груп, на які система може розділити вхідні дані. Збільшуючи кількість нейронів вихідного шару можна збільшувати деталізацію результатів процесу кластеризації.

Основною відмінністю даної технології від розглянутих раніше нейромереж, яких навчають за алгоритмом ЗП, є те, що при навчанні використовується метод навчання без учителя, тобто результат навчання залежить тільки від структури вхідних даних. Нейронні мережі даного типу часто застосовуються для вирішення найрізноманітніших завдань, від відновлення пропусків у даних до аналізу даних і пошуку закономірностей, наприклад, у фінансовій задачі.

Алгоритм функціонування карт, що самонавчаються, є один з варіантів кластеризації багатовимірних векторів. Прикладом таких алгоритмів може служити алгоритм k-найближчих середніх (k-means).

1.2.4. Когнітрон та неокогнітрон

Когнітрон [35] своєю архітектурою схожий на будову зорової кори, має ієрархічну багатошарову організацію, в якій нейрони між шарами пов'язані тільки

локально. Він є потужним засобом розпізнавання зображень, однак вимагає високих обчислювальних витрат, які на сьогоднішній день недосяжні [3].

Неокогнітрон [36-38] є подальшим розвитком ідеї когнітрону, більш точно відображає будову зорової системи і дозволяє розпізнавати образи незалежно від їх перетворень, обертань, перекручувань і змін масштабу. Ця ієрархічна багат шарова нейронна мережа згорткового типу, була створена для розпізнавання рукописних символів.

Основна перевага нейронної мережі неокогнітрон - це його здатність розпізнавати коректно не тільки навчальну вибірку, а й вектори, утворені шляхом часткових зрушень, ротації, афінних перетворень і зашумлення. Так само неокогнітрон є інваріантним до зміни кольорової палітри образу.

Неокогнітрон показав відмінні результати з розпізнавання образів, підданих зсувам у позиції, зашумленню, спотворенню форми. Проте, неокогнітрон не справлявся із завданнями розпізнавання образів, повернених на деякий кут, а процес розпізнавання займав досить тривалий час. З метою усунення цього недоліку були запропоновані дві моделі неокогнітрону, стійких до повороту вхідного образу:

- R-неокогнітрон, з динамічним числом нейронів або обертально-інваріантний неокогнітрон;
- TD-R-неокогнітрон, з постійним числом нейронів.

R-неокогнітрон може розпізнати візерунки під будь-якими кутами. Розпізнавальна здатність TD-R-неокогнітрона є такою ж, як і здатність розпізнавання R-неокогнітрона; він толерантний для поворотів, деформацій тощо.

1.2.5. Автоенкодер

Однією з найпоширеніших архітектур автоенкодера є нейронна мережа прямого поширення без зворотних зв'язків, що містить вхідний, прихований і вихідний шари [39,40]. На відміну від персептрона вихідний шар автоенкодера повинен містити стільки ж нейронів, скільки і вхідний шар. Дані на вхідному шарі

стискаються на прихованому шарі і відновлюються на вихідному шарі, таким чином виділяються приховані ознаки.

Мета автоенкодера полягає в тому, щоб вихід нейронної мережі був найбільш близьким до вхідного вектора. Для цього на топологію мережі накладаються особливі умови:

- 1) кількість нейронів прихованого шару повинна бути менше, ніж розмірність вхідних даних;
- 2) кількість неактивних нейронів в прихованому шарі має значно перевищувати кількість активних нейронів.

Перше обмеження дозволяє отримати стиснення даних при передачі вхідного сигналу на вихід мережі. Таке стиснення можливо, якщо в даних є приховані взаємозв'язки, кореляція ознак або структура. Друге обмеження дозволяє отримувати нетривіальні результати, навіть коли кількість нейронів прихованого шару перевищує розмірність вхідних даних, іншими словами, мета автоенкодера – отримувати тільки найбільш особливі ознаки.

1.2.6. Ймовірнісні мережі

Ймовірнісні мережі (PNN) введені Д. Шпехтом [41,42], мають цілу низку цікавих властивостей, які роблять їх особливо привабливими для вирішення прикладних завдань розпізнавання образів. До числа таких властивостей відносяться:

- висока швидкість навчання, що досягається за рахунок відмови від застосування ітеративного оптимізаційного процесу настройки внутрішніх параметрів;
- оптимальність, яка полягає в тому, що при наявності достатнього для конкретного завдання обсягу даних рішення, отримується, збігається з рішенням, які даються байєсовським класифікатором;

– можливість отримання ефективного вирішення при наявності пропущених або зашумлених даних.

Архітектура мережі PNN базується на архітектурі радіальної базисної мережі (RBF), але в якості другого шару використовує так званий конкуруючий шар, який підраховує ймовірність приналежності вхідного вектора до того чи іншого класу i , в кінцевому рахунку, зіставляє вектор з тим класом, ймовірність приналежності до якого вище.

Запропонована Д. Шпехтом мережа моделювала роботу байєсівського класифікатора, що забезпечує мінімум середнього ризику класифікації.

Функціонування ймовірнісної нейронної мережі описується в такий спосіб:

$$z_i = \exp\left[\frac{-D_i}{2\sigma^2}\right], \quad (1.7)$$

де $D_i = (x - u_i)^T (x - u_i)$ – це відстань між вхідним вектором x та навчальним вектором u з категорії, що класифікується, σ - параметр згладжування. Функція оцінки щільності розподілу $f_j(x)$ для класу з другого шару нейронної мережі обчислюється за формулою:

$$f_j(x) = \frac{1}{(2\pi)^{p/2} \sigma^p n \sum_{i=1}^n z_i}, \quad (1.8)$$

де n – розмірність вхідного вектора.

Ідея полягає в тому, що для кожного зразка можна прийняти рішення на основі вибору найбільш ймовірного класу з тих, яким міг би належати цей зразок. Таке рішення вимагає оцінки функції щільності ймовірностей для кожного класу. Ця оцінка виходить в результаті розгляду навчальних даних. Формальним правилом є те, що клас з найбільш щільним розподілом в області невідомого примірника буде мати перевагу в порівнянні з іншими класами. Так само буде мати перевагу і клас з високою апріорною ймовірністю або високою ціною помилки класифікації. Для двох класів A і B відповідно до цього правила обирається клас A , якщо

$$p_A L_A f_B(x) > p_B L_B f_B(x), \quad (1.9)$$

де p – апіорна ймовірність, L — ціна помилки класифікації, а $f(x)$ — функція щільності ймовірностей. Взагалі, оцінка вартості помилки класифікації вимагає гарного знання додатку, але в багатьох додатках ціна помилки класифікації і апіорні ймовірності вибираються однаковими для всіх класів.

Використання зазначеного правила передбачає, що всі співмножники творів $p_i L_i f(x)$ відомі. На практиці ж справа йде трохи інакше. Якщо апіорні ймовірності p_i ($i = \overline{1, k}$) або відомі, або можуть бути досить просто оцінені, то функції розподілу ймовірностей $f_i(x)$ ($i = \overline{1, k}$), як правило, не відомі, а їх оцінювання представляє досить складне завдання. Зазвичай для спрощення припускають, що вони є нормальними і оцінюють параметри цих розподілів. Більш доцільним, однак, видається застосування непараметричних методів оцінювання, зокрема, вікон (методу) Парзена [43].

Для мережі PNN не потрібно навчання в тому сенсі, яке потрібне для мереж зі зворотним поширенням помилок, так як всі параметри мережі PNN (число елементів і значення ваг) визначаються безпосередньо навчальними даними.

Істотним недоліком таких мереж є їх обсяг. PNN-мережу фактично вміщує в себе все навчальні дані, тому вона вимагає великого обсягу пам'яті і може повільно працювати.

У роботах [42-43] розглянуті підходи для практичного застосування PNN - мереж. В роботі [44] запропонована реалізація ймовірнісної нейронної мережі для розпізнавання зашумлених образів.

1.2.7. Згорткова нейронна мережа

В основі згорткових нейронних мереж лежать такі ідеї [11-14]:

– локальні рецепторні поля (забезпечують локальну двовимірну зв'язність нейронів);

- загальні ваги (забезпечують детектування деяких рис в будь-якому місці зображення і зменшують загальне число вагових коефіцієнтів);
- ієрархічна організація з просторовими підвибірками.

Згорткові мережі являють собою варіацію архітектури багатошарового персептрона, і включають в себе згорткові шари, шари підвибірки (субдискретизація) і повнозв'язні шари. Перші два типи шарів, чергуючись між собою, формують вхідний вектор ознак для багатошарового персептрону.

Архітектура згорткової мережі використовує переваги двовимірної структури вхідних даних – зображень за допомогою методу локальної зв'язності, обмежуючи кількість зв'язків між нейронами прихованого згорткового шару і вхідними даними. Тобто, кожен нейрон прихованого шару пов'язаний тільки з обмеженою локальною (які не мають розривів) ділянкою зображення.

Детально дана мережа буде розглянута в розділі 3.

1.3. Методи визначення зміни характеристик зображення

1.3.1. Метод контролю зміни розмірів об'єкта за допомогою шаблону

Однією з основних завдань при обробці зображень (особливо в нестационарних умовах) є оцінка зміни розмірів досліджуваного об'єкта. Слід зазначити, що дана проблема особливо важлива при постановці медичного діагнозу. Для визначення зміни розмірів досліджуваного об'єкта використовується метод контролю цих розмірів за допомогою шаблону. Даний метод належить до науково - технічного напрямку обробки і аналізу цифрових зображень і може бути використаний для дистанційного визначення зміни розмірів об'єкта, що спостерігається, наприклад, в медицині для контролю дерматологічних захворювань.

Відзначимо, що даний спосіб знайшов широке застосування саме в медичних дослідженнях, зокрема, в способі оцінювання та порівняння з плином часу цифрових зображень поверхні шкіри [45]. Цифрове зображення області інтересу спостережуваного об'єкта заноситься в базу даних персонального комп'ютера для

обробки і калібрування освітлення, створення монохроматичних зображень кожного з трьох кольорових каналів (R, G, B), виконання сегментації і обчислення статистики зображення.

Головна перевага даного способу - різноманітність статистичних даних, як наслідок, оперувати даним методом може тільки професіонал. Визначення відстані до області інтересу спостережуваного об'єкта є основною проблематикою даного типу завдань, так як в разі помилкових вихідних даних будуть отримані значно спотворені результати. Цей пристрій вирішує дану проблему за допомогою ПЗЗ камери з телецентричною оптикою, що в свою чергу залишає якість спостереження на тому ж рівні, але збільшує його вартість і обмежує контингент користувачів даної системи.

Відомий також спосіб контролю зміни параметрів об'єкта порівнянням зображень об'єкта до і після змін параметрів [46]. Отримують зображення проекції об'єкта до і після ймовірної зміни параметрів на цифровій світлочутливій матриці в умовах постійних координат і кутів орієнтації фотоапарата. Поєднують ці зображення проекцій і визначають різницю координат (x, y) проекції характерного елемента об'єкта, а саме точки, лінії, площі на цифровій світлочутливій матриці.

Його істотним недоліком є умова постійних координат і кутів орієнтації фотоапарата, що з одного боку вирішує проблему спотворення проекції об'єкта на світлочутливій матриці в залежності від положення камери по відношенню до об'єкта, але з іншого боку на практиці вище зазначена вимога не виконується, так як ставить дуже жорсткі рамки перед користувачем, і в разі порушення умов зйомки призведе до помилкового результату.

Найбільш близьким до способу, який розробляється нижче, є спосіб відображення цифрового зображення пошкодження шкіри [47], заснований на візуальній ідентифікації зміни розмірів пошкоджень шкіри. Отримують фотографії за допомогою цифрового фотоапарата, що позиціонується, для одночасного захоплення шаблону і області інтересу об'єкта, що спостерігається, заноситься в базу

даних персонального комп'ютера. Програмне забезпечення на персональному комп'ютері фільтрує зображення та надає за допомогою друку напівпрозоре зображення. Персональний комп'ютер по черзі відображає вирівняні поточне і базове зображення, що дозволяє візуально ідентифікувати прогрес чи регрес в області інтересу об'єкта.

В даному винаході використовується видошукач, принтер і призначена досить трудомістка і складна система зйомки об'єкта, що спостерігається. Результат можна побачити в якості маркування ділянок, які відрізняються, що не дозволяє отримувати достатню точність.

В основу пристрою, що розроблюється, поставлено завдання збільшення точності обчислення реальної площі об'єкта, що спостерігається, збільшення якості та зручності користування, а також зниження його вартості шляхом одночасного цифрового фотографування шаблону і області інтересу об'єкта, що спостерігається, автоматичної обробки отриманих даних і усуваючи залежність результату від дорогої техніки [48,49].

Дане завдання вирішується за рахунок того, що може використовуватися цифровий фотоапарат стандартної конфігурації, технічні характеристики якого для всіх зроблених з його допомогою фотографій є постійними і не враховуються. Основні характеристики, що вносять спотворення параметрів об'єктів на фотографії, такі як фокусна відстань, система лінз об'єктива, масштаб зйомки і розмір матриці, спотворюють на отриманому зображенні в рівній мірі як область інтересу об'єкта, що спостерігається, так і шаблон, тому в математичних розрахунках також опускаються.

При роботі з фотоапаратом необхідно враховувати, що реальні об'єкти на фотографії в залежності від фокусної відстані, відстані від об'єктива до площини орендованих об'єктів, зберігають свої параметри (кути, кольоровість, рельєф і т.д.), але змінюють масштаб. Для вирішення цієї проблеми використовується заданий шаблон, параметри і площа якого відомі.

Обчислення реальної площі об'єкта, що спостерігається засноване на використанні властивостей перетворення подібності об'єктів, при якому відстані між точками змінюються в одне і те ж число раз.

Нехай відома реальна площа шаблону, площі шаблону і спостережуваного об'єкта на фотографії, тоді пропонується наступне обчислення реальної площі об'єкта, що спостерігається:

$$\frac{S_{об.р.}}{S_{ш.р.}} = \frac{S_{об.ф.}}{S_{ш.ф.}}, \quad (1.10)$$

$$S_{об.р.} = \frac{S_{об.ф.} \cdot S_{ш.р.}}{S_{ш.ф.}}, \quad (1.11)$$

де $S_{об.р.}$ – реальна площа спостережуваного об'єкта, $S_{ш.р.}$ – реальна площа шаблону, $S_{об.ф.}$ – площа спостережуваного об'єкта на фотографії, $S_{ш.ф.}$ – площа шаблону на фотографії.

Відстань до об'єкта, що спостерігається і шаблону на одній площині в даному рішенні не використовується, що значно полегшує спосіб і знижує вартість спостереження, так як не потрібно таке дороге обладнання як видошукач, принтер, CCD фотоапарат з телецентричною оптикою і ін.

Другою перевагою використання обчислення (1.11) є його незалежність від одиниць виміру розмірів об'єкта і шаблону.

Детальний опис даного способу пов'язане з прикладом його конкретного виконання на тестових об'єктах і для підрахунку площі досліджуваного зображення.

Сутність методу полягає у виконанні наступних дій:

1. Виконується цифрова фотозйомка області інтересу (поверхні шкіри, наприклад, з невусом) і шаблону (монети) в одній площині, дотримуючись перпендикулярності оптичної осі до спостережуваного об'єкту. Спостережуваний об'єкт і шаблон для найбільшої точності вимірювань повинні перебувати у фокусі. Перпендикулярність оптичної осі забезпечує збереження параметрів об'єкта, що спостерігається без спотворення кутів і порушення пропорційності відрізків, інакше

при зміщенні оптичної осі фокус фотоапарата також зміщується на площині, яка фотографується, і спотворює об'єкти.

2. Виконується обчислення реальної площі шаблону.

Визначаючи параметри реального шаблону, можна скористатися стандартною лінійкою, штангель - циркулем, рулеткою і т.д.

3. Цифрове зображення області інтересу об'єкта, що спостерігається і шаблону заноситься в базу даних персонального комп'ютера.

4. Програмне забезпечення на персональному комп'ютері виконує сегментацію і обчислення площ об'єкта, що спостерігається і шаблону на цифровому зображенні.

5. Виконується обчислення реальної площі невиса (2) при першому спостереженні.

Грунтуючись на властивостях перетворення подібності, обчислюють реальну площу об'єкта, що спостерігається на основі розрахованої площі цифрового зображення об'єкта $S_{об.ф.}^{(1)}$ і шаблону $S_{ш.ф.}^{(1)}$, що дозволяє ідентифікувати прогресування змін в області інтересу. На підставі обчислення (1.11), вираховується дійсна площа об'єкта, що спостерігається $S_{об.р.}^{(1)}$, яка вважається базовою.

6. Виконується повторна цифрова фотозйомка області інтересу і шаблону в одній площині, дотримуючись перпендикулярності оптичної осі до спостережуваного об'єкту.

Умовою цього винаходу є використання одного і того ж цифрового фотоапарата протягом усього часу спостереження.

7. Цифрове зображення області інтересу об'єкта, що спостерігається і шаблону заноситься в базу даних персонального комп'ютера.

8. Програмне забезпечення персонального комп'ютера виконує сегментацію і обчислення площ об'єкта, що спостерігається і шаблону на цифровому зображенні.

9. Виконується обчислення реальної площі об'єкта (1.11) при другому спостереженні $S_{об.р.}^{(2)}$.

10. Оцінка змін розмірів об'єкта, що спостерігається.

Маючи базову площу об'єкта (п.5) і відповідну площу об'єкта при повторному спостереженні (п.9), обчислюють різницю Δ_1 на другому спостереженні:

$$\Delta_1 = S_{об.р.}^{(1)} - S_{об.р.}^{(2)}, \quad (1.12)$$

У разі позитивної різниці можна зробити висновок, що об'єкт в розмірах зменшився, в разі від'ємної різниці - об'єкт в розмірах збільшився, а в разі нульової різниці - об'єкт в розмірах не змінився.

11. При необхідності повторити кроки 2 - 10.

Фотозйомка поверхні шкіри і шаблону проводиться в одній площині, дотримуючись перпендикулярність оптичної осі до спостережуваного об'єкту. Допустимий кут відхилення від перпендикуляра при фотографуванні α визначається за формулою:

$$\alpha = \arcsin \frac{S_{ш.ф.}}{\pi \times r^2}, \quad (1.13)$$

де, r - радіус шаблону.

На рисунку 1.1 представлений графік залежності допустимого кута відхилення від перпендикуляра при фотографуванні.

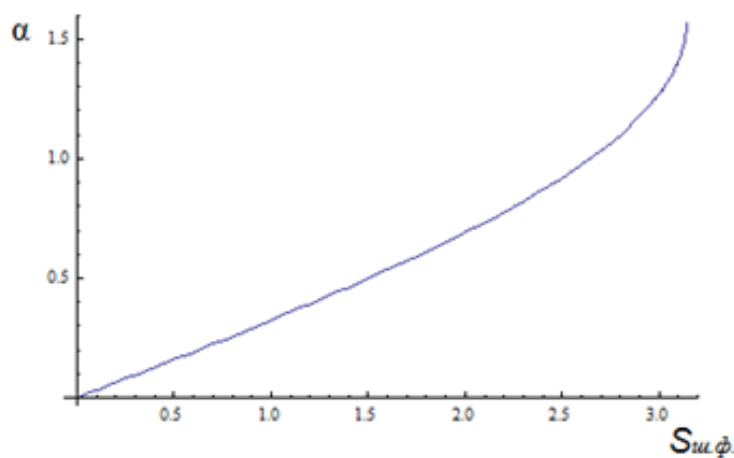


Рисунок 1.1 – Графік залежності допустимого кута відхилення від перпендикуляра

Таким чином, застосування даного методу забезпечує незалежність від параметра відстані до області інтересу об'єкта, що спостерігається завдяки врахуванню властивостей перетворення подібності. Автоматичний розрахунок, який дає можливість збільшити точність обчислень площі об'єкта, що спостерігається, а також поліпшити якість спостереження і можливість здійснювати спостереження віддалених значно розширює діапазон застосування даного підходу.

1.3.2. Метод визначення неоднорідності забарвлення об'єкта

Неоднорідність забарвлення досліджуваного об'єкта містить інформацію про його стан і може свідчити про негативні тенденції в зміні цього стану (особливо це характерно для задач медичної діагностики). Однак визначити неоднорідність алгоритмічним шляхом досить складно.

Нейронні мережі добре підходять для вирішення подібних завдань. Існує безліч підходів щодо перетворення вхідного зображення в вектор перед подачею на вхід ШНМ.

Перший підхід має на увазі кодування з використанням унітарного коду (One-Hot encoding). Так як зазвичай зображення представляється в форматі RGB (рис. 1.2), на градації кожного кольору (каналу) відводиться 8 біт, а для подання в повному діапазоні потрібно 24 біта (16777216 кольорів). Таким чином, кожен піксель зображення може бути представлений в діапазоні від 0 (білий) до 16777216 (чорний).



Рисунок 1.2—Кольоровий простір RGB.

При навчанні ШНМ на вхід подається кожен піксель області інтересу, а для його уявлення потрібно 16777216 нейронів. Якщо зображення мають дозвіл 200×200 , то кількість нейронів у вхідному шарі буде дорівнює 671 088 640 000.

Для зменшення кількості нейронів у вхідному шарі можна працювати з зображеннями в колірному просторі LAB. На відміну від RGB, де всі канали

відповідають за колір і контраст, в LAB весь контраст міститься в каналі L, а вся інформація про колір (або якщо бути більш точним, кольоровість) - в каналах A і B. Тому канал L можна уявити, як чорно-білу версію кольорового зображення. (рис. 1.3).



Рисунок 1.3 – Кольоровий простір LAB.

Таким чином, в LAB для представлення кольору досить 16 біт (канали A і B). Тобто для подання кожного пікселя нам буде потрібно 65536 нейронів, а для зображення 200×200 - 2621440000.

Для подальшого зменшення кількості вхідних нейронів можна використовувати принцип кодування типу термометра. При такому способі кодування область зміни параметра ділиться на кінцеве число інтервалів і для подання всіх значень, що лежать в m -му інтервалі, використовуються стани бінарних нейронів.

Недоліком даного методу є, те, що, якщо ми маємо справу навіть з зображенням маленької розмірності час навчання мережі дуже великий через велику розмірність вхідного вектору.

Ще одним підходом для перетворення RGB зображення в вектор є поділ значення окремого елемента в кожному з каналів на 255 (нормалізація) і подальше схлопування в вектор. Таким чином для зображення 200×200 буде потрібен вектор розмірності 120000 кожен елемент якого знаходиться в діапазоні $[0..1]$.

Для вирішення завдання визначення неоднорідності забарвлення навчальний набір даних необхідно розділити на два класи A-окрас однорідний і B-окрас неоднорідний. Незважаючи на те, що ШНМ є універсальними апроксиматорами,

підбір відповідної архітектури далеко нетривіальний. Оптимальна кількість прихованих шарів, а також кількість нейронів в цих шарах неможливо визначити заздалегідь - надто велика їх кількість веде до перенавчання, в той час як мала не забезпечує достатньої точності навчання. Цей процес ресурсномісткий і забирає багато часу.

1.4. Висновки по розділу 1 і постановка задач дослідження

З аналізу стану проблем розпізнавання, класифікації та кластеризації, обробки зображень видно, що за останні роки був представлений ряд методів та алгоритмів, застосовуваних для вирішення цих завдань, серед яких одними з найбільш ефективних є штучні нейронні мережі (ШНМ).

Серед існуючої в даний час великої кількості мережевих структур для задач класифікації і розпізнавання зазвичай використовують статичні ШНМ прямого поширення: БП, радіально-базисна мережа (РБМ), мережа Кохонена, автоенкодер, неокогнітрон, стохастичні та згорткові мережі. Однак новий поштовх до інтересу досліджень в даній області дала поява в 1998 р нового типу мереж – згортальних ШНМ (ЗНМ).

Для того, щоб для деякого вхідного впливу ШНМ давала бажану відповідь, її потрібно навчити. Навчання здійснюється шляхом послідовного пред'явлення вхідних векторів з одночасним підстроюванням ваг відповідно до певної процедури. В процесі навчання ваги мережі поступово стають такими, щоб кожен вхідний вектор виробляв вихідний вектор.

Один з найбільш важливих критеріїв оцінки роботи нейронної мережі в разі розпізнавання зображення - це якість розпізнавання зображень. Варто зазначити, що для кількісної оцінки якості розпізнавання зображення за допомогою функціонування нейронної мережі найчастіше застосовується алгоритм мінімізації середньоквадратичної помилки. Навчання ШНМ прямого поширення полягає в корекції її вагових параметрів на основі мінімізації цієї помилки [3]. Реалізація

застосовуваних для цього градієнтних алгоритмів труднощів не викликає, тим більше, що для обчислення градієнтів можна використовувати спеціальні бібліотеки, такі як Theano і TensorFlow.

Багато задач розпізнавання образів (зорових, мовних і т.д.), виконання функціональних перетворень при обробці сигналів, управління, прогнозування, ідентифікації складних систем і т.д., зводяться до наступної математичної постановки. Необхідно побудувати відображення $X \rightarrow Y$ таке, щоб на кожен можливий вхідний сигнал X формувалася правильний вихідний сигнал Y . Відображення задається кінцевим набором пар ((вхід), (відомий вихід)). Число таких пар (навчальних прикладів) істотно менше загального числа можливих поєднань значень вхідних і вихідних сигналів.

В задачах розпізнавання образів X - деяке уявлення образу (зображення, вектор чисел і т.д.), Y - номер класу, до якого належить вхідний образ.

В результаті побудови такого відображення (тобто $X \rightarrow Y$) необхідно домогтися того, щоб:

- забезпечувалося формування правильних вихідних сигналів у відповідності з усіма прикладами навчальної вибірки;
- забезпечувалося формування правильних вихідних сигналів у відповідності з усіма можливими вхідними сигналами, які не ввійшли в навчальну вибірку.

При вирішенні задач класифікації на підставі деякої інформації про приклад (зразок), необхідно визначити, до якого класу його можна віднести. Класифікатор таким чином відносить об'єкт до одного з класів відповідно до визначеного розбиттям N -мірного простору, яке називається простором входів, і розмірність цього простору є кількістю компонент вектора.

При вирішенні реальних завдань маємо обмежену кількість зразків, на підставі яких і проводиться побудова класифікатора. При цьому неможливо провести таку передобробку даних, при якій буде досягнута лінійна роздільність прикладів. Мережі з прямим зв'язком є універсальним засобом апроксимації функцій, що

дозволяє їх використовувати в рішенні задач класифікації. Як правило, нейронні мережі виявляються найбільш ефективним способом класифікації, тому що генерують фактично велике число регресійних моделей (які використовуються в рішенні задач класифікації статистичними методами).

На жаль, при застосуванні нейронних мереж в практичних завданнях виникає ряд проблем. По-перше, заздалегідь не відомо, якої складності (розміру) може знадобитися мережа для досить точної реалізації відображення. Ця складність може виявитися надмірно високою, що потребує складної архітектури мереж.

Якщо завдання визначення структури є дискретною оптимізаційною (комбінаторною), то пошук оптимальних параметрів здійснюється в безперервному просторі за допомогою класичних методів оптимізації, що вимагають обчислення градієнта функціоналу, що використовується (алгоритм зворотного поширення помилки (ЗП), метод сполучених градієнтів, алгоритм Гаусса-Ньютона, Левенберга-Марквардта і т.д.).

Одним з основних завдань при обробці зображень (особливо в нестационарних умовах) є оцінка зміни розмірів досліджуваного об'єкта. Слід зазначити, що дана проблема особливо важлива при постановці медичного діагнозу. У зв'язку з цим в даному розділі запропонований метод контролю розмірів об'єкта за допомогою шаблону, який може бути використаний для дистанційного визначення зміни розмірів об'єкта, що спостерігається, наприклад, в медицині для контролю дерматологічних захворювань.

Відзначимо, що даний спосіб знайшов широке застосування саме в медичних дослідженнях, зокрема, в способі оцінювання та порівняння з плином часу цифрових зображень поверхні шкіри.

Вельми важливим і складним завданням є розпізнавання неоднорідності забарвлення досліджуваного об'єкта, яке містить інформацію про його стан і може свідчити про негативні тенденції в зміні цього стану (особливо це характерно для задач медичної діагностики). Однак визначити неоднорідність алгоритмічним

шляхом досить складно. Тому представляється доцільним застосування для вирішення цього завдання ШНМ.

Метою дисертаційної роботи є розвиток нейромережевих методів і засобів обробки і оцінювання змін властивостей зображення з підвищеною точністю при наявності апіорної і поточної невизначеності.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- проаналізувати стан проблеми і особливості завдань класифікації, кластеризації та розпізнавання зображень;
- розглянути методи визначення зміни характеристик зображення і розробити метод контролю зміни розмірів об'єкта на основі шаблону;
- розвинути нейромережевий метод визначення неоднорідності забарвлення досліджуваного об'єкта;
- розробити робастний метод прискореного навчання нейронних мереж прямого поширення;
- удосконалити метод адаптації ШНМ прямого поширення під високопродуктивні обчислювальні системи із загальною та розподіленою пам'яттю;
- розвинути нейромережевий метод класифікації, кластеризації та розпізнавання зображень на основі згортальних нейронних мереж;
- розвинути нейромережевий метод розпізнавання зміни параметрів зображень на прикладі меланом з використанням мереж PNN і CNN;
- програмно реалізувати інтелектуальну систему діагностики первинної меланоми шкіри, що складається з ансамблю мереж PNN и CNN, яка на підставі опитування пацієнта та обробки ділянок шкірних новоутворень дозволить лікареві полегшити постановку діагнозу

Вирішенню поставлених завдань, що виникають в процесі класифікації та обробки зображень на основі нейромережевого підходу, присвячені наступні розділи.

2 РОЗПІЗНАВАННЯ ОБРАЗІВ І КЛАСИФІКАЦІЯ НА ОСНОВІ БАГАТОШАРОВОГО ПЕРСЕПТРОНА

2.1. Визначення структури персептрона

Існують деякі рекомендації щодо визначення необхідної кількості нейронів для БП і РБМ. Кількість нейронів прихованого шару БП попередньо можна оцінити таким чином [5]:

$$\frac{N}{10} - n - m \leq n_k \leq \frac{N}{2} - n - m, \quad (2.13)$$

де n_k – кількість нейронів в прихованому шарі, n – кількість вхідних сигналів;
 m – кількість вихідних сигналів, N – кількість елементів статичної вибірки, необхідної для навчання.

З іншого боку, значення n_k можна оцінити таким чином:

$$\frac{N \cdot \varepsilon_o}{n} > n_k, \quad (2.2)$$

де ε_o – відносна помилка програмування нейромережі.

З урахуванням (2.2) можна записати

$$\begin{cases} \frac{N}{10} - n - m \leq \frac{N \cdot \varepsilon_o}{n}; \\ \frac{N \cdot \varepsilon_o}{n} \leq \frac{N}{2} - n - m. \end{cases} \quad (2.3)$$

Вирішуючи (2.3) відносно N , отримуємо

$$\frac{2n(n+m)}{(n-10\varepsilon_o)} \leq N \leq \frac{10n(n+m)}{(n-10\varepsilon_o)}. \quad (2.4)$$

За $\varepsilon_o \rightarrow 0$ нерівність (2.4) набуває вигляду

$$2n(n+m) \leq N \leq 10n(n+m). \quad (2.5)$$

Кількість синаптичних ваг для персептрона з одним прихованим шаром обчислюються так

$$K_w = n \cdot n_k \cdot m. \quad (2.6)$$

2.2. Вибір критерію якості навчання БП

2.2.1. Використання в якості критерію функції помилки

Навчання ШНМ - ітеративний процес [1-3]. На кожній ітерації обчислюються мережеві виходи для одного (або більше) зразка в наборі навчання, і коригуються мережеві ваги з метою зменшення помилки між фактичним мережевим виходом $(y_{i,p}^L)(i=1,...N^L)$ і цільовим виходом для даного зразка. При використанні відповідної мережної архітектури і алгоритму навчання мережеві ваги наближаються до значень, при яких мережевий вихід стає прийнятно близьким до цільового виходу для кожного зразка в наборі навчання.

При нейромережевому підході завдання навчання мережі полягає у визначенні вектору її параметрів θ (ваг, параметрів активаційних функцій, зсувів і т.ін.), що забезпечує мінімум функціоналу

$$E = \frac{1}{2} \sum_{p=1}^P \sum_{i=1}^{N^L} \rho(e_{i,p}) \quad (2.7)$$

де $\rho(e_{i,p})$ – деяка функція втрат, що залежить від виду закону розподілу перешкоди ξ ; $e_{i,p} = d_{i,p} - y_{i,p}^L$ – помилка реакції мережі на i -му виході при поданні p зразка, тобто що є рішенням системи рівнянь

$$\frac{\partial E}{\partial \theta_j} = \sum_{p=1}^P \sum_{i=1}^{N^L} \rho'(e_{i,p}) \frac{\partial e_{i,p}}{\partial \theta_j} \quad (2.8)$$

де $\rho'(e_{i,p}) = \frac{\partial \rho(e_{i,p})}{\partial e_{i,p}}$ – функція впливу.

Введення вагової функції $\omega_{i,p} = \frac{\rho'(e_{i,p})}{e_{i,p}}$ дозволяє записати систему рівнянь

(2.8) наступним чином:

$$\sum_{p=1}^P \sum_{i=1}^{N^L} \omega_{i,p} e_{i,p} \frac{\partial e_{i,p}}{\partial \theta_i} = 0 \quad (2.9)$$

При цьому мінімізація функціоналу (2.7) буде еквівалентна мінімізації зваженого квадратичного функціоналу

$$E = \frac{1}{2PN^L} \sum_{p=1}^P \sum_{i=1}^{N^L} \omega_{i,p} e_{i,p}^2. \quad (2.10)$$

і його нормалізована її версія

$$E = \frac{1}{2PN^L} \sum_{p=1}^P \sum_{i=1}^{N^L} \omega_{i,p} e_{i,p}^2. \quad (2.11)$$

Вибір функції помилки має настільки ж істотний вплив на ефективність ШНМ, як і вибір алгоритму навчання. Найбільш широко використовуваними функціями помилки є квадратична функція помилки (2.10) і нормалізована її версія, яка називається середньоквадратичною функцією помилки.

Перевага останньої полягає в тому, що вона нечутлива як до числа зразків в наборі навчання, так і до числа нейронів у вихідному шарі мережі, що дозволяє використовувати її для порівняння в різних завданнях навчання.

Особливістю функцій (2.10) і (2.11) є те, що зменшення E може бути пов'язано зі збільшенням числа помилково класифікованих зразків; фактично, зменшення E до мінімально прийнятного рівня будь-якої з цих функцій помилки не обов'язково відповідає мінімальному числу нерозпізнаних образів. Це пов'язано з двома особливостями поверхні функціоналу помилки БП – локальними мінімумами і багатовимірними «плато» [50].

Кажуть, що БП застряг в локальному мінімумі, коли величина мережевої помилки для послідовних періодів навчання стабілізується на порівняно високому рівні. Локальні мінімуми, як відомо, утворюються в специфічних тестових завданнях [51]. Однак вони відсутні, якщо завдання, що навчається, є лінійно розподілювальне [52] або число прихованих вершин мережі збігається з кількістю зразків в наборі навчання [53] або число зразків менше або дорівнює числу параметрів моделі [54]. У

кожному з цих випадків відбувається навчання мережі з обраної архітектурою. Однак, на жаль, жоден з наявних результатів не містить практичних рекомендацій для реальних додатків.

Все частіше дослідники використовують альтернативні функції помилки, що дозволяють поліпшити характеристики збіжності алгоритмів навчання БП. Один з таких підходів полягає в застосуванні функції помилки, що дозволяє мережі вийти з локального мінімуму, зокрема, функції помилки з взаємною ентропією [50]

$$E = - \sum_{p=1}^P \sum_{i=1}^{N^L} \ln \left[\left(y_{i,p}^L \right)^{d_{i,p}} \left(1 - y_{i,p}^L \right)^{1-d_{i,p}} \right] \quad (2.12)$$

При використанні (2.12), градієнти помилки для погано помітних зразків значно вище, ніж при використанні квадратичної і середньоквадратичної функцій помилки, що дозволяє мережі швидше рухатися в плоских областях простору ваг (тобто при перетині плато).

Навчання БП є успішним при досягненні мережевою помилкою допустимого значення, що визначається з умов конкретного завдання. Слід зазначити, що досягнення мінімального значення помилки часто є небажаним, оскільки призводить до явища перенавчання БП і погіршення його фільтруючих і узагальнюючих властивостей. Виняток становлять завдання апроксимації функцій, для яких потрібна висока точність одержуваних рішень.

Перший теоретично доведений алгоритм для настройки ваг при навчанні БП відомий як алгоритм зворотного поширення помилки (ЗП) [1,55-57]. Сьогодні ЗП може розглядатися як шаблонний тест, з яким порівнюються всі інші методи навчання.

2.2.2. Використання функції перехресної ентропії як функції втрат

Так як в якості функції активації в вихідному шарі буде використовуватися функція активації softmax, яка перетворює будь-який вхідний вектор в вектор ймовірностей, то для порівняння двох імовірнісних розподілів необхідно вибрати

коректну міру. В якості такого заходу буде використовуватися перехресна (крос) ентропія [1]:

$$C = -\sum_j d_j \ln y_j^L, \quad (2.13)$$

де d_j – необхідний вихід для поточного навчального прикладу; y_j^L – реальний вихід нейронної мережі.

Властивості функції втрат:

$$1) C \geq 0$$

$$2) \text{Мінімум } C \text{ досягається за } y^L = d.$$

Крім того, необхідно відзначити, що

$$C = -\sum_j d_j \ln y_j^L \geq 0.$$

Слід зазначити, що для цієї функції проблема уповільнення відсутня, так як

$$\frac{\partial C}{\partial b_i^L} = y_i^L - d_i, \quad (2.14)$$

$$\frac{\partial C}{\partial w_{ik}^L} = y_k^{L-1} (y_k^L - d_i). \quad (2.15)$$

2.3. Вибір функції активації нейронів

При виборі функції активації керуються наступним:

- функція активації повинна бути диференційована. Це необхідно для алгоритму ЗП, який буде розглянуто в наступних розділах;
- простота і швидкість в обробці;
- вихід повинен бути центрований відносно нуля (також для ЗП).

2.3.1. Сигмоїда

Дана функція, що описується співвідношенням, показана на рис. 2.1

$$f_{\log}(z) = \frac{1}{1 + e^{-\alpha z}}. \quad (2.16)$$

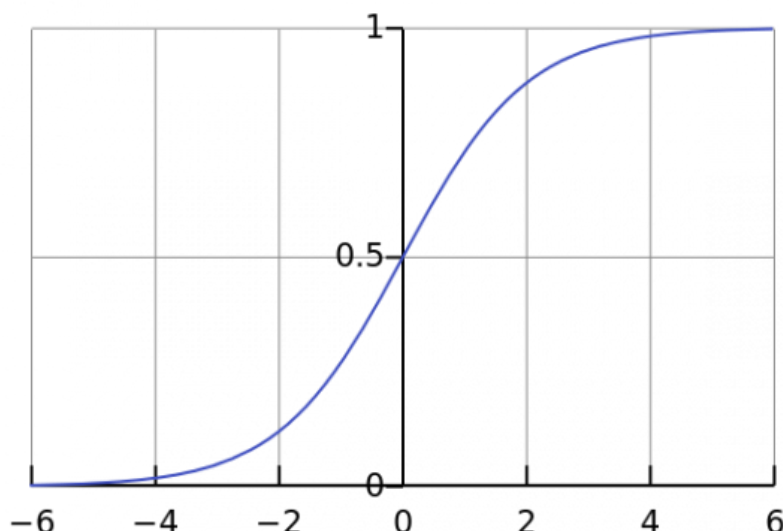


Рисунок2.1– Сигмоїда

Розглянемо її переваги.

По-перше, сигмоїда нелінійна за своєю природою, а комбінація таких функцій виробляє теж нелінійну функцію.

По-друге, вона не бінарна, що робить активацію аналоговою. Для сигмоїд також характерний гладкий градієнт.

Сигмоїда дійсно виглядає відповідною функцією для задач класифікації, так як будь-яке мала змінна X в діапазоні значень від -2 до 2 тягне істотні зміни змісту Y , тобто Y має тенденцію притискатися до одного з країв кривої, що дозволяє знаходити чіткі межі при прогнозі.

Крім того, наявність фіксованого діапазону значень функції — $[0,1]$, не приводить до помилок в разі великих значень активації.

Похідна цієї функції, що використовується в алгоритмі навчання, має вигляд

$$\frac{d}{dz} f_{\log}(z) = \alpha f_{\log}(z) (1 - f_{\log}(z)), \quad (2.17)$$

тобто обчислюється дуже просто.

Недоліки:

– обчислення складні і займають багато часу;

- на кінцях сигмоїд значення Y мають тенденцію слабо реагувати на зміни в X , тобто градієнт в таких областях приймає маленькі значення, що призводить до зникнення градієнтів (в алгоритмі ЗП) в разі, якщо значення вихідного значення близько до 0 або 1 (сигнали не будуть проходити через дані нейрони);
- забезпечують повільну збіжність;
- вихід сигмоїд не відцентрований щодо нуля. Ця властивість є небажаною, оскільки нейрони в наступних шарах отримуватимуть значення, що не відцентровані щодо нуля, що впливає на динаміку градієнтного спуску.

На сьогодні сигмоїда є однією з найбільш застосовуваних активаційних функцій в нейромережах при вирішенні задач класифікації.

2.3.2. Гіперболічний тангенс

Ще одна активаційна функція, яка часто використовується - гіперболічний тангенс, що описується виразом

$$f_{th}(z) = \tanh(\alpha z) = \frac{e^{\alpha z} - e^{-\alpha z}}{e^{\alpha z} + e^{-\alpha z}}, \quad (2.18)$$

показана на рис. 2.2

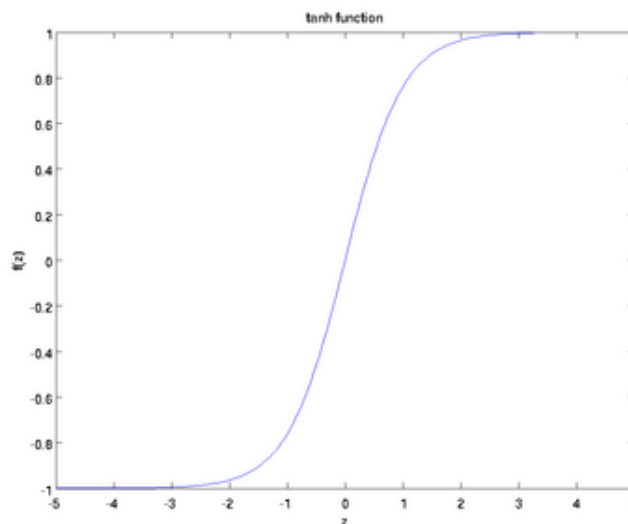


Рисунок 2.2 – Гіперболічний тангенс

Гіперболічний тангенс є скоригованою сигмоїдною функцією

$$f_{th}(z) = 2 f_{\log}(2z) - 1.$$

Тому така функція має ті ж характеристики, що й сигмоїда. При цьому градієнт тангенційної функції, що обчислюється за формулою

$$\frac{d}{dz} f_{th}(z) = 1 - \tanh^2(\alpha z), \quad (2.19)$$

більше, ніж у сигмоїди (похідна крутіша). Також, як і сигмоїди, гіперболічному тангенсу властива проблема зникнення градієнта. Однак, на відміну від сигмоїд, вихід даної функції відцентрований щодо нуля. Тому на практиці завжди краще використовувати гіперболічний тангенс, а не сигмоїду.

2.3.3. ReLU

В останні роки велику популярність набула функція активації під назвою «випрямляч» (rectifier, за аналогією з однополуперіодним випрямлячем в електротехніці). Нейрони з цією функцією активації називаються ReLU (rectified linear unit). ReLU (рис.2.3) описується виразом

$$f(z) = \max(0, z), \quad (2.20)$$

і реалізує простий пороговий перехід в нулі.

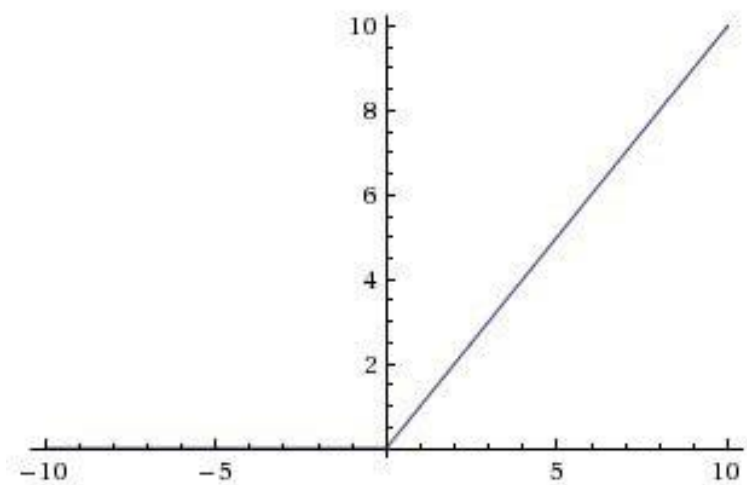


Рисунок 2.3 – Функція ReLU

ReLU нелінійна за своєю природою, є хорошим апроксиматором, так як будь-яка функція може бути апроксимована комбінацією ReLU. Область допустимих значень ReLU — $[0, \infty)$, тобто активація може «вибухнути».

Переваги:

- обчислення сигмоїд і гіперболічного тангенса вимагає виконання ресурсномістких операцій, таких як зведення в ступінь, в той час як ReLU може бути реалізована за допомогою простого порогового перетворення матриці активацій в нулі. Крім того, ReLU не схильна до насичення;

- застосування ReLU істотно підвищує швидкість збіжності стохастичного градієнтного спуску (в деяких випадках до 6 разів [58]) в порівнянні з сигмоїдою і гіперболічним тангенсом. Вважається, що це обумовлено лінійним характером і відсутністю насичення даної функції;

- відсікає непотрібні деталі в каналі за умови від'ємного виходу.

Недоліки:

- ReLU не завжди досить надійні і в процесі навчання можуть виходити з ладу («вмирати»). Наприклад, великий градієнт, що проходить через ReLU, може привести до такого оновлення ваг, що даний нейрон ніколи більше не активується. Ця проблема вирішується за допомогою вибору належного значення параметра швидкості навчання;

- в даний час існує ціле сімейство різних модифікацій ReLU.

Leaky ReLU. ReLU з «витоком» (leaky ReLU, LReLU) (рис. 2.4) являє собою одну зі спроб вирішити описану вище проблему виходу з ладу звичайних ReLU. Звичайна ReLU на інтервалі $x < 0$ дає на виході нуль, в той час як LReLU має на цьому інтервалі невелике від'ємне значення (кутовий коефіцієнт близько 0,01). Тобто функція для LReLU має вигляд $f(x) = \alpha x$ коли $x < 0$ і $f(x) = x$ коли $x \geq 0$, де α – мала константа. Деякі дослідники повідомляють про успішне застосування даної функції активації, але результати не завжди стабільні.

ParametricReLU. Для параметричної ReLU (parametricReLU, PReLU) (рис. 2.4) кутовий коефіцієнт на від'ємному інтервалі не задається попередньо, а визначається на основі даних. Автори публікації [59] стверджують, що застосування даної функції активації є ключовим фактором, що дозволив перевершити рівень людини в задачі розпізнавання зображень ImageNet. Процес зворотного поширення помилки і поновлення для PReLU досить простий і подібний до відповідного процесу для традиційних ReLU.

Randomized ReLU. Для рандомізованої ReLU (randomized ReLU, RReLU) (рис. 2.4) кутовий коефіцієнт на від'ємному інтервалі під час навчання генерується випадковим чином із заданого інтервалу, а під час тестування залишається постійним.

Результати говорять про те, що для всіх трьох наборів даних модифіковані ReLU перевершили традиційні. У разі LReLU більшого значення кутового коефіцієнта α забезпечує більш високу точність. PReLU схильні до перенавчання на малих наборах даних (помилка на навчальному наборі найменша з усіх, в той час як помилка на тестовому наборі більше, ніж у конкуруючих модифікацій ReLU). При цьому PReLU все ж перевершує традиційний ReLU. Слід зазначити, що RReLU істотно перевершує інші функції активації на наборі даних NDSB. Це говорить про те, що RReLU дозволяє уникнути перенавчання, оскільки цей набір містить менше навчальних даних.

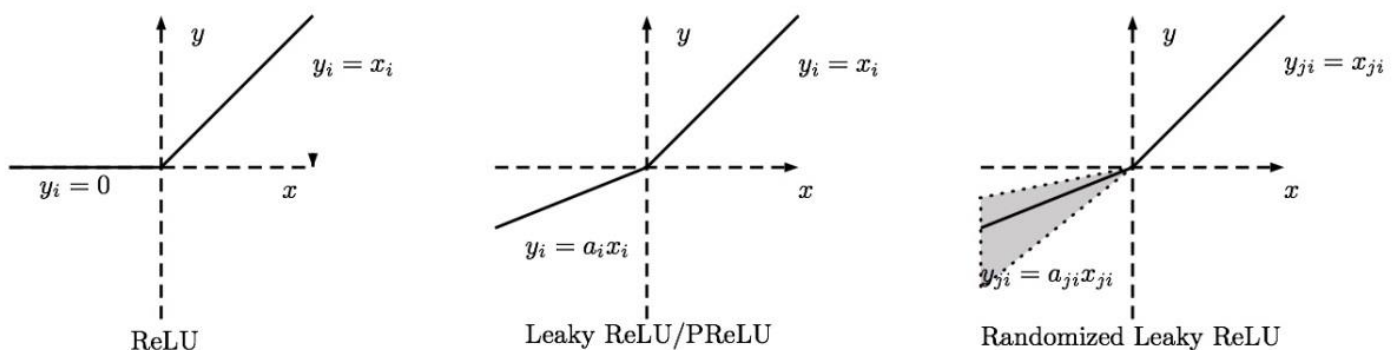


Рисунок 2.4 – Модифікації ReLU.

2.3.4. SoftMax

Дана функція використовується в якості функції активації вихідного (L -го) шару [60]

$$y_j^L = f_k^L(\sum_{k=1}^{N^{L-1}} w_{ik}^L x_k + b_j^L), \quad (2.21)$$

і описується виразом

$$f_j^L = \frac{e^{x_j^L}}{\sum_{i=1}^{N^L} e^{x_i^L}}. \quad (2.22)$$

Перевага функції SoftMax в тому, що вона використовує експоненту. За рахунок цього вирішується проблема з негативними вихідними значеннями, так як експонента завжди позитивна. Крім того, експонента значно збільшує великі значення.

Функцію активації SoftMax зручно застосовувати для задач класифікації, так як вона дозволяє трактувати вихідні значення нейронів як ймовірність приналежності даного класу, а також забезпечує, щоб тільки одне вихідне значення було близьке до одиниці за рахунок застосування експоненти.

Слід зазначити, що немає теоретичної необхідності використовувати саме SoftMax для задач класифікації. При наявності достатнього обсягу даних нейронна мережа може навчитися з іншими функціями активації на останньому шарі. Але з функцією м'якого максимуму навчання відбудеться швидше, тому що дана функція добре підходить саме для класифікації з непересічними класами.

Для нейронних мереж з SoftMax труднощі представляє знаходження її похідної. Під час переходу до згортальних і поворотних нейронних мереж для обчислення градієнтів можна використовувати спеціальні бібліотеки, такі як Theano і TensorFlow.

2.4. Навчання персептрона

2.4.1. Алгоритм зворотного поширення помилки

Термін «зворотне поширення» (ЗП) відноситься до процесу, за допомогою якого можуть бути обчислені похідні функціоналу помилки за параметрами мережі. Цей процес може використовуватися в поєднанні з різними стратегіями оптимізації [80].

У стандартній реалізації алгоритму ЗП для мінімізації цільової функції (функції помилки) використовується градієнтний метод пошуку (заснований на локальній лінійній апроксимації цільової функції), згідно з яким на кожній ітерації k обчислюється градієнт функціоналу, що мінімізується $\nabla f(w(k))$ (2.8) та коригуються мережеві ваги за правилом

$$\begin{cases} w(k+1) = w(k) + \Delta w(k); \\ \Delta w(k) = -\eta \nabla f(k), \eta > 0, \end{cases} \quad (2.23)$$

де η - параметр швидкості навчання, зазвичай встановлюється в діапазоні (0,1)

$$\nabla f(k) = \left(\frac{\partial E(w(k))}{\partial w_1(k)} \quad \frac{\partial E(w(k))}{\partial w_2(k)} \quad \dots \quad \frac{\partial E(w(k))}{\partial w_N(k)} \right)^T.$$

Правило функціонування вихідного шару L записується так (2.21):

$$y_{k,p}^L = f^L \left(\sum_{l=0}^{N^{L-1}} w_{kl}^L y_{l,p}^{L-1} \right), \quad k = \overline{1, N^L}.$$

При використанні квадратичної функції помилки $E(w(k))$ вираз для градієнта критерію якості функціонування для нейронів вихідного шару буде мати наступний вигляд:

$$\begin{aligned} \frac{\partial E}{\partial w_{ik}^L} &= \sum_{p=1}^P \sum_{k=1}^{N^L} \frac{\partial E}{\partial y_{k,p}^L} \frac{\partial y_{k,p}^L}{\partial f_{k,p}^L} \frac{\partial f_{k,p}^L}{\partial w_{ik}^L} = \frac{\partial}{\partial w_{ik}^L} \left(\frac{1}{2} \sum_{p=1}^P \sum_{k=1}^{N^L} (d_{k,p} - y_{k,p}^L)^2 \right) = \\ &= - \sum_{p=1}^P \sum_{k=1}^{N^L} (d_{k,p} - y_{k,p}^L) \frac{\partial y_{k,p}^L}{\partial w_{ik}^L}, \end{aligned} \quad (2.24)$$

а узагальнена формула (узагальнене дельта правило) має вигляд:

$$\frac{\partial E}{\partial w_{ij}^r} = - \sum_{p=1}^P \delta_{i,p}^{(r-1)} y_{j,p}^{(r-1)}, \quad r = \overline{1, L}, \quad i = \overline{1, N^r}, \quad j = \overline{0, N^{r-1}}, \quad (2.25)$$

де r – номер шару, а $\delta_{i,p}^r$ – локальна помилка шару r

$$\begin{aligned} \delta_{i,p}^r &= f'^r(f_{i,p}^r) \sum_{k=1}^{N^{r+1}} \delta_{k,p}^{r+1} w_{ki}^{r+1}, & r = \overline{1, L-1}; \\ \delta_{i,p}^L &= (d_{i,p} - y_{i,p}^L) f'^L(f_{i,p}^L), & i = \overline{1, N^L}. \end{aligned}$$

Однак головний недолік градієнтного методу - його повільну збіжність для погано обумовлених завдань - такими коштами усунути не вдається.

На рис.2.5 представлена схема налаштування ваг відповідно до алгоритму ЗП для тришарового персептрона містить N^0 входів, N^1 нейронів в першому шарі, N^2 нейронів у другому шарі і N^3 нейронів у вихідному шарі.

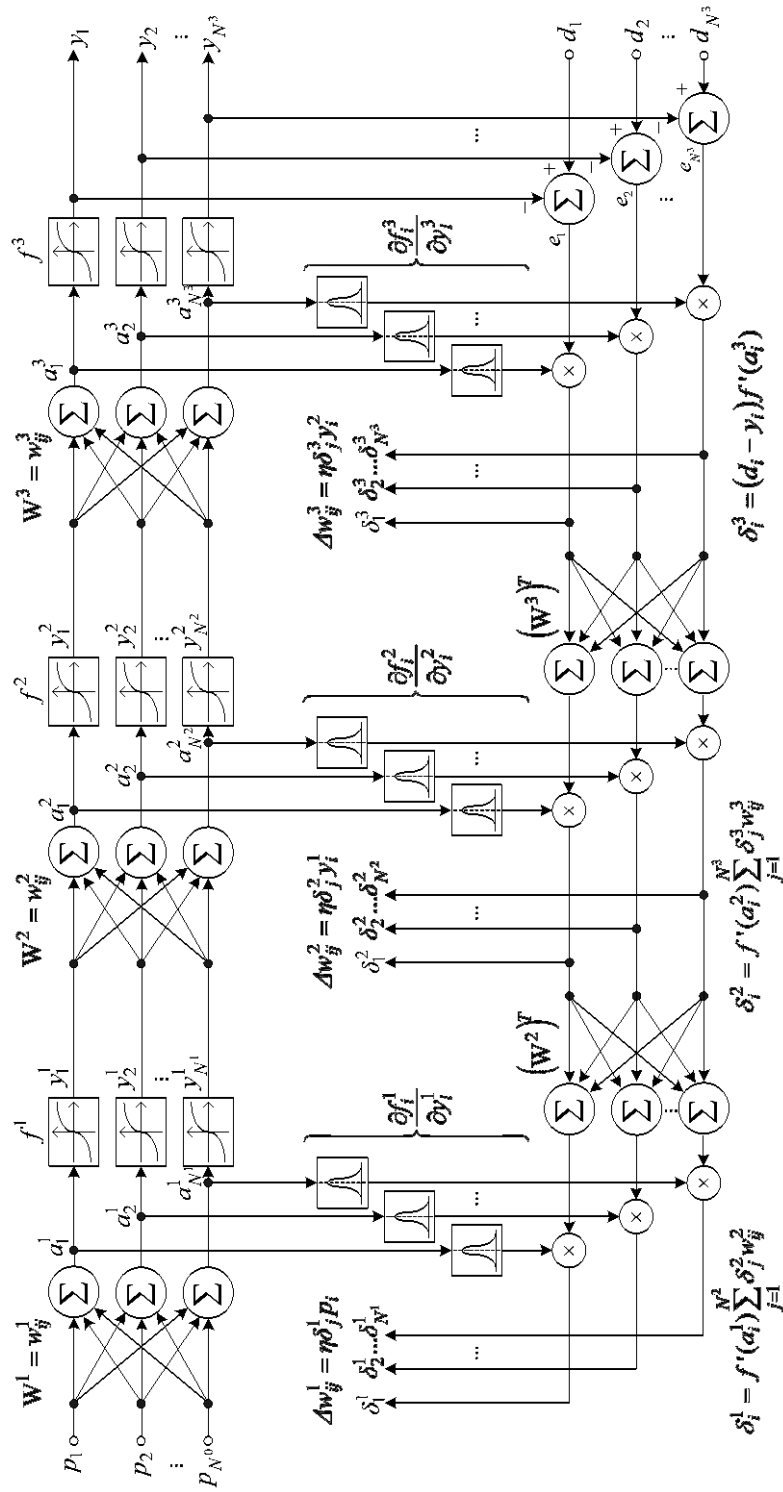


Рисунок.2.5- Структура тришарового перцептрона під час реалізації алгоритму ЗП

2.4.2. Метод Ньютона

В градієнтному методі основою є ідея локальної лінійної апроксимації цільової функції, що мінімізується. Якщо ж цільова функція $f(w(k))$ двічі диференційована, то ефективніше використовувати її квадратичну апроксимацію в точці $w(0)$ [55]. Ітераційна процедура пошуку мінімуму (метод Ньютона) має вигляд:

$$w(k+1) = w(k) - G^{-1}(k) \nabla f(k), \quad (2.26)$$

де $G(k)$ - матриця Гаусса.

Для квадратичної функції метод Ньютона збігається за один крок. Чим ближче цільова функція до квадратичної, тим швидше збігається метод Ньютона.

Для навчання нейронних мереж застосування методу Ньютона, заснованого на виразі (2.26), недоцільно з огляду на те, що, по-перше, цей метод можна використовувати тільки, якщо матриця Гессе позитивно визначена; по-друге, він вимагає істотних обчислювальних витрат і значного обсягу пам'яті для зберігання даних; по-третє, при його використанні необхідні аналітичні вирази для перших і других похідних в кожній точці $w(k)$.

Використання різних стратегій для регулювання позитивної визначеності матриці Гессе і для вибору прийнятною довжини кроку на кожній ітерації дозволяє модифікувати метод Ньютона таким чином, щоб надати йому властивість глобальної збіжності і знизити його обчислювальні витрати.

2.4.3. Метод Левенберга-Марквардта

Вельми ефективною є модифікація методу Ньютона, що називається методом Левенберга-Марквардта [61,62], в якій напрям пошуку відрізняється від заданого методом Ньютона. В даному методі до апроксимуючої функції додається квадратичний штраф за відхилення від точки, $w(k)$, а нова точка $w(k+1)$ визначається з умови мінімуму функції

$$f_k(w) + \alpha(k) \|w - w(k)\|^2 / 2,$$

де $f_k(w) = f(w(k)) + (w - w(k))^T \nabla f(w(k)) + 0,5(w - w(k))^T G(k)(w - w(k))$,

що призводить до методу

$$w(k+1) = w(k) - [G(k) + \alpha(k)I]^{-1} \nabla f(k). \quad (2.27)$$

За умови $\alpha(k) = 0$ метод переходить в метод Ньютона, за умови $\alpha(k) \rightarrow \infty$ напрямок пошуку прагне до антиградієнта. Таким чином, метод Левенберга-Марквардта (2.27) являє собою компроміс між цими двома методами. За рахунок вибору $\alpha(k)$ можна домогтися глобальної збіжності методу.

Однак в розглянутих алгоритмах кожна ітерація вимагає великих обчислювальних витрат (обчислення других похідних, рішення систем лінійних рівнянь), а швидкість збіжності удалині від мінімуму не висока.

2.4.4. Квазіньютонівські методи

В основі цих методів лежить ідея відновлення квадратичної апроксимації цільової функції за значеннями її градієнтів у ряді точок. Тим самим методи об'єднують переваги градієнтного методу (не потрібно обчислення матриці других похідних) і методу Ньютона (швидка збіжність внаслідок використання квадратичної апроксимації).

Ці методи мають загальну структуру [62]

$$w(k+1) = w(k) - \alpha(k)H(k)\nabla f(k), \quad (2.28)$$

де матриця $H(k)$ перераховується рекурентним способом на основі інформації, отриманої на k -й ітерації, так що $H(k) - G^{-1}(k) \rightarrow 0$.

Основна ідея квазіньютонівських методів полягає в побудові апроксимації безпосередньо для оберненої матриці $G^{-1}(k)$.

Найбільш популярні такі формули для перерахунку матриці $H(k+1)$:

а) метод Давидона-Флетчера-Пауелла (ДФП)

$$H(k+1) = H(k) - \frac{H(k)p(k)p^T(k)H(k)}{p^T(k)H(k)p(k)} + \frac{s(k)s^T(k)}{p^T(k)s(k)}, \quad H(0) > 0;$$

б) метод Бroyдена

$$H(k+1) = H(k) - \frac{(s(k) - H(k)p(k))(s(k) - H(k)p(k))^T}{p^T(k)(s(k) - H(k)p(k))}, \quad H(0) > 0;$$

в) метод Бroyдена-Флетчера-Шенно (БФШ)

$$H(k+1) = H(k) + \left(1 + \frac{p^T(k)H(k)p(k)}{s^T(k)p(k)} \right) \frac{s(k)s^T(k)}{s^T(k)p(k)} - \frac{s(k)p^T(k)H(k) + H(k)p(k)s^T(k)}{s^T(k)p(k)}, \quad H(0) > 0,$$

де $p(k) = \nabla f(k+1) - \nabla f(k)$; $s(k) = w(k+1) - w(k)$.

Незважаючи на відмінність цих формул, для квадратичних функцій послідовності $w(k)$, що генеруються кожним варіантом методу, збігаються.

Головний недолік квазіньютонівських методів в порівнянні, наприклад, з методом сполучених градієнтів полягає в необхідності зберігати і перераховувати матрицю $H(k)$ розмірності $n \times n$, що для великих n вимагає великого обсягу пам'яті і виконання великого числа арифметичних операцій.

2.4.5. Методи сполучених градієнтів

У градієнтному методі на кожному кроці не використовується інформація, отримана на попередніх ітераціях. Методи, в яких нове наближення залежить від s попередніх $w(k+1) = \varphi(w(k), \dots, w(k-s+1))$, називаються s -кроковими. Градієнтний метод і метод Ньютона–однокрокові методи. Методи сполучених градієнтів відносяться до багатокрокових методів і, на відміну від методів Ньютона, не вимагають зберігання матриці Гессе. Ці методи засновані на тому, що послідовність взаємно пов'язаних напрямків пошуку, яка задовольнить умові $t(k) = -H^{-1}(k)\nabla f(k)$, може бути знайдена ітераційно, без використання матриці Гессе, відповідно до правила

$$t(k+1) = -\nabla f(k+1) + \beta(k)t(k), \quad (2.29)$$

з початковою установкою напрямку пошуку $t(0) = -\nabla f(0)$.

Тоді процес пошуку мінімуму цільової функції набирає вигляду: $w(k+1) = w(k) + \alpha(k)t(k)$, де $\alpha(k) = \alpha$, а α - значення, яке мінімізує функцію $E(w(k) + \alpha t(k))$.

Відмінності методів сполучених градієнтів полягають в способі обчислення $\beta(k)$. Найбільш відомими з них є [62]: метод Флетчера-Рівса $\beta(k) = \nabla f(k+1)^T \nabla f(k+1) / \nabla f(k)^T \nabla f(k)$ і метод Полака-Райбера $\beta(k) = (\nabla f(k+1) - \nabla f(k))^T \nabla f(k+1) / \nabla f(k)^T \nabla f(k)$.

Оптимізатори навчання прискорюють процес знаходження параметрів мережі шляхом адаптації швидкості навчання під кожен з параметрів окремо. Деякі з відомих варіантів оптимізації – шляхом додавання моменту в стандартний стохастичний градієнтний спуск. Хоча оптимізатори можуть прискорити або навіть забезпечити отримання оптимальної моделі, результати досліджень показують неочевидність вигоди від їх використання. Зокрема, результати досліджень показали погіршення узагальнення моделі на нові дані при використанні оптимізаторів в порівнянні зі звичайним стохастичним градієнтним спуском [63-65].

Регуляризація (згасання ваг) – додавання в loss-функцію виразу, що залежить від ваг нейронів, з деяким коефіцієнтом. Це не дозволяє ШНМ перенавчитися на тренувальних даних за допомогою обмеження їх можливості рости до великих значень.

2.4.6. Матрична процедура навчання БП

Розглянемо навчання багатошарового персептрона (БП) з M входами і S виходами, L шарами, числом нейронів вихідного шару P , прихованого – N , всі нейрони якого мають активаційну функцію виду $f(x) = \frac{1}{1 + e^{-\alpha x}}$.

Введемо наступні позначення:

$W^{*L}(k) = (W_1^{*L}(k), W_2^{*L}(k), \dots, W_p^{*L}(k))^T$ – матриця оптимальних ваг вихідного шару $P \times L$;

$w_i^{*L}(k) = (w_{i1}^{*L}(k), w_{i2}^{*L}(k), \dots, w_{is}^{*L}(k))^T$ – вектор оптимальних ваг i -го нейрону вихідного шару $P \times 1$;

$W^L(k) = (W_1^L(k), W_2^L(k), \dots, W_p^L(k))^T$ – матриця оцінок ваг вихідного шару БП $P \times S$;

$w_i^L(k) = (w_{i1}^L(k), w_{i2}^L(k), \dots, w_{ip}^L(k))^T$ – вектор оцінок ваг i -го нейрона вихідного шару $P \times 1$;

$W^{*i}(k) = (W_1^{*i}(k), W_2^{*i}(k), \dots, W_p^{*i}(k))^T$ – матриця оптимальних ваг нейронів i -го прихованого шару ($i \neq L$) $N \times P$;

$w_i^{*i}(k) = (w_{i1}^{*i}(k), w_{i2}^{*i}(k), \dots, w_{in}^{*i}(k))^T$ – вектор оптимальних ваг нейронів i -го прихованого шару $N \times 1$;

$F^L(x, w^L) = (f_1^L(w_1^{TL}(k)x(k)) \quad f_2^L(w_2^{TL}(k)x(k)) \quad \dots \quad f_L^L(w_L^{TL}(k)x(k)))^T$ – вектор активаційних функцій нейронів вихідного шару $L \times 1$;

$\nabla f(x, w) = \text{diag} \left[\nabla f(w_1^T(k), x(k)) \quad \nabla f(w_2^T(k), x(k)) \quad \dots \quad \nabla f(w_L^T(k), x(k)) \right]$ – матриця похідних активаційних функцій шару $L \times L$;

$\nabla f(w_i^T(k), x(k))$ – перша похідна активаційної функції i -го нейрону.

Тоді вихідні сигнали БП і помилка апроксимації можуть бути представлені таким чином [66]:

– необхідний вектор вихідних сигналів

$$d(k) = W^{*L}(k)F^L(x, w^{*L}(k)) + \xi_1(k) \quad (2.30)$$

– реальний вектор вихідних сигналів

$$y(k) = W^L(k)F^L(x, w^L(k)) + \xi_2(k) \quad (2.31)$$

– помилка

$$e(k) = d(k) - y(k) + \xi(k) = \tilde{\theta}^T(k) F^L(x, w(k)) + W^{*L}(k)(k) \tilde{F}^L(x, w(k)) + \xi_1(k) + \xi_2(k), \quad (2.32)$$

де $\tilde{\theta}(k) = W^{*L}(k) - W^L(k)$ – матриця помилок навчання $P \times S$, $\tilde{F}^L(k) = F^L(x, w^{*L}(k)) - F^L(x, w^L(k))$ – вектор помилок активаційних функцій, $\xi_1(k)$, $\xi_2(k)$ – вектор помилок вихідного сигналу і нейромережевої апроксимації відповідно.

З урахуванням цих позначень градієнтні процедури навчання матриць ваг вихідного і i -го прихованого шару БП, які мінімізують квадратичний функціонал і представляють собою процедуру зворотного поширення помилки, можуть бути записані таким чином:

$$\begin{aligned} W^L(k) &= W^L(k-1) - \gamma_{w^L}(k) \frac{1}{\|F^L(x, w^L(k-1))\|^2} \frac{\partial E(k)}{\partial W^L(k-1)} = \\ &= W^L(k-1) + \gamma_{w^L}(k) \frac{F^L(x, w^L(k-1)) e^T(k)}{\|F^L(x, w^L(k-1))\|^2}; \end{aligned} \quad (2.33)$$

$$\begin{aligned} W^i(k) &= W^i(k-1) - \gamma_{w^i}(k) \frac{1}{\|\nabla f(x(k), W^i(k-1)) x(k)\|^2} \frac{\partial E(k)}{\partial W^i(k-1)} = \\ &= W^i(k-1) + \gamma_{w^i}(k) \frac{\nabla f(x(k), w^i(k-1)) e^T(k) x^T(k)}{\|\nabla f(x(k), W^i(k-1)) x(k)\|^2}, i \neq L, \end{aligned} \quad (2.34)$$

являє собою матричні варіанти процедури Качмажа (Уїдроу-Хоффа).

Тут $\gamma_{w^L}(k)$; $\gamma_{w^i}(k)$, $i \neq L$ – коефіцієнти навчання.

Дослідження питань збіжності даних процедур труднощів не викликає і здійснюється стандартними способами.

Для підвищення обчислювальної стійкості (2.33), (2.34) вони можуть бути модифіковані, тобто регуляризовані

$$W^L(k) = W^L(k-1) + \gamma_{w^L}(k) \frac{F^L(x, w^L(k-1))e^T(k)}{\beta_{w^L}(k) + \|F^L(x, w^L(k-1))\|^2}; \quad (2.35)$$

$$W^i(k) = W^i(k-1) + \gamma_{w^i}(k) \frac{\nabla f(x(k), w^i(k-1))e(k)x^T(k)}{\beta_{w^i}(k) + \|\nabla f(x, W^i(k-1))x(k)\|^2}, i \neq L, \quad (2.36)$$

де $\beta_{w^L}(k)$, $\beta_{w^i}(k)$, $i \neq L$ – параметри регуляризації.

Слід, однак, відзначити, що якщо статистичні властивості перешкод $\xi_1(k)$ і $\xi_2(k)$ не відомі, а відомо, що вони обмежені за амплітудою, тобто

$$|\xi(k)| < \delta \quad (2.37)$$

то процедури (2.31) - (2.34) слід модифікувати шляхом використання в них зон нечутливості, наприклад, наступним чином:

$$W^L(k) = W^L(k-1) + \gamma_{w^L}(k) \frac{\alpha_{w^L}(k)F^L(x, w^L(k-1))e^T(k)}{\beta_{w^L}(k) + \|F^L(x, w^L(k-1))\|^2}; \quad (2.38)$$

$$\text{де } \alpha_{w^s}(k) = \begin{cases} g(e(k), \Delta_{w^s}(k)), & \text{если } \|e(k)\| > \Delta_{w^s}(k); \\ 0, & \text{если } \|e(k)\| \leq \Delta_{w^s}(k). \end{cases} \quad (2.39)$$

Так як при обчисленні оцінок матриці W використовується похідна функції активації, то параметр зони нечутливості повинен містити коефіцієнт α , який визначає нахил сигмоїд

$$W^i(k) = W^i(k-1) + \gamma_{w^i}(k) \frac{\alpha_{w^i}(k)\nabla f(x(k), W^i(k-1))e(k)x^T(k)}{\beta_{w^i}(k) + \|\nabla f(x, W^i(k-1))x(k)\|^2}, i \neq L, \quad (2.40)$$

$$\text{де } \alpha_{w^i}(k) = \begin{cases} g(e(k), \Delta_{w^i}(k)), & \text{если } \frac{\nabla f_{\min}}{\alpha} \|e(k)\| > \Delta_{w^i}(k); \\ 0, & \text{если } \frac{\nabla f_{\min}}{\alpha} \|e(k)\| \leq \Delta_{w^i}(k), \end{cases} \quad (2.41)$$

де $\nabla f_{\min} = \min[\nabla f_1(k), \nabla f_2(k), \dots, \nabla f_L(k)] > 0$.

Параметри зон нечутливості для процедур (2.38) і (2.39) може здійснюватися в такий спосіб:

$$\Delta_{w^L}(k) = \Delta_{w^L}(k-1) + \frac{\alpha_{w^L}(k) \|e(k-1)\|}{\beta_{w^L}(k-1) + \|F^L(x, w^L(k-1))\|^2}; \quad (2.42)$$

$$\Delta_{w^i}(k) = \Delta_{w^i}(k-1) + \frac{\alpha_{w^i}(k) \|e(k-1)\|}{\beta_{w^i}(k-1) + \|\nabla f(x, w^i(k-1)) x(k)\|^2}, i \neq L. \quad (2.43)$$

2.4.7. Робастне навчання багат шарового персептрона

Завдання розпізнавання в умовах перешкод з не гауссовськими розподілами, розглядалася в роботах [67-73]. У цих роботах використовувалася ідея регуляризації зображення з метою його реконструкції при наявності пуассоновського і імпульсного шумів. Для цієї мети застосовувався квазіньютонівський метод. Ступінь відповідності реконструйованого зображення реального зазвичай оцінюється евклідовою метрикою (L2-нормою), що є оптимальною в разі гауссовських перешкод.

Застосування квазіньютонівського методу з L2-нормою до моделі Рудіна-Ошер-Фатемі [73,74] забезпечує шумозаглушення зображення, проте при цьому неявно робиться припущення, що шум є гауссовським і не враховуються особливості пуассонівського шуму. Таким чином, при цьому забезпечується рівне шумозаглушення як більш сильного, так і слабкого шуму.

Принципово інший підхід заснований на застосуванні процедур, що мінімізують деякі не квадратичні функціонали. Використовувані для навчання БП процедури, засновані на методі найменших квадратів, тобто одержувані при виборі в (2.7) $\rho(e) = 0,5e^2$, що є оптимальними для гауссівського розподілу, будуть нестійкими при наявності перешкод вимірів, розподіли яких відмінні від нормального, так як $\rho'(e(i, \theta)) = e(i, \theta)$ зростає лінійно зі збільшенням $e(i, \theta)$.

В якості альтернативи, для забезпечення робастності цільову функцію модифікують таким чином, щоб обмежити вплив найбільших вимірів. Основним наслідком цього є, як правило, більш низька швидкість збіжності алгоритмів оптимізації. Це пояснюється тим, що дуже складно розрізнити в перший раз викиди і корисні вимірювання. У зв'язку з цим, деякі викиди можуть бути відфільтровані, що призводить до зменшення швидкості збіжності. У найбільш скрутному випадку невеликі, але зміщені вимірювання переміщують мінімум цільової функції.

Існує досить велика кількість функціоналів, що забезпечують отримання робастних М-оцінок (див., наприклад [75-77]), проте найбільш поширеними є комбіновані функціонали, запропоновані Хьюбером [75] і Хемпелем [78,79] і складаються з квадратичного, що забезпечує оптимальність оцінок для гауссівського розподілу, і модульного, що дозволяє отримати більш робастний до розподілів з важкими «хвостами» (викидами) оцінку. Дані функціонали і їх функції впливу ψ , які мають відповідно вид

$$\begin{aligned} \rho_1(e) &= \begin{cases} \frac{e^2}{2}, & |e| \leq c; \\ c|e| - \frac{c^2}{2}, & c < |e|; \end{cases} & \rho'_1(e) = \psi_1(e) &= \begin{cases} e, & |e| \leq c; \\ c \operatorname{sign}(e), & c < |e|; \end{cases} \\ \rho''_1(e) = \psi'(e) &= \begin{cases} 1, & |e| \leq c; \\ 0, & c < |e|; \end{cases} & \omega_1(e) &= \begin{cases} 1, & |e| \leq c; \\ \frac{c}{e}, & c < |e|; \end{cases} \end{aligned} \quad (2.45)$$

$$\rho_2'(e) = \psi_2(e) = \begin{cases} e, & 0 \leq |e| < b; \\ b \operatorname{sign}(e), & b \leq |e| < c; \\ \frac{b(|e| - d)}{c - d} \operatorname{sign}(e), & c \leq |e| < d; \\ 0, & d \leq |e|; \end{cases} \quad \rho_2''(e) = \psi_2'(e) = \begin{cases} 1, & 0 \leq |e| < b; \\ 0, & b \leq |e| < c; \\ \frac{b}{c - d}, & c \leq |e| < d; \\ 0, & d \leq |e|; \end{cases} \quad (2.46)$$

$$\omega_2(e) = \begin{cases} 1, & 0 \leq |e| < b; \\ \frac{b}{|e|}, & b \leq |e| < c; \\ \frac{b(|e| - d)}{(c - d)|e|}, & c \leq |e| < d; \\ 0, & d \leq |e|; \end{cases}$$

приведені на рис.2.6.

Зазвичай М-оцінки описуються шляхом завдання функції впливу ψ , а не функції ρ .

Функція ψ , запропонована Хьюбером, є монотонною, а ψ - функція Хемпела – немонотонною. Як зазначається в [75], при наявності в розподілі важких хвостів, застосування немонотонних ψ - функцій дозволяє поліпшити результати оцінювання.

Ефективність застосування цих функціоналів залежить від того, наскільки вдало вибрані константи a , b , c і d , що входять до них, які і визначають ступінь завадостійкості. У зазначених вище роботах рекомендується вибирати значення a з інтервалу $[\sigma, 2\sigma]$, де σ - стандартне відхилення спостереженнях, а значення b , c і d задавати рівними 1.5, 3.5 і 8 відповідно.

Таким чином, при наявності перешкод, розподіли яких відмінні від нормального, навчання персептрона слід використовувати градієнтну процедуру, в якій градієнт обчислюється відповідно до (2.8).

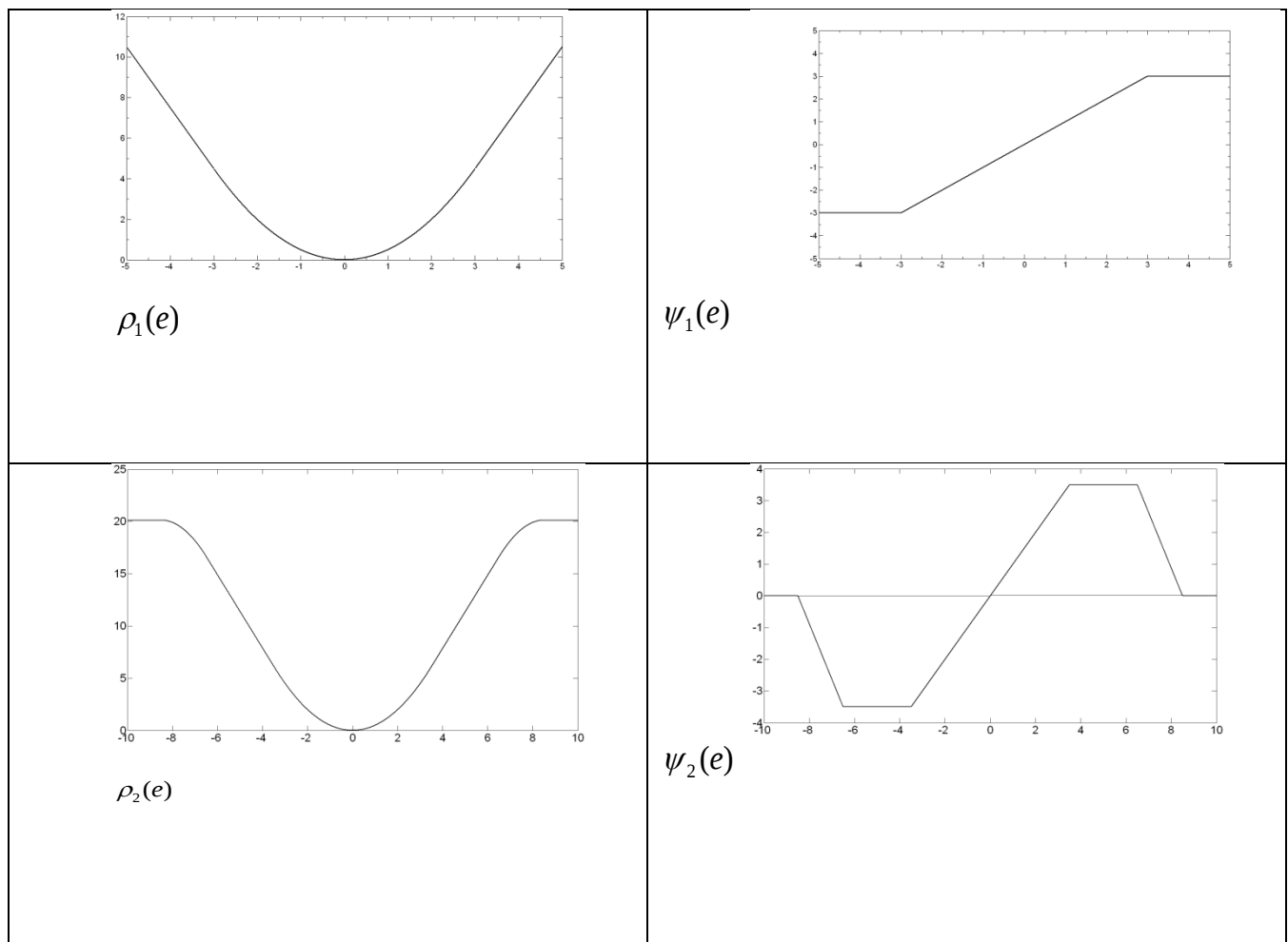


Рисунок 2.6 – Комбіновані функціонали Хьюбера (2.45) і Хемпела (2.46)

Відзначимо, що для забезпечення робастних властивостей одержуваних оцінок досить ефективним є застосування комбінованого функціоналу навчання [80]

$$F[e(k)] = \lambda e^2(k) + (1 - \lambda)|e(k)|, \quad (2.47)$$

де $\lambda \in [0, 1]$.

Градiєнтний алгоритм мінімізації (2.47) має вигляд

$$\theta(k) = \theta(k-1) + \gamma(k) [\lambda 2e(k) + (1-\lambda) \operatorname{sign} e(k)] x(k). \quad (2.48)$$

Питання збіжності процедури (2.48) розглянуті в Додатку А.

Даний алгоритм поєднує властивості МНК з властивостями МНМ, тому що при $\lambda = 1$ (2.48) впливає алгоритм МНК, а при $\lambda = 0$ – алгоритм МНМ, і дозволяє боротися з імпульсними перешкодами. Варіюючи параметр λ , можна змінювати властивості алгоритму.

2.5. Адаптація багат шарового персептрона на багатопроцесорні системи

2.5.1. Реалізація БП на багатопроцесорній системі

Час навчання нейронних мереж, особливо при використанні стандартних персональних комп'ютерів і великих обсягах вихідних даних, може бути дуже великий. Проблему високої обчислювальної складності навчання нейромереж найчастіше вирішують, використовуючи більш потужний комп'ютер або спеціалізований апаратний нейроприскорювач. Однак такий шлях доступний далеко не всім.

На сьогодні, внаслідок відсутності чітких інструкцій щодо адаптації нейронних мереж на високопродуктивні архітектури, стоїть завдання розробки методики та алгоритмів адаптації.

При розробці паралельного обчислювального пристрою - нейрокомп'ютера, здатного імітувати роботу нейромережі високої продуктивності, слід враховувати наступні фактори:

- необхідно розподіляти між процесорами нейрокомп'ютера (НК), синхронізуючи обробку нейронів відповідно до матриці проходження, тобто реалізувати спосіб розпаралелювання за інформацією;
- однакові для всіх процесорів програми одночасно оброблюваних нейронів в загальному випадку повинні виконуватися по різним гілкам[81].

Знизити обчислювальну складність нейромережевих алгоритмів можна шляхом розпаралелювання однотипних операцій, які виконуються в процесі функціонування нейронної мережі. Подібні підходи дозволяють навчати нейронні мережі з меншими витратами часу і дають можливість отримання максимального прискорення, що обмежується числом наявних процесорів [82,83].

В роботі для прискорення процесу навчання НМ, пропонується проводити розпаралелювання обчислень нейронів кожного шару з використанням технологій OpenMP і Cuda.

Технологія OpenMP є засобом програмування комп'ютерів із загальною пам'яттю, що базуються на традиційних мовах програмування. За основу береться послідовна програма. У OpenMP будь-який процес складається з декількох ниток управління, які мають загальний адресний простір, але різні потоки команд і роздільні стеки.

Сучасні графічні процесори (GPU) ефективні в роботі над операціями обчислення з плаваючою точкою, а також матричним множенням за рахунок розпаралелювання обчислень на багатопроцесорних системах з використанням безлічі потоків одночасно. Спочатку вони використовувалися тільки для вирішення графічних завдань, але в даний час знаходять все більше застосування в областях пов'язаних з машинним навчанням. Компанія Nvidia пропонує технологію Cuda, яка дозволяє писати програми для графічних процесорів високорівневої мовою програмування і задіяти GPU як процесори загального призначення. В основі архітектури CUDA лежить масштабований масив поточкових мультипроцесорів на базі SIMT архітектури. Такий мультипроцесор здатний обробляти паралельно сотні ниток.

Реалізація БП (рис.2.4) на багатопроцесорній системі може бути представлена схемою, що наведена на рисунку 2.7 [85-93].

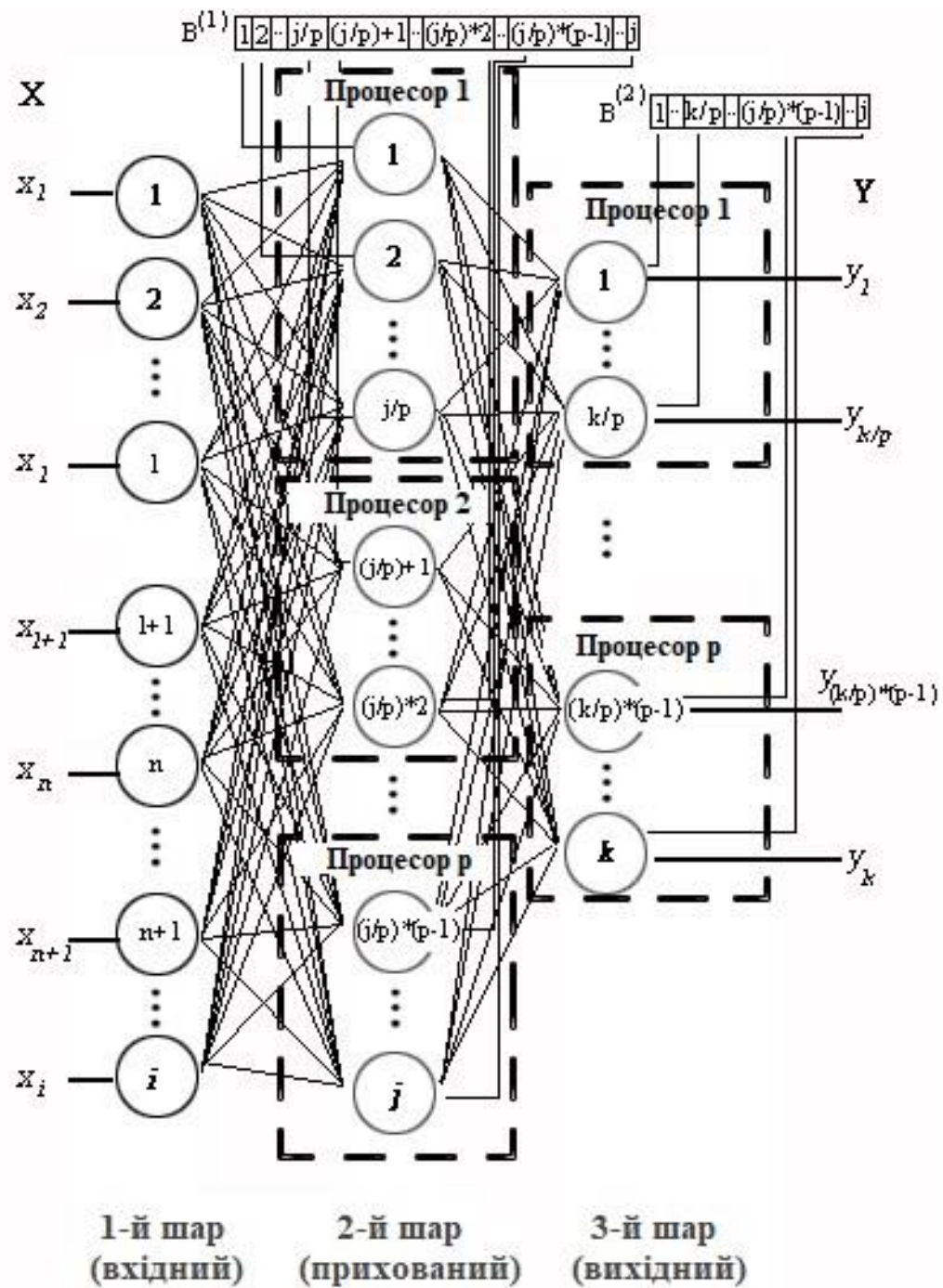


Рисунок 2.7 – Відображення БП на p – процесорну архітектуру

При такій реалізації співвідношення (1.2) можна представити в матричному вигляді як множення матриці вагових коефіцієнтів $W^{(1)}$ прихованого шару на вектор входу X :

$$\begin{aligned}
Y^{(1)} &= \varphi(W^{(1)T} X + B^{(1)}) = \phi \left(\begin{bmatrix} W_1^{(1)} \\ W_2^{(1)} \\ \dots \\ W_j^{(1)} \end{bmatrix} \times X + B^{(1)} \right) = \\
&= \varphi \left(\begin{bmatrix} w_{11}^{(1)} w_{21}^{(1)} \dots w_{i1}^{(1)} \\ w_{12}^{(1)} w_{22}^{(1)} \dots w_{i2}^{(1)} \\ \dots \\ w_{1j}^{(1)} w_{2j}^{(1)} \dots w_{ij}^{(1)} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_i \end{bmatrix} + \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \\ \dots \\ b_j^{(1)} \end{bmatrix} \right).
\end{aligned} \tag{2.49}$$

Аналогічно для другого шару маємо:

$$\begin{aligned}
Y^{(2)} &= \varphi(W^{(2)T} Y^{(1)} + B^{(2)}) = \phi \left(\begin{bmatrix} W_1^{(2)} \\ W_2^{(2)} \\ \dots \\ W_j^{(2)} \end{bmatrix} \times Y^{(1)} + B^{(2)} \right) = \\
&= \varphi \left(\begin{bmatrix} w_{11}^{(2)} w_{22}^{(2)} \dots w_{j1}^{(2)} \\ w_{12}^{(2)} w_{22}^{(2)} \dots w_{j2}^{(2)} \\ \dots \\ w_{1k}^{(2)} w_{2k}^{(2)} \dots w_{jk}^{(2)} \end{bmatrix} \times \begin{bmatrix} y_1^{(1)} \\ y_2^{(1)} \\ \dots \\ y_k^{(1)} \end{bmatrix} + \begin{bmatrix} b_1^{(2)} \\ b_2^{(2)} \\ \dots \\ b_k^{(2)} \end{bmatrix} \right).
\end{aligned} \tag{2.50}$$

Множення матриці вагових коефіцієнтів $W^{(2)}$ прихованого шару на вектор локальних градієнтів вихідного шару має вигляд:

$$\delta_j^{(1)} = \begin{bmatrix} \delta_1^{(2)} w_{11}^{(2)} \delta_2^{(2)} w_{12}^{(2)} \dots \delta_k^{(2)} w_{1k}^{(2)} \\ \delta_1^{(2)} w_{21}^{(2)} \delta_2^{(2)} w_{22}^{(2)} \dots \delta_k^{(2)} w_{2k}^{(2)} \\ \dots \\ \delta_1^{(2)} w_{j1}^{(2)} \delta_2^{(2)} w_{j2}^{(2)} \dots \delta_k^{(2)} w_{jk}^{(2)} \end{bmatrix} \times \begin{bmatrix} y_1^{(1)} \\ y_2^{(1)} \\ \dots \\ y_j^{(1)} \end{bmatrix} \tag{2.51}$$

Тим самим, отримання результуючого вектору $Y^{(2)}$ на кожному кроці навчання передбачає повторення $n = (2j + k)$ однотипних операцій по множенню рядків матриць на вектори. Отримання кожної такої операції включає по-елементне множення елементів рядка матриці на вектори і подальше складання отриманих добутків. Загальна кількість необхідних скалярних операцій необхідних для навчання НМ оцінюється величиною:

$$T_1 = 2n^2M, \quad (2.52)$$

де M – кількість епох, n – кількість однотипних операцій.

При використанні кількості процесорів $p \leq n$ паралельна обчислювальна схема множення матриці на вектор може бути конкретизована в такий спосіб:

- на кожен з наявних керуючих процесорів пересилається вектор X і $s = n/p$ рядків матриці;
- виконання операції множення рядків матриці на вектор виконується за допомогою послідовного алгоритму.

В основу моделей продуктивності паралельного алгоритму покладено співвідношення (2.53) (2.57).

Основний закон Амдала [94] полягає у тому, що прискорення залежить від потенційного паралелізму завдання і параметрів апаратури

$$S_p = T_1 / T_p, \quad (2.53)$$

де T_1, T_p – час виконання паралельного алгоритму на одному і p процесорах відповідно.

Закон Амдала, стосовно багатоядерним обчисленням відображає можливості багатоядерної системи

$$R_c = \frac{1}{S + (1+S)/n + H(n)}, \quad (2.54)$$

де S – час виконання послідовної частини коду, n – кількість процесорних ядер, $H(n)$ – накладні витрати.

Мережевий закон Амдала відображає втрати часу на міжпроцесорний обмін повідомленнями

$$R_c = \frac{1}{S + (1+S)/n + H(n)}, \quad (2.55)$$

де $a = W_{ск} / W$ – питома вага скалярних операцій, $W_{ск}$ – кількість скалярних операцій, що не розпаралелюються, M – кількість операцій, що виконуються паралельно, t – час виконання однієї операції, $c = \frac{W_c \cdot t_c}{W \cdot t} = c_A \cdot c_T$ – коефіцієнт мережевої деградації обчислень визначає обсяг обчислень, що припадає на одну передачу даних, W_c – кількість передач даних, t_c – час однієї передачі даних, c_A – алгоритмічна складова коефіцієнта деградації, c_T – технічна складова.

Ефективність паралельних обчислень [82] визначає реальну вигоду від використання багатопроцесорних систем в порівнянні з однопроцесорними

$$\varepsilon = S_p / p \quad (2.56)$$

Декомпозиція є основним принципом підвищення ефективності паралельних обчислень при вирішенні задач великої розмірності.

На основі обчислювальної схеми нейроалгоритму спільно з розкладом побудуємо модель продуктивності [92,93].

В якості незалежних операцій, що виконуються паралельно, представлені функції обчислення виходу нейрона, помилки і налаштування вагових коефіцієнтів, а скалярні операції для обчислення перерахованих функцій виконуються послідовно на окремих обчислювальних вузлах.

На прикладі налаштування вагових коефіцієнтів прихованого шару визначимо кількість:

- процесорів $P_{\max}$,
- скалярних $W_{ск}$ операцій, що не розпаралелюються,
- операцій, які виконуються паралельно W ,
- передач даних W_c .

Під час налаштування вагових коефіцієнтів $W_{ij}^{(1)T} = \begin{bmatrix} w_{01}^{(1)} w_{01}^{(1)} \dots w_{i1}^{(1)} \\ w_{02}^{(1)} w_{12}^{(1)} \dots w_{i2}^{(1)} \\ \dots \\ w_{0j}^{(1)} w_{1j}^{(1)} \dots w_{ij}^{(1)} \end{bmatrix}$ прихованого

шару обчислювальне навантаження обробки незалежних операцій, що виконуються паралельно на всіх обчислювальних вузлах, розподіляється на $P_{\max} = \overline{1, N_j}$ процесорах і потребує $N_i + 1$ тактів: на першому такті обчислюється вектор $W_0 = [w_{01}^{(1)} w_{02}^{(1)} w_{03}^{(1)} \dots, w_{0j}^{(1)}]$, де $w_{01}^{(1)} = b_1^{(1)}, \dots, w_{0j}^{(1)} = b_j^{(1)}$, на другому такті обчислюється вектор $W_1 = [w_{11}^{(1)} w_{12}^{(1)} w_{13}^{(1)} \dots, w_{1j}^{(1)}]$, на третьому такті обчислюється вектор $W_2 = [w_{21}^{(1)} w_{22}^{(1)} w_{23}^{(1)} \dots, w_{2j}^{(1)}]$, на $N_i + 1$ -му – $W_i = [w_{i1}^{(1)} w_{i2}^{(1)} w_{i3}^{(1)} \dots, w_{ij}^{(1)}]$. Кожен обчислювальний вузол окремо здійснює послідовну обробку скалярних операцій множення. На даному етапі кількість операцій, що виконуються паралельно, обчислюється відповідно до

$$W = N_j + N_i N_j \quad (2.57)$$

Таким чином, настройка вагових коефіцієнтів $w_{ij}^{(1)}$ прихованого шару потребує $P_{\max} = N_j$ процесорів, $W_{ск} = N_i + 2$ операцій, які не розпаралелюються, $W = N_j(N_i + 1)$, операцій, що виконуються паралельно, і $W_c = 2N_i N_j$ передач даних.

Тоді для реалізації алгоритму навчання кількість операцій, які не розпаралелюються, відповідає величині $W_{ск} = 2(2N_i + N_j + 2N_k + 4)$, кількість операцій, що виконуються паралельно $W = 2(N_i N_j + N_j N_k + N_j + N_k)$, кількість передач даних $W_c = 4(N_i N_j + N_j N_k - N_j + N_k - 1)$, максимально необхідну кількість процесорів $P_{\max} = \max N_j, N_k$.

Для побудови моделі продуктивності із застосуванням стандарту OpenMP враховується не тільки необхідність підтримки багатопоточності для рівномірного

розподілу потоків по ядрах процесора, але і накладні витрати, такі як інтерпотокова активність, синхронізація і ін.

2.5.2. Аналіз продуктивності і результати тестування розробленої системи

Для експериментальної оцінки обраних середовищ паралельного програмування використані дві гетерогенні одновузлові системи:

- IntelXeonE5-26502 v4 і IntelXeonPhi 7120P – 244 ядра з частотою 1.2 GHz.
- IntelXeonE5-26502 v4 і GeForceGTX 1070 GPU - 24 потокових мультипроцесора (Streaming Multiprocessors - SM), 1920CUDA ядер з частотою 1.5GHz.

Таблиця 2.1–Час виконання операції активації одного нейрона

Параметри зображення	Час операції активації виконання (мілісекунди)		
	Cuda	OpenMP	Без використання розпаралелювання (послідовне виконання)
32*32*3	0.7318	1.552	0.567
50*50*3	1.0478	2.278	0.781
100*100*3	1.2397	3.798	5.807
150*150*	1.4706	8.57	17.156
200*200*3	1.9689	21.233	43.535
250*250*3	2.3156	36.12	84.995
300*300*3	2.7751	49.135	130.65
350*350*3	2.7807	82.27	186.93
400*400*3	2.7703	132.59	287.991
450*450*3	2.7496	165.807	345.581
500*500*3	3.0862	208.514	468.78
550*550*3	3.1674	330.457	645.456

В таблиці представлено час, витрачений на виконання операції активації одного нейрона першого прихованого шару, без використання розпаралелювання (послідовні обчислення), а також з використанням багатоядерних CPU і GPU (технології OpenMP і Cuda).

На рисунку 2.8 подана залежність часу виконання операції множення від роздільної здатності зображення.

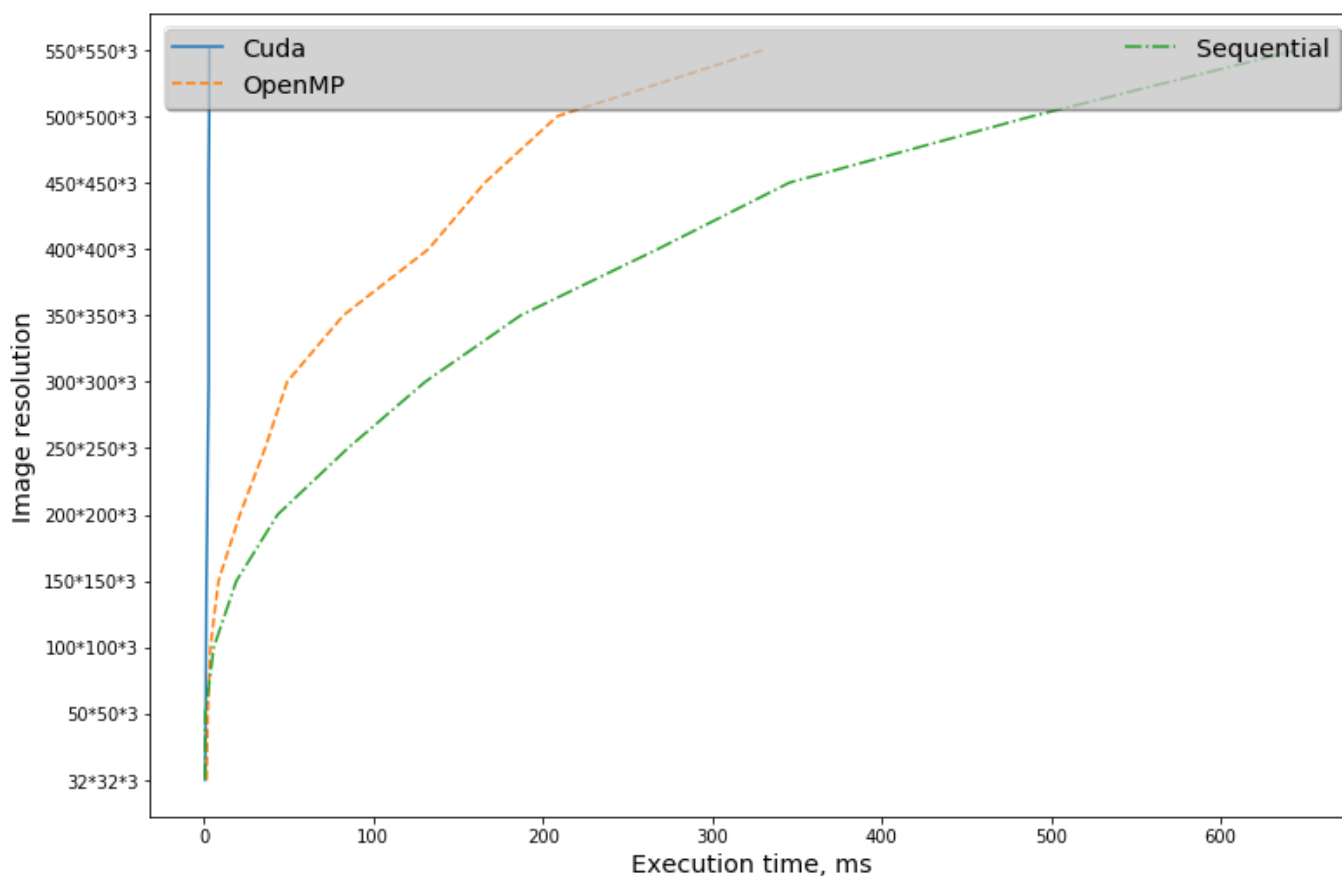


Рисунок 2.8 – Залежність часу виконання операції множення від роздільної здатності зображення

На данному рисунку час виконання операції множення представлено у вигляді лінійної залежності. Як можна помітити, для Cuda воно виражається практично вертикальною лінією. Це відбувається через те, що час виконання при послідовних обчисленнях зі збільшенням роздільної здатності зображення різко зростає –

від 0.567 мс для зображення $32 \times 32 \times 3$ до 645.456 мс для зображення $550 \times 550 \times 3$, а для Cuda немає від 0.7318 мс до 3.1674 мс відповідно.

Нижче представлена також залежність, однак для часу виконання запропоновано використовувати логарифмічну шкалу, що робить графік залежностей більш наочним.

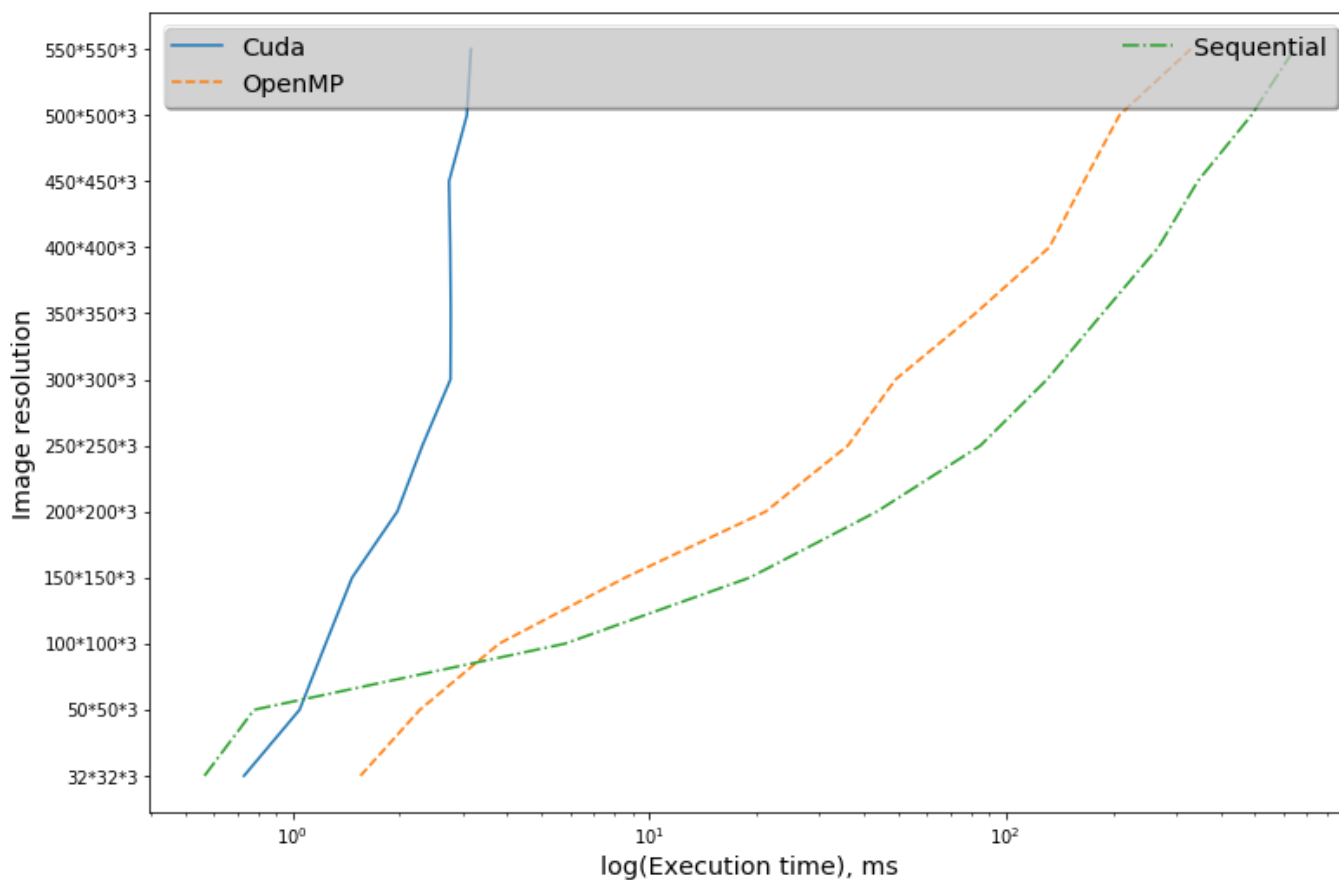


Рисунок 2.9 – Логарифмічна залежність часу виконання операції множення від роздільної здатності зображення

Як видно з проведених експериментальних досліджень, при роботі з зображеннями низької розмірності переваги паралельної обробки нівелюються накладними витратами, а використання OpenMP і Cuda стає раціональним при вирішенні 100×100 і більше.

На основі отриманих даних, було виконано розрахунок прискорення, отриманого в результаті розпаралелювання, результати якого наведені в таблиці 2.2.

Таблиця 2.2 – Прискорення, отримане за рахунок розпаралелювання

	Прискорення отримане в результаті розпаралелювання в порівнянні з послідовним виконанням	
Параметри зображення	Прискорення для Cuda	Прискорення для OpenMP
32*32*3	0.7748	0.3653
50*50*3	0.7453	0.3428
100*100*3	4.6841	1.5289
150*150*	11.6659	2.00185
200*200*3	22.1113	2.0503
250*250*3	36.7053	2.3531
300*300*3	47.0793	2.6590
350*350*3	67.2240	2.2721
400*400*3	103.9566	2.1720
450*450*3	125.6840	2.0842
500*500*3	151.8955	2.0514
550*550*3	203.7810	1.9532

Нижче представлені графіки залежності прискорення від розмірності зображення.

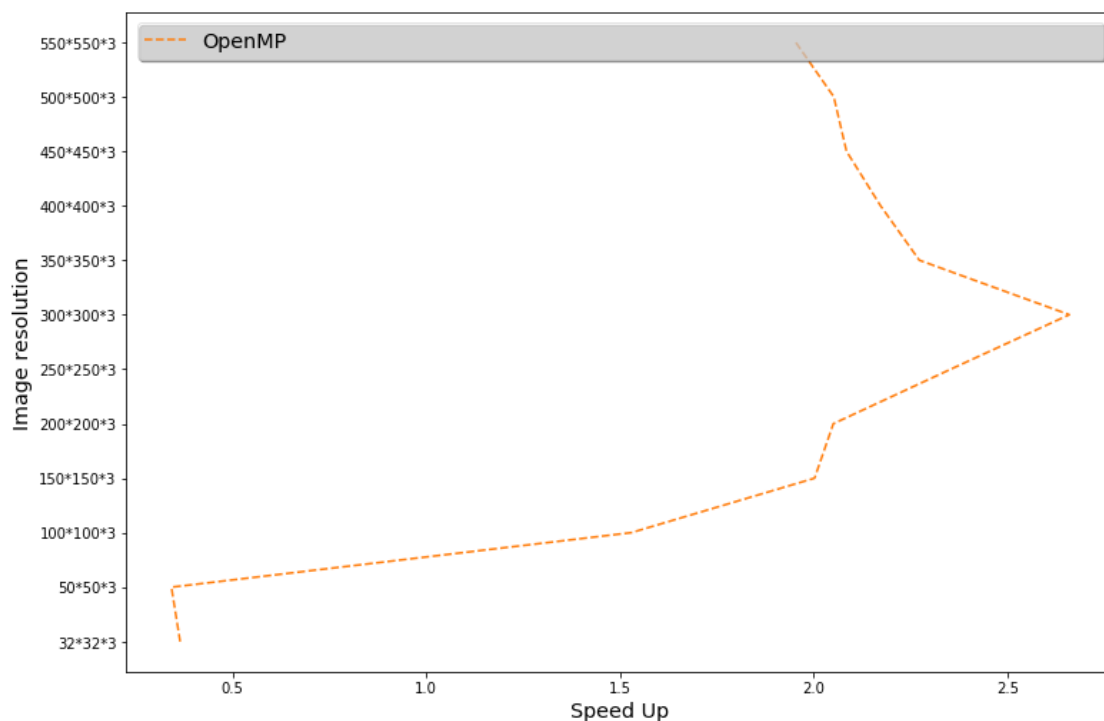


Рисунок 2.10 – Прискорення, отримане при використанні OpenMP

Як можна бачити з графіка (рис. 2.10) максимальне прискорення для OpenMP виходить при обробці зображення розмірністю 300*300*3. У той же час для Cuda прискорення безперервно зростає по всій кривій (рис. 2.11)

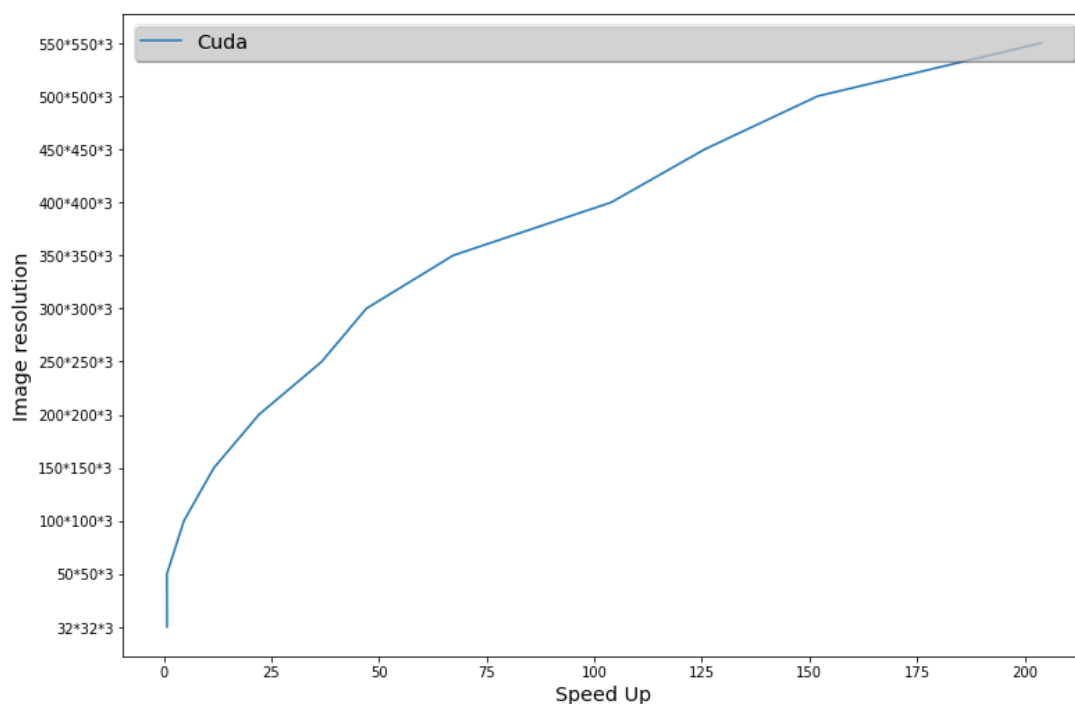


Рисунок 2.11 – Прискорення, отримане при використанні Cuda

Видно, що, хоча розпаралелювання з використанням CPU (OpenMP) і дає значний приріст в продуктивності, максимальне прискорення все ж досягається з використанням GPU (Cuda).

Таким чином, аналізуючи дані, наведені в таблицях можна отримати оцінку потенційно можливого виграшу в продуктивності в залежності від використовуваних ресурсів, а також розмірності досліджуваного зображення.

2.6. Висновки по розділу 2

Розглянуто задачу вибору критерію якості навчання БП, рішення якої має настільки ж істотний вплив на ефективність ШНМ, як і вибір алгоритму навчання. Описано загальний критерій, з якого випливають найбільш широко використовувані квадратична функція помилки і нормалізована її версія. Однак останнім часом дослідники все частіше використовують альтернативні функції помилки, що дозволяють поліпшити процес навчання мережі і забезпечити виведення з локального мінімуму, зокрема, функції перехресної ентропії.

2. Досліджено питання вибору функції активації нейронів. Проаналізовано властивості традиційних (сигмоїдальної і гіперболічного тангенса), а також функції ReLU і SoftMax, що останнім часом одержали широкого поширення. На відміну від традиційних дані функції не вимагають виконання ресурсномістких операцій. Крім того, функція ReLU може бути реалізована за допомогою простого порогового перетворення матриці активацій в нулі і не схильна до насичення. Функцію активації SoftMax зручно застосовувати для задач класифікації, так як вона дозволяє трактувати вихідні значення нейронів як ймовірність належності даного класу, а також забезпечує, щоб тільки одне вихідне значення було близьке до одиниці за рахунок застосування експоненти. Для обчислення градієнтів даної функції, які використовуються при навчанні, можна використовувати спеціальні бібліотеки, такі як Theano і TensorFlow.

3. Проведено аналіз алгоритмів навчання БП, заснованих на обчисленні градієнта функціоналу, який мінімізує. Показана ефективність використання методів другого порядку завдяки їх високій швидкості збіжності. Розроблено процедуру навчання БП, що представляє собою матричні варіанти процедури Качмажа (Уїдроу-Хоффа), запропонована її модифікація, яка містить зону нечутливості і розглянуті процедури налаштування цієї зони.

4. Методи навчання ШНМ, засновані на мінімізації квадратичних функціоналів, не є стійкими, якщо в вимірюваннях присутні викиди або перешкоди, мають розподіли відмінні від гауссівських. В якості альтернативи, для забезпечення робастності цільову функцію модифікують таким чином, щоб обмежити вплив найбільших вимірювань. Розглянуто найбільш поширені комбіновані функціонали Хьюбера і Хемпела, що складаються з квадратичного, що забезпечує оптимальність оцінок для гауссівського розподілу, і модульного, що дозволяє отримати більш робастний до розподілів з важкими «хвостами» (викидами) оцінку.

5. Розглянуто питання адаптації багат шарового перцептрона на однопроцесорні системи і проведено аналіз продуктивності та результатів тестування розробленої системи. В експерименті порівнювався час, витрачений на виконання операції активації одного нейрона першого прихованого шару без використання розпаралелювання (послідовні обчислення), а також з використанням багатоядерних CPU і GPU (технології OpenMP і Cuda). Результати свідчать про те, що, хоча розпаралелювання з використанням CPU (OpenMP) і дає значний приріст в продуктивності, максимальне прискорення все ж досягається з використанням GPU (Cuda). Аналіз результатів дозволяє отримати оцінку потенційно можливого виграшу в продуктивності в залежності від використовуваних ресурсів, а також розмірності досліджуваного зображення.

З ОБРОБКА ЗОБРАЖЕНЬ НА ОСНОВІ ЗНМ

Згорткові мережі є вдалою серединою між біологічно правдоподібними мережами і звичайним багатошаровим перцептроном. На сьогоднішній день кращі результати в розпізнаванні зображень отримують з їх допомогою. В середньому точність розпізнавання таких мереж перевершує звичайні ШНМ на 10-15%. ЗНМ - це ключова технологія Deep Learning [13,14].

Спочатку структура згортальної нейронної мережі створювалася з урахуванням особливостей будови деяких частин людського мозку, що відповідають за зір. В основу розробки таких мереж закладено три механізми:

- локальні сприйняття;
- формування шарів у вигляді набору карт ознак (колективні ваги);
- субдискретизація (підвибірка).

Під локальним сприйняттям мається на увазі, що на вхід нейрона надходить не все зображення, а тільки деяка його частина. Це дозволяє зберігати конфігурацію зображення при переході від шару до шару.

Ідея ваг, що розділяються, має на увазі, що до великої кількості зв'язків застосовується невеликий набір ваг, тобто кожна область зображення, на які воно розділене, буде оброблена одним і тим же набором ваг. При такому штучно створеному обмеженні ваг поліпшується властивість мережі до узагальнення.

Основною причиною успіху ЗНМ стала концепція загальних ваг. Незважаючи на великий розмір, ці мережі мають невелику кількість параметрів, що настроюються в порівнянні з їх предком – неокогнітроном. Є варіанти ЗНМ (Tiled Convolutional Neural Network), схожі на неокогнітрон, в таких мережах відбувається часткова відмова від пов'язаних ваг, але алгоритм навчання залишається тим же і ґрунтується на зворотному поширенні помилки.

Згорткові нейронні мережі забезпечують часткову стійкість до змін масштабу, зсувів, поворотам, зміні ракурсу і іншим спотворень. ЗНМ об'єднують три

архітектурні ідеї, для забезпечення інваріантності до зміни масштабу, повороту зрушення і просторовим спотворень:

- локальні рецепторні поля (забезпечують локальну двовимірну зв'язність нейронів);
- загальні вагові коефіцієнти синапсів (забезпечують детектування деяких рис в будь-якому місці зображення і зменшують загальне число вагових коефіцієнтів);
- ієрархічна організація з просторовими підвибірками.

На даний момент ЗНМ і її модифікації вважаються кращими по точності і швидкості алгоритмами знаходження об'єктів на сцені.

Згорткові нейронні мережі (ЗНМ, CNN) дуже схожі на звичайні нейронні мережі: вони також побудовані на основі нейронів, які володіють постійно змінюваною вагою і зсувами. Кожен нейрон отримує деякі вхідні дані, виконує скалярний добуток інформації і в окремих ситуаціях супроводжує це нелінійністю. Як і у випадку зі звичайними нейронними мережами, вся ЗНМ виражає одну диференційовану функцію внеску (ефективний внесок): з одного боку це необроблені пікселі зображення, з іншого - висновок класу або групи ймовірних класів, які характеризують картинку. Тут також присутня функція втрати на останньому (повністю підключеному) шарі.

Як відомо, нейронні мережі отримують вхідні дані (один вектор), після чого трансформують інформацію, проводячи її через ряд прихованих шарів. Кожен прихований шар складається з безлічі нейронів, де всякий нейрон має стійкий зв'язок з усіма нейронами в попередньому шарі і де нейрони в функції одного шару повністю незалежні один від одного і не мають спільних з'єднань. Останній повнозв'язний шар називається вихідним шаром, і в налаштуваннях класифікації він демонструє число класів.

Звичайні нейронні мережі погано масштабуються у випадку з зображеннями великих розмірів. Так, в системі комп'ютерного зору CIFAR-10, розмір картинки становить $[32 \times 32 \times 3]$ (32 - ширина, 32 - висота, 3 – канали кольору), тому один

повністю підключений нейрон в першому прихованому шарі звичайної нейронної мережі має вагу 3 072 ($32 \times 32 \times 3$). Здається, що це значення можна змінювати, але повнозв'язна структура не масштабується для великих зображень. Картинка з дозволом, наприклад, $[200 \times 200 \times 3]$, призведе до того, що повністю підключений нейрон буде важити 120 000. Крім того, ми майже напевно хотіли б мати кілька таких нейронів, що призвело б до додавання параметрів. Повна зв'язність – це марнотратство, і величезна кількість параметрів може швидко привести до перенавчання.

Згортальні нейронні мережі користуються тим, що вхідні дані складаються з зображень, і вони обмежують побудову мережі більш розумним шляхом. На відміну від звичайної нейронної мережі, шари CNN складаються з нейронів, розташованих в трьох вимірах: ширині, висоті і глибині, тобто вимірах, які формують обсяг.

3.1. Архітектура ЗНМ

Зазвичай сучасні завдання комп'ютерного зору поділяють на чотири види:

- **Classification** – класифікація зображення по типу об'єкта, яке воно містить;
- **Semantic segmentation** – визначення всіх пікселів об'єктів певного класу або фону на зображенні. Якщо кілька об'єктів одного класу перекриваються, їх пікселі ніяк не відокремлюються один від одного;
- **Object detection** – виявлення всіх об'єктів зазначених класів і визначення рамки, що охоплює, для кожного з них;
- **Instance segmentation** – визначення пікселів, що належать кожному об'єкту кожного класу окремо;

Визначення топології мережі орієнтується на вирішуване завдання, дані з наукових статей і власний експериментальний досвід.

Можна виділити наступні етапи, що впливають на вибір топології:

- визначити вирішуване завдання нейромережею (класифікація, прогнозування, модифікація);

- визначити обмеження в задачі, що розв'язується (швидкість, точність відповіді);
- визначити вхідні (тип: зображення, звук, розмір: 100×100 , 30×30 , формат: RGB, в градаціях сірого) і вихідних дані (кількість класів).

Архітектура згортальних нейромереж робить явне припущення виду «вхідні дані є зображення», що дозволяє закодувати певні властивості під архітектуру. На відміну від звичайної нейронної мережі, шари ЗНМ складаються з нейронів, розташованих в 3-х вимірах: ширині, висоті і глибині, тобто вимірах, які формують обсяг.

Як вже було сказано вище, схематично ЗНМ – це послідовність шарів. Кожен шар перетворює один активаційний об'єм в інший за допомогою функції, що диференціюється. Для організації згортальної нейронної мережі застосовується три основних шари:

- згортки,
- пулінгу (інакше підвибірки або субдискретизації),
- повнозв'язний шар.

Ці шари використовуються з метою побудови повної архітектури ЗНМ.

Розглянемо організацію ЗНМ більш детально, а простим прикладом згортальної нейромережі в контексті класифікації системи розпізнавання зображень CIFAR-10 може послужити архітектура [INPUT – CONV – RELU – POOL – FC].

В INPUT (вхідні дані) $[32 \times 32 \times 3]$ містяться вихідна інформація про зображення (в даному випадку 32 - ширина, 32 - висота, 3 – канали кольору R, G, B).

Шар CONV (шар згортки) примножує значення фільтра на вхідні значення пікселів зображення (по елементне множення), після чого всі ці множення підсумовуються. Кожна унікальна позиція введеного зображення виробляє число. Приклад: Якщо використовується 12 фільтрів, обсяг буде дорівнює $32 \times 32 \times 12$ $[32 \times 32 \times 12]$.

Шар RELU (блок лінійної ректифікації) застосовує поелементну функцію активації на кшталт $f(x) = \max(0, x)$, встановлюючи нульовий поріг. Іншими словами, RELU виконує наступні дії: якщо $x > 0$, то обсяг залишається колишнім ($[32 \times 32 \times 12]$), а якщо $x < 0$, то відсікаються непотрібні деталі в каналі шляхом заміни на 0.

Шар POOL (шар пулінгу) виконує операцію по зменшенню дискретизації просторових розмірів (ширина і висота), в результаті чого об'єм може скоротитися до $[16 \times 16 \times 12]$, тобто на цьому етапі виконується нелінійне ущільнення карти ознак. Логіка роботи така: якщо на попередній операції згортки вже були виявлені деякі ознаки, то для подальшої обробки необхідності в такому докладному зображенні не має, і воно ущільнюється до менш докладної картинки.

Шар FC (повнозв'язний шар) виводить N-мірний вектор (N – число класів) для визначення потрібного класу. Робота організовується шляхом звернення до виходу попереднього шару (карті ознак) і визначення властивостей, які найбільш характерні для певного класу.

Таким чином, у найпростішому випадку архітектура ЗНМ – це набір шарів, які перетворюють образ зображення в вихідний образ (наприклад, визначення класу).

Кожен шар відповідає за певний етап процесу обробки зображення (шар згортки, лінійної ректифікації, пулінгу і повнозв'язний шар найпопулярніші на сьогоднішній день).

Кожен шар отримує на вході об'ємну 3D інформацію і трансформує зі збереженням 3D-об'єму за допомогою функції, що диференціюється.

Шар може не мати параметрів (CONV і FC мають, RELU і POOL - немає).

Шар може не мати (наприклад, CONV, FC і POOL мають, RELU - немає).

Для того щоб отримати можливість належності до якогось класу, потрібно використовувати крос-ентропію як функцію помилки і на вихідному шарі - SoftMax, як функцію активації.

Архітектура ЗНМ наведена на рис.3.1

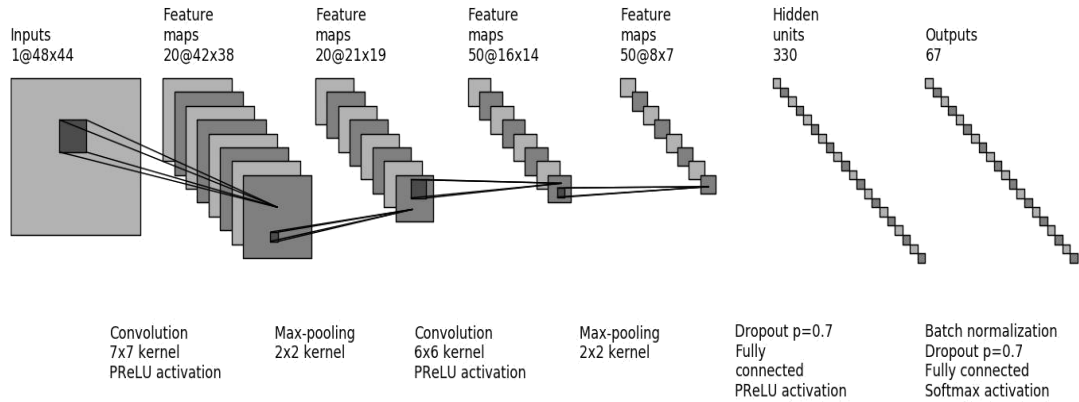


Рисунок 3.1 – Архітектура ЗНМ

Як видно з рисунку, крім описаних вище шарів, вона містить також і інші.

3.1.1 Шар підвибірки (пулінгу, MAX - pooling)

В архітектурі ЗНМ звичайною практикою є вставка шару пулінгу (підвибірки) між послідовностями згортальних шарів. Його функція полягає в поступовому зменшенні просторових габаритів зображення з метою зменшення кількості параметрів і обчислень в мережі, а також контролю перенавчання. Шар пулінгу працює незалежно від зрізу глибини вхідних даних і масштабує обсяг просторово, використовуючи функцію максимуму. Найчастіше використовується шар з фільтрами розміру 2×2 з кроком 2; подібний шар знижує дискретизацію кожного зрізу глибини входу в 2 рази як по ширині, так і по висоті, відкидаючи при цьому 75% активації. Кожна операція MAX в цьому випадку буде вибирати максимальне значення з 4 чисел. Розмір по глибині при цьому залишається незмінним.

На додаток до максимальної підвибірки, шари пулінгу можуть виконувати інші функції, наприклад, усереднюючу підвибірку або навіть L2-нормовану підвибірку. Історично усереднююча підвибірка використовувалася досить часто, але останнім часом вона відійшла на другий план у порівнянні з максимізаційною підвибіркою, яка на практиці працює краще.

У деяких роботах, наприклад, в [95] відзначається, що даний шар може бути виключений, і архітектура ЗНМ може складатися виключно з повторюваних шарів

згортки. Відмова від субдискретизації може зіграти важливу роль в навчанні генеративних моделей на зразок варіаційних автоасоціаторів або генеративних змагальних мереж. Звісно ж можливо, що вже в найближчому майбутньому архітектура типової ЗНМ буде позбавлена шарів пулінгу, або вони будуть представлені в дуже малій кількості.

3.1.2. Нелінійний шар активації

На даних шарах всередині мережі до всіх значень входу застосовується нелінійна функція активації, і результат подається на вихід. Таким чином, шар активації не змінює розмір входу. Найбільш популярними нелінійними функціями активації є ReLU ($\text{ReLU}(x) = \max(0; x)$) і різні її модифікації, розглянуті в другому розділі. Такий підхід дозволяє поліпшити результати роботи мережі за рахунок підбору оптимальної функції активації для кожного нейрона.

3.1.3. Дропаут шар.

Dropoute простим і ефективним методом регуляризації і полягає в тому, що в процесі навчання мережі з її сукупної топології багаторазово випадковим чином виділяється підмережа, тобто частина нейронів вимикається з процесу, і чергове оновлення ваг відбувається тільки в рамках виділеної підмережі, тобто ваги змінюються тільки у решти нейронів. Кожен нейрон виключається з сукупної мережі з певною ймовірністю, яка називається коефіцієнтом dropout.

Даний шар зменшує час однієї епохи навчання в зв'язку з меншим числом параметрів, що оптимізуються, а також дозволяє краще боротися з перенавчанням мережі щодо стандартних методів регуляризації [10].

3.1.4. Нормалізуючий шар.

На даному шарі відбувається стандартна нормалізація входів (віднімається вибіркоче середнє їх значень, і результат ділиться на корінь вибіркової дисперсії). Вибіркові величини обчислюються з урахуванням значень на входах даного шару на

попередніх ітераціях навчання. Даний підхід дозволяє збільшити швидкість навчання мережі і поліпшити підсумковий результат [4].

3.1.5. Повнозв'язний шар

Даний шар являє собою звичайний багатошаровий персептрон, метою якого є класифікація. Він моделює складну нелінійну функцію, оптимізація якої покращує якість розпізнавання.

Нейрони кожної карти попереднього підвибіркового шару пов'язані з одним нейроном прихованого шару. Таким чином, число нейронів прихованого шару дорівнює числу карт підвибіркового шару. Однак зв'язки можуть бути не обов'язково такими, наприклад, тільки частина нейронів будь-якої з карт підвибіркового шару може бути пов'язана з першим нейроном прихованого шару, а частина, що залишилася з другим, або всі нейрони першої карти пов'язані з нейронами 1 і 2 прихованого шару.

Як і в звичайних нейромережах, в повнозв'язному шарі нейрони мають повні сполуки з усіма активаціями в попередньому шарі. Їх активації можуть бути обчислені за допомогою множення матриць, що супроводжується зміщенням.

Відмінності між повнозв'язними і згортальними шарами полягають у тому, що нейрони шару згортки з'єднані тільки з локальною областю на вході, і що нейрони цього шару можуть спільно використовувати параметри. Однак нейрони в обох шарах, незважаючи на свої особливості, підраховують скалярний добуток, тому їх функціональна форма ідентична. Більш того, виявляється, можна виконати конвертацію між повнозв'язним і згортальним шаром [4]:

- для будь-якого шару згортки існує повнозв'язний шар, який реалізує ту ж функцію проходу. Матриця ваг буде являти собою велику матрицю, яка в основному заповнена нулями за винятком деяких блоків (через локальну зв'язність), ваги більшості з яких будуть рівними (через спільне використання параметрів).

- будь-який повнозв'язний шар може бути перетворений в шар згортки [4].

3.2. Вибір функції активації і функції втрат

3.2.1. Вибір функції активації

Наявність в ЗНМ шарів різного типу призводить до того, що для навчання нейронів цих шарів можуть використовуватися різні функції активації.

При вирішенні задачі класифікації до функції активації ставляться такі вимоги:

1. Значення на виході з шару можуть трактуватися як ймовірність приналежності об'єкта заданому класу. Це означає, що кожне значення має бути в діапазоні від 0 до 1, і сума всіх значень повинна дорівнювати одиниці.

2. Функція повинна бути диференційована, щоб можна було застосовувати градієнтні методи навчання нейронної мережі.

При синтезі ЗНМ можливе застосування всіх функцій активації, розглянутих у другому розділі, так як вони задовольняють перерахованим вимогам. Зазвичай внаслідок істотних позитивних властивостей для прихованих шарів використовується ReLu, а для повнозв'язного шару – функція SoftMax (при вирішенні задач класифікації) або лінійна (для регресійних задач).

Перевага функції SoftMax в тому, що вона використовує експоненту. За рахунок цього вирішується проблема з від'ємними вихідними значеннями, так як експонента завжди додатна. Крім того, експонента значно збільшує великі значення. А також, ця функція диференційована, тобто не виникають проблеми з навчанням.

Таким чином, функцію активації SoftMax зручно застосовувати для задач класифікації, так як вона дозволяє трактувати вихідні значення нейронів як ймовірність належності до даного класу, а також забезпечує, щоб тільки одне вихідне значення було близьке до одиниці за рахунок застосування експоненти.

Слід зазначити, що немає теоретичної необхідності використовувати саме SoftMax для задач класифікації. При наявності достатнього обсягу даних нейронна мережа може навчитися з іншими функціями активації на останньому шарі. Але з функцією м'якого максимуму навчання відбудеться швидше, тому що дана функція

добре підходить саме для класифікації з непересічними класами. Якщо ж класи можуть перетинатися, то функція SoftMax не підходить.

Єдиною складністю як скористатися цією функцією є обчислення її похідних, необхідних для реалізації алгоритму навчання. Однак ці труднощі не є принциповими, оскільки для обчислення градієнтів можна використовувати спеціальні бібліотеки, такі як Theano і TensorFlow.

3.2.2. Вибір функції втрат

При синтезі ЗНМ в якості опції втрат використовують зазвичай квадратичну функцію або перехресну ентропію [1].

Якщо в якості функції активації в вихідному шарі використовується функція активації SoftMax, яка перетворює будь-який вхідний вектор в вектор ймовірностей, то для порівняння двох імовірнісних розподілів необхідно вибрати коректну міру. В якості такого заходу доцільно використовувати перехресну ентропію [3, 7]:

$$C = -\sum_{j=1}^n d_j \log y_j, \quad (3.1)$$

де d_j – необхідний вихід для поточного навчального прикладу, y_j – реальний вихід нейронної мережі.

3.3. Навчання мережі

Навчання ЗНМ аналогічно навчання ШНМ прямого поширення і складається в корекції її вагових параметрів на основі мінімізації деякої обраної функції втрат, наприклад, найпоширенішою в теорії нейронних мереж функції середньоквадратичної помилки. Більш детально питання вибору функції втрат розглядалося вище. Слід зазначити, що для навчання ЗНМ можуть бути використані всі алгоритми навчання персептрона, описані в другому розділі.

Для мінімізації функції помилки досить ефективними є градієнтні методи першого і другого порядків. Зокрема, найбільш ефективним виявився метод

Левенберга-Марквардта (ЛМ) [3]. Хоча алгоритм ЛМ вимагає обробки всієї навчальної вибірки, останнім часом з'явилися його рекурентні аналоги, які, як і алгоритм градієнтного спуску, можуть працювати з кожною окремо взятою навчальною вибіркою [1]. В останньому випадку градієнтний алгоритм називають стохастичним градієнтом.

Для обчислення помилок у всіх шарах мережі і корекції її параметрів використовується широко відомий в теорії ШНМ метод зворотного поширення помилки.

На практиці зазвичай використовується найбільш поширений алгоритм навчання нейронних мереж, заснований на методі градієнтного спуску (метод зворотного поширення помилки) і його модифікації: SGD, Adam, AdaGrad, AdaDelta. [7, 10-13], які будуть розглянуті нижче.

Аналіз способів організації алгоритму навчання (стохастичне і пакетне навчання) показує, що для навчання нейронних мереж переважно використовувати стохастичне навчання, так як:

- стохастичне навчання сходиться зазвичай набагато швидше, ніж пакетний метод навчання;
- стохастичне навчання часто призводить до кращих рішень;
- витрати на одновимірну оптимізацію кроку не дають практичної користі, навчання за сумарним градієнтом (з оптимізацією кроку або без неї) завжди програє стохастичному навчанню.

На підставі аналізу практичних результатів робиться висновок, що в якості активаційної функції необхідно використовувати гіперболічний тангенс. Це обумовлено наступними причинами:

- симетричні активаційні функції, типу гіперболічного тангенса, забезпечують більш швидку збіжність, ніж стандартна логістична функція;
- дана функція має просту і безперервну першу похідну.

У задачах класифікації цільові виходи типово є бінарними (наприклад, ± 1), тобто цільові виходи встановлюються по асимптотам активаційної функції. Такий підхід має кілька недоліків. По-перше, результат навчання може бути нестабільний. По-друге, в разі якщо виходи насичені, мережа не дає впевненості в правильності класифікації.

Вирішення цих проблем полягає в тому, щоб змусити цільові виходи бути в межах діапазону активаційної функції. Встановлення значень на межі максимуму другої похідної активаційної функції - кращий метод використовувати в своїх інтересах нелінійність, не насичуючи активаційну функцію. З цієї причини в якості активаційної функції використовується гіперболічний тангенс $y = 1,7159 \tanh(2x/3)$. Вона має максимальну другу похідну в точках ± 1 , які відповідають використовуваним нами цільовим значенням.

Початкова ініціалізація синапсів нейронної мережі має величезний вплив на кількість ітерацій навчання. Вагові коефіцієнти синапсів повинні бути обрані випадково, але таким способом, щоб активаційна функція перш за все активізувалася в своїй лінійній області. Для досягнення даного ефекту необхідно узгодження між нормалізацією вхідних значень нейронної мережі, вибором активаційної функції і вибором початкових значень вагових коефіцієнтів. Таким чином, для обраної активаційної функції і обраного методу нормалізації вхідних даних в цій роботі вагові коефіцієнти синапсів ініціалізувалися наступним чином: значення вагових коефіцієнтів були обрані випадковим чином з нормального розподілу з нульовим середнім і стандартним відхиленням $\sigma_w = \sqrt{m}$, де m – число зв'язків, що входять в нейрон.

3.3.1. Навчання нейронів вихідного (повнозв'язного) шару

Для навчання нейронів цього шару, як і інших, використовується алгоритм зворотного поширення помилки

$$\Delta \theta(k) = -\eta \nabla_{\theta} J(\theta(k)); \quad (3.2)$$

$$\theta(k+1) = \theta(k) - \eta \nabla_{\theta} J(\theta(k)), \quad (3.3)$$

де θ – параметри мережі (елементи вагових матриць, зсувів, кутів нахилу активаційних функцій і т.д.); $J(\theta(k))$ – цільова функція (функція втрат); η – параметр швидкості навчання.

Якщо в якості функції втрат мережі обрана квадратична функція E , а в якості активаційної функції нейронів сигмоїдальна (2.16), то для налаштування матриці ваг даного шару використовується

$$\begin{aligned} \nabla_{w_{ki}^L} J(w_{ki}^L) &= \frac{\partial E}{\partial w_{ki}^L} = \frac{\partial E}{\partial y_i^L} \frac{\partial y_i^L}{\partial x_i^L} \frac{\partial x_i^L}{\partial w_{ki}^L} = \delta_i^L \cdot \frac{\partial x_i^L}{\partial w_{ki}^L} = \delta_i^L \cdot y_k^{L-1}, \\ \forall i \in (0, \dots, N^L), \forall k \in (0, \dots, N^{L-1}), \end{aligned} \quad (3.4)$$

де

$$\begin{aligned} \delta_i^L &= \frac{\partial E}{\partial y_i^L} \frac{\partial y_i^L}{\partial x_i^L}; \\ \frac{\partial E}{\partial y_i^L} &= y_i^L - d_i; \\ \frac{\partial y_i^L}{\partial x_i^L} &= y_i^L (1 - y_i^L). \end{aligned} \quad (3.5)$$

Якщо ж в якості активаційної функції нейронів обрана SoftMax, то в (3.4) слід взяти $\frac{\partial y_i^L}{\partial x_i^L} = y_i^L (1 - y_i^L)$, і крім того, $\frac{\partial y_i^L}{\partial x_j^L} = -y_i^L y_j^L$, так як $\frac{\partial E}{\partial x_i^L} = \sum_{j=1}^{N^L} \frac{\partial E}{\partial y_j^L} \frac{\partial y_j^L}{\partial x_i^L}, i = (1, \dots, N^L)$.

У матричному вигляді градієнт $\frac{\partial E}{\partial w_{ki}^L}$ може бути представлений таким чином:

$$\frac{\partial E}{\partial w_{ki}^L} = (y^{L-1})^T \delta^L. \quad (3.6)$$

Аналогічно можна отримати процедуру налаштування зсувів для нейронів повнозв'язного шару

$$\frac{\partial E}{\partial b_i^L} = \frac{\partial E}{\partial y_i^L} \frac{\partial y_i^L}{\partial x_i^L} \frac{\partial x_i^L}{\partial b_i^L} = \delta_i^L \cdot \frac{\partial x_i^L}{\partial b_i^L} = \delta_i^L. \quad (3.7)$$

При навчанні нейронів інших шарів знадобиться обчислення градієнтів, наприклад,

$$\frac{\partial E}{\partial y_k^{L-1}} = \sum_{i=1}^{N^L} \delta_i^L \cdot \frac{\partial x_i^L}{\partial y_k^{L-1}} = \sum_{i=1}^{N^L} \delta_i^L \cdot \frac{\partial \left(\sum_{k'=1}^{N^{L-1}} w_{k'i}^L y_{k'}^{L-1} + b_i^L \right)}{\partial y_k^{L-1}} = \sum_{i=1}^{N^L} \delta_i^L \cdot w_{ki}^L, \quad (3.8)$$

$$\forall i \in (1, \dots, N^L) \forall k \in (1, \dots, N^{L-1}).$$

3.3.2. Навчання вихідного (повнозв'язного) шару мережі при виборі функції перехресної ентропії як функції втрат

Для навчання ЗНМ використовується алгоритм стохастичного градієнтного спуску

$$\Delta \theta(k) = -\eta \nabla_{\theta} J(\theta(k)); \quad (3.9)$$

$$\theta(k+1) = \theta(k) - \eta \nabla_{\theta} J(\theta(k)), \quad (3.10)$$

де θ – параметри мережі (елементи вагових матриць, зсувів, кутів нахилу активаційних функцій і т.д.); $J(\theta(k))$ – цільова функція (функція втрат); η – параметр швидкості навчання.

При виборі в якості опції втрат перехресної ентропії процедура навчання вихідного (повнозв'язного) шару ЗНМ набуде вигляду

$$\Delta \theta(k) = -\eta \nabla_{\theta} C(\theta(k)); \quad (3.11)$$

$$\theta(k+1) = \theta(k) - \eta \nabla_{\theta} C(\theta(k)), \quad (3.12)$$

де $\theta(k) = (w, b)$.

Обчислюючи окремі похідні за налаштованими параметрами, маємо

(Тут взято до уваги, що для вихідного шару $y_i^L = f_i^L$)

$$\nabla_w C = \frac{\partial C}{\partial w_{ik}^L} = \frac{\partial C}{\partial y_i^L} \frac{\partial y_i^L}{\partial w_{ik}^L} = \frac{\partial C}{\partial y_i^L} f_k^{L-1} = f_k^{L-1} (f_k^{L-1} - d_i); \quad (3.13)$$

$$\nabla_b C = \frac{\partial C}{\partial b_i^L} = \frac{\partial C}{\partial y_i^L} \frac{\partial y_i^L}{\partial b_i^L} = f_i^L - d_i. \quad (3.14)$$

$$\nabla_f C = \frac{\partial C}{\partial f_j^L} = -\frac{\partial}{\partial f_j^L} \left(\sum_{i=1}^n d_i \ln f_i^L \right) = -\frac{d_j}{f_j^L} = f_i^L - d_i; \quad (3.15)$$

При обчисленні градієнта використовуються наступні похідні:

$$\frac{\partial f_i^L}{\partial x_j^L} = \begin{cases} \frac{e^{x_j^L}}{\sum_{k=1}^n e^{x_k^L}} - \left(\frac{e^{x_j^L}}{\sum_{k=1}^n e^{x_k^L}} \right)^2 & i = j; \\ \frac{e^{x_i^L} e^{x_j^L}}{\left(\sum_{k=1}^n e^{x_k^L} \right)^2} & i \neq j; \end{cases} \quad (3.16)$$

$$= \begin{cases} f_i^L (1 - f_i^L) & , i = j; \\ -f_i^L f_j^L & \end{cases}$$

$$\begin{aligned} \frac{\partial C}{\partial x_i^L} &= -\frac{\partial}{\partial x_i^L} \left(\sum_{j=1}^n d_j \ln f_j^L \right) = \sum_{j=1}^n \frac{\partial C}{\partial f_j^L} \frac{\partial f_j^L}{\partial x_i^L} = \frac{\partial C}{\partial f_i^L} \frac{\partial f_i^L}{\partial x_i^L} - \sum_{j \neq i} \frac{\partial C}{\partial f_j^L} \frac{\partial f_j^L}{\partial x_i^L} = \\ &= -d_i^L (1 - y_i^L) + \sum_{j \neq i} d_j^L y_i^L = -d_i^L + y_i^L \sum_j d_j^L = y_i^L - d_i^L. \end{aligned} \quad (3.17)$$

3.3.3. Навчання нейронів конволюційного шару

Ядро являє собою фільтр, який ковзає по всьому зображенню і знаходить його ознаки в будь-якому місці, тобто забезпечує інваріантність до зсувів.

Формули для поновлення ядра згортки мають вигляд [105–108]

$$\frac{\partial E}{\partial w_{ab}^L} = \sum_i \sum_j \frac{\partial E}{\partial y_{ij}^L} \frac{\partial y_{ij}^L}{\partial x_{ij}^L} \frac{\partial x_{ij}^L}{\partial w_{ab}^L} = \sum_i \sum_j \frac{\partial E}{\partial y_{ij}^L} \frac{\partial y_{ij}^L}{\partial x_{ij}^L} \cdot \frac{\partial \left(\sum_{a'=-\infty}^{+\infty} \sum_{b'=-\infty}^{+\infty} w_{a'b'}^L \cdot y_{(is-a')(js-b')}^{L-1} + b^L \right)}{\partial w_{ab}^L}, \quad (3.18)$$

$\forall a \in (-\infty, \dots, +\infty), \forall b \in (-\infty, \dots, +\infty).$

З іншого боку, розкладаючи суму в чисельнику по a і b , маємо

$$\frac{\partial E}{\partial w_{ab}^L} = \sum_i \sum_j \frac{\partial E}{\partial y_{ij}^L} \frac{\partial y_{ij}^L}{\partial x_{ij}^L} \cdot y_{(is-a)(js-b)}^{L-1}, \forall a \in (-\infty, \dots, +\infty) \forall b \in (-\infty, \dots, +\infty). \quad (3.19)$$

Градiєнт, який використовується в алгоритмі зворотного поширення помилки для крос-кореляції виглядає аналогічно, за винятком зміни знаку при aib :

$$\frac{\partial E}{\partial w_{ab}^L} = \sum_i \sum_j \frac{\partial E}{\partial y_{ij}^L} \frac{\partial y_{ij}^L}{\partial x_{ij}^L} \cdot y_{(is+a)(j+b)}^{L-1}. \quad (3.20)$$

Для однієї карти ознак можна використовувати лише одне зміщення, яке "пов'язано" з усіма елементами цієї карти. Відповідно, при коригуванні значення цього зміщення повинні враховуватися всі значення з карти, отримані при зворотному поширенні помилки. У цьому випадку (при використанні для однієї карти ознак одного зсуву), з урахуванням того, що

$$x_{ij}^L = \sum_{a=-\infty}^{+\infty} \sum_{b=-\infty}^{+\infty} w_{ab}^L \cdot y_{(is-a)(js-b)}^{L-1} + b^L, \quad (3.21)$$

Маємо

$$\frac{\partial E}{\partial b^L} = \sum_i \sum_j \frac{\partial E}{\partial y_{ij}^L} \frac{\partial y_{ij}^L}{\partial x_{ij}^L} \frac{\partial x_{ij}^L}{\partial b^L} = \sum_i \sum_j \frac{\partial E}{\partial y_{ij}^L} \frac{\partial y_{ij}^L}{\partial x_{ij}^L}. \quad (3.22)$$

В якості альтернативного варіанту можна брати стільки зсувів b_{ij}^L для окремої карти ознак, скільки елементів знаходиться в цій карті, але в такому випадку параметрів зсуву буде дуже багато - більше, ніж параметрів самих ядер згортки. Для даного випадку $\frac{\partial E}{\partial b_{ij}^L} = \frac{\partial E}{\partial x_{ij}^L}$.

Для навчання нейронів в шарі конволюції в градієнтній процедурі використовується похідна $\frac{\partial E}{\partial y_{ij}^{L-1}}$, що обчислюється в такий спосіб:

$$\frac{\partial E}{\partial y_{ij}^{L-1}} = \sum_{i'} \sum_{j'} \frac{\partial E}{\partial y_{i'j'}^L} \frac{\partial y_{i'j'}^L}{\partial x_{i'j'}^L} \frac{\partial x_{i'j'}^L}{\partial y_{ij}^{L-1}} = \sum_{i'} \sum_{j'} \frac{\partial E}{\partial y_{i'j'}^L} \frac{\partial y_{i'j'}^L}{\partial x_{i'j'}^L} \cdot w_{(i's-i)(j's-j)}^L, \quad (3.23)$$

де, як і раніше,

$$x_{ij'}^L = \sum_{a=-\infty}^{+\infty} \sum_{b=-\infty}^{+\infty} w_{ab}^L \cdot y_{(i's-a)(j's-b)}^{L-1} + b^L. \quad (3.24)$$

Розкладаючи суму в чисельнику по a і b , отримаємо, що всі приватні похідні дорівнюють нулю, крім того випадку, коли $i's - a = i$ і $j's - b = j$, і, відповідно, $a = i's - i$, $b = j's - j$. Це справедливо тільки для конволюції, для крос-кореляції має бути $i's + a = i$ і $j's + b = j$, відповідно, $a = i - i's$ і $j's - b = j$. І тоді підсумкова формула в разі крос-кореляції буде виглядати так:

$$\frac{\partial E}{\partial y_{ij}^{L-1}} = \sum_{i'} \sum_{j'} \frac{\partial E}{\partial y_{ij'}^L} \frac{\partial y_{ij'}^L}{\partial x_{ij'}^L} \cdot w_{(i-i's)(j-j's)}^L \quad (3.25)$$

3.4. Навчання на основі стохастичного градієнта

Звичайний градієнтний спуск описується співвідношеннями

$$\Delta \theta(k) = -\eta \nabla_{\theta} J(\theta(k)); \quad (3.26)$$

$$\theta(k+1) = \theta(k) - \eta \nabla_{\theta} J(\theta(k)), \quad (3.27)$$

де θ – параметри мережі, $J(\theta(k))$ – цільова функція (функція втрат); η – параметр швидкості навчання.

Йому притаманні такі недоліки:

- застрягання в локальних мінімумах або сідлових точках мінімізуючого функціоналу;
- повільна збіжність внаслідок складного ландшафту цільової функції, коли плато чергуються з регіонами сильної нелінійності (похідна на плато практично дорівнює нулю, а раптовий обрив, навпаки, може занадто змінити оцінку параметра);
- деякі параметри оновлюються значно рідше інших, особливо коли в даних зустрічаються інформативні, але рідкісні ознаки, що погано позначається на нюансах узагальнюючого правила мережі. З іншого боку, надання занадто великої значимості взагалі всім ознакам, що рідко зустрічаються, і може привести до перенавчання.

– занадто малі значення параметра призводять до повільної збіжності і застрягання в локальних мінімумах, занадто великі – до «проскакування» вузьких глобальних мінімумів або зовсім розбіжності.

Використання методів другого порядку, розглянутих вище, вимагає обчислення гессіану – матриці похідних по кожній парі параметрів, а для методу Ньютона – ще й зворотний до неї, тобто реалізація даних методів пов'язана зі значними обчислювальними витратами.

У зв'язку з цим на практиці широкого поширення набули методи, засновані на методі стохастичного градієнта і володіють в порівнянні з ним цілою низкою переваг.

Розглянемо ці методи більш докладно.

NAG (Nesterov Accelerated Gradient) [108]. Даний алгоритм реалізує ідею накопичення імпульсу, використовуючи інформацію про зміну кожного параметра у вигляді експоненційної середньої змінної

$$v(k+1) = \gamma v(k) + (1-\gamma)x. \quad (3.28)$$

В алгоритмі накопичується градієнт цільової функції мережі

$$v(k+1) = \gamma v(k) + \eta \nabla_{\theta} J(\theta(k)), \quad (3.29)$$

який і використовується при корекції параметрів

$$\theta(k+1) = \theta(k) - v(k). \quad (3.30)$$

Більш точні співвідношення для алгоритму NAG мають вигляд

$$v(k+1) = \gamma v(k) + \eta \nabla_{\theta} J(\theta(k) - \gamma v(k)); \quad (3.31)$$

$$\theta(k+1) = \theta(k) - v(k+1). \quad (3.32)$$

Adagrad [101]. Adagrad (adaptive gradient) враховує частоту активації нейронів шляхом зберігання для кожного параметра мережі суми квадратів його оновлень. При цьому використовується модифікована формула оновлення

$$M\{g^2(k+1)\} = \gamma M\{g^2(k)\} + (1-\gamma)g^2(k), \quad (3.33)$$

а корекція параметрів здійснюється за правилом

$$\theta(k+1) = \theta(k) - \frac{\eta}{\sqrt{G(k) + \varepsilon}} g(k), \quad (3.34)$$

де $G(k)$ – сума квадратів оновлень, а ε – параметр, що згладжує, необхідний для того, щоб уникнути поділу на 0. У часто оновлюваного в минулому параметра $G(k)$ буде великий (великий знаменник в (3.35)), тобто параметр зміниться небагато. Рідко змінюваний параметр зміниться суттєво. Параметр ε зазвичай обирають близько 10^{-6} - 10^{-8} .

RMSProp. Недолік Adagrad в тому, що $G(k)$ в (3.35) може збільшуватися скільки завгодно, в результаті чого через деякий час оновлення стають занадто маленькими, що призводить до паралічу алгоритму. RMSProp і Adadelta спрямовані виправити цей недолік. RMSprop – це неопублікований метод адаптивної швидкості навчання, запропонований Джеффом Хінтоном в лекції 6-го його курсу Coursera.

З цією метою Adagrad модифікується таким чином, щоб часто оновлювані ваги корегувалися рідше. Для цього замість повної суми оновлень використовується усереднений по історії квадрат градієнта, тобто середня змінна виду

$$M\{g^2(k+1)\} = \gamma M\{g^2(k)\} + (1-\gamma)g^2(k), \quad (3.35)$$

тоді замість (3.35) отримаємо

$$\theta(k+1) = \theta(k) - \frac{\eta}{\sqrt{M\{g^2(k)\} + \varepsilon}} g(k). \quad (3.36)$$

Так як знаменник являє собою корінь із середнього квадратів градієнтів

$$RMS\{g(k)\} = \sqrt{M\{g^2(k)\} + \varepsilon}, \quad (3.37)$$

Алгоритм отримав назву RMSProp — root mean square propagation.

Adadelta [13] є продовженням Adagrad, метою якого є недопущення монотонного зменшення швидкості навчання. Замість того, щоб накопичувати всі градієнти минулого квадрата, Adadelta обмежує вікно накопичених градієнтів минулого фіксованим розміром.

Adadelta відрізняється від RMSProp тим, що ми додаємо в чисельник (3.36) стабілізуючий член пропорційний RMS від $\Delta\theta(k)$. На кроці $k+1$ значення $RMS\{\Delta\theta(k)\}$ ще не відомо, тому оновлення параметрів відбувається в три етапи, а не в два: спочатку накопичується квадрат градієнта, потім оновлюється θ , після чого оновлюється $RMS\{\Delta\theta(k)\}$, тобто

$$\theta(k+1) = \theta(k) - \frac{RMS\{\Delta\theta(k)\}}{RMS\{g(k)\}} g(k); \quad (3.38)$$

$$M\{\|\Delta\theta(k+1)\|^2\} = \gamma M\{\|\Delta\theta(k)\|^2\} + (1-\gamma)\|\Delta\theta(k)\|^2; \quad (3.39)$$

$$RMS\{\Delta\theta(k)\} = \sqrt{M\{\|\Delta\theta(k)\|^2\} + \varepsilon}. \quad (3.40)$$

Для RMSProp, Adadelta і для Adagrad не потрібно дуже точно підбирати швидкість навчання, досить його наближеного значення. Зазвичай радять почати підгін η з $0,1-1$, а γ так і залишити $0,9$. Чим ближче γ до 1 , тим довше RMSProp і Adadelta з більшим $RMS\{\Delta\theta(k)\}$ будуть сильно оновлювати мало використовувані ваги. Якщо ж $\gamma \approx 1$ і $RMS\{\Delta\theta(k)\} = 0$, то Adadelta буде довго «с» відноситись до рідко використовуваних ваг, що може привести як до паралічу алгоритму, так і викликати навмисно «жадібну» поведінку, коли алгоритм спочатку оновлює нейрони, які кодують найкращі ознаки.

Adam (adaptive moment estimation) - ще один оптимізаційний алгоритм. Він поєднує в собі і ідею накопичення руху і ідею слабшого поновлення ваг для типових ознак. За аналогією з (3.29) маємо:

$$m(k+1) = \beta_1 m(k) + (1-\beta_1)g(k). \quad (3.41)$$

Від Нестерова Adam відрізняється тим, що ми накопичуємо не $\Delta\theta$, а значення градієнта. Для отримання інформації про зміну градієнта в [99] запропоновано оцінювати ще й середню нецентрованну дисперсію:

$$v(k+1) = \beta_2 v(k) + (1 - \beta_2) g^2(k). \quad (3.42)$$

Так як це, по суті, являє собою вираз (3.35), то даний алгоритм не відрізняється від RMSProp.

Автори Adam пропонують в якості значень за замовчуванням $\beta_1 = 0,9, \beta_2 = 0,999, \varepsilon = 10^{-8}$ і стверджують, що алгоритм виступає краще або приблизно так само, як і всі попередні алгоритми на широкому наборі датасета за рахунок початкового калібрування.

Adamax[99]. Алгоритм Adamax є модифікацією алгоритма Adam, в якому замість дисперсії (3.42) використовується інерційний момент розподілу градієнтів довільного ступеня p . Хоча це може привести до нестабільності обчислень, на практиці випадок $p \rightarrow \infty$ працює дуже добре.

$$v(k+1) = \beta_2^p v(k) + (1 - \beta_2^p) |g(k)|^p. \quad (3.43)$$

3.5. Робастне навчання

Як вже зазначалося вище, в якості повнозв'язного шару ЗНМ вибирається зазвичай багат шаровий перцептрон. Тому для навчання цього шару можуть бути використані всі алгоритми навчання перцептрона, розглянуті в другому розділі.

Описані в третьому розділі процедури навчання повнозв'язного і конволюційного шарів використовують градієнтні алгоритми мінімізації функціоналів якості, що представляють собою або квадратичну функцію, або функцію перехресної ентропії, що є оптимальними для гауссівського розподілу, будуть нестійкими при наявності перешкод вимірів, розподіли яких відмінні від нормального, так як $\rho'(e(i, \theta)) = e(i, \theta)$ зростає лінійно зі збільшенням $e(i, \theta)$.

Тому в якості функціоналів, що мінімізуються, доцільно вибирати комбіновані функціонали Хьюбера (2.45) і Хемпела (2.46) і які складаються з квадратичного, що забезпечує оптимальність оцінок для гауссівського розподілу, і модульного, що

дозволяє отримати більш робастну до розподілів з важкими «хвостами» (викидами) оцінку.

При виборі даних функціоналів в якості функцій втрат для градієнтного (узагальненого) дельта правила навчання використовується похідна

$$\frac{\partial E}{\partial w_{ij}^l} = \frac{\partial \rho}{\partial y_i^{l-1}} \frac{\partial y_i^{l-1}}{\partial x_i^l} \frac{\partial x_i^l}{\partial w_{ij}^l}, \quad (3.44)$$

де для функціоналу Хьюбера (2.45)

$$\frac{\partial \rho}{\partial y_i^{l-1}} = \begin{cases} d_i - y_i, & |d_i - y_i| \leq c; \\ c \operatorname{sign}(d_i - y_i), & c < |d_i - y_i| \end{cases} \quad (3.45)$$

а для функціоналу Хемпела (2.46)

$$\frac{\partial \rho}{\partial y_i^{l-1}} = \begin{cases} d_i - y_i, & |d_i - y_i| \leq b; \\ b \operatorname{sign}(d_i - y_i), & b < |d_i - y_i| < c; \\ \frac{b(|d_i - y_i| - d)}{c - d} \operatorname{sign}(d_i - y_i), & c \leq |d_i - y_i| < d; \\ 0, & d \leq |d_i - y_i|. \end{cases} \quad (3.46)$$

Тут a, b, c і d – константи, що визначають ступінь завадостійкості.

Враховуючи, що для p -го образу

$$\begin{aligned} x_i^l &= \sum_{k=1}^{N^{l-1}} w_{ki}^l y_k^{l-1} + b_i^l; \\ y_i^l &= f(x_i^l), \end{aligned} \quad (3.47)$$

Співмножники, що входять до співвідношення (3.45) обчислюються наступним чином:

$$\frac{\partial y_i^{l-1}}{\partial x_i^l} = f'(x_i^l) = (1 - f(x_i^l)); \quad (3.48)$$

$$\frac{\partial x_i^l}{\partial w_{ij}^l} = y_i^{l-1}. \quad (3.49)$$

Таким чином, узагальнене дельта - правило навчання ваг і зсуву набуде вигляду

$$\begin{aligned}\Delta w_{ij}^l &= \gamma \delta_i^l y_i^{l-1}; \\ \Delta b_i^l &= \gamma \delta_i^l,\end{aligned}\tag{3.50}$$

$$\text{де } \delta_i^l = -\frac{\partial \rho}{\partial x_i^l} = -f'(x_i^l) \frac{\partial \rho}{\partial y_i^l}.$$

Як і в разі навчання традиційного БП, забезпечення робастних властивостей одержуваних оцінок параметрів мереж можливо шляхом використання комбінованого функціоналу навчання (2.47)

$$E = \lambda \sum_{p=1}^P \sum_{j=1}^{N^L} (d_{p,j} - y_{p,j})^2 + (1-\lambda) \sum_{p=1}^P \sum_{j=1}^{N^L} |d_{p,j} - y_{p,j}|,\tag{3.51}$$

де $\lambda \in [0,1]$, що приводить до градієнтного алгоритму виду (2.48)

З урахуванням того, що для p -го образу виконується (3.47), за аналогією з (3.45) можна записати

$$\frac{\partial E}{\partial w_{ij}^l} = \frac{\partial E}{\partial y_i^{l-1}} \frac{\partial y_i^{l-1}}{\partial x_i^l} \frac{\partial x_i^l}{\partial w_{ij}^l}.\tag{3.52}$$

Співмножники, що входять в дане співвідношення, обчислюються аналогічно викладеному вище. При цьому

$$\frac{\partial E}{\partial y_i^{l-1}} = 2\lambda(d_i^{l-1} - y_i^{l-1}) + (1-\lambda)\text{sign}(d_i^{l-1} - y_i^{l-1}),\tag{3.53}$$

А $\frac{\partial y_i^{l-1}}{\partial x_i^l}$ і $\frac{\partial x_i^l}{\partial w_{ij}^l}$ обчислюються відповідно до (3.48), (3.51).

Остаточно узагальнене дельта - правило навчання ваг і зсуву для даного функціоналу (3.50) набуде вигляду

$$\begin{aligned}\Delta w_{ij}^l &= \gamma \delta_i^l y_i^{l-1}; \\ \Delta b_i^l &= \gamma \delta_i^l,\end{aligned}\tag{3.54}$$

$$\text{де } \delta_i^l = -\frac{\partial E}{\partial x_i^l} = -f'(x_i^l) \frac{\partial E}{\partial y_i^l}.$$

Для стандартного алгоритму ЗП, коли помилка поширюється від виходу до входу мережі і ваги попереднього шару обчислюються через уже відомі ваги наступного шару, величина δ_i^l наступним чином:

$$\delta_i^l = f'(x_i^l) \sum_{k=1}^{N^{l+1}} \delta_k^{l+1} w_{ki}^{l+1}.\tag{3.55}$$

Як вже зазначалося вище, даний алгоритм поєднує властивості МНК з властивостями методу найменших модулів. Варіюючи параметр λ , можна змінювати властивості алгоритму.

Слід також зазначити, що розглянуті робастні процедури навчання можуть бути модифіковані шляхом використання в них ідеї накопичення імпульсу, як це зроблено в розглянутих вище модифікаціях алгоритму стохастичного градієнта.

Згорткові нейронні мережі ідеально підходять для реалізації на GPU завдяки природному паралелізму. Однак навіть при використанні графічних прискорювачів сучасні ЗНМ, що включають десятки і сотні мільйонів параметрів, не в змозі обігнати класичні каскадні бустінг-детектори. Але недавно було показано, що глибокі ЗНМ містять надмірну кількість параметрів, яка може бути в рази скорочена без істотної втрати точності класифікації [26, 29].

Графічні процесори дозволяють в повній мірі розкрити всі переваги природного паралелізму нейронних мереж, а також мають апаратну підтримку вибірки з двовимірних масивів.

Недоліком ЗНМ є необхідність зберігання великого обсягу проміжних обчислень (карт ознак), що збільшує обсяг необхідної пам'яті, а також знижує

ефективність багатопоточного виконання при недостатньому обсязі кеш-пам'яті процесора.

3.6. Висновки по розділу 3

Розглянуто особливості обробки зображень на основі згортальних нейронних мереж, що дозволяють отримати на сьогоднішній день кращі результати в розпізнаванні зображень і представляють собою ключову технологію Deep Learning. ЗНМ забезпечують часткову стійкість до змін масштабу, зсувів, поворотів, зміні ракурсу і іншим спотворенням. У даних мережах ввідні дані складаються з зображень, і вони обмежують побудову мережі найбільш розумним шляхом. На відміну від звичайної нейронної мережі, шари ЗНМ складаються з нейронів, розташованих в трьох вимірах: ширині, висоті і глибині, тобто вимірах, які формують обсяг. Для організації ЗНМ застосовується три основні шари: згортки, пулінгу (підвибірки або субдискретизації), повнозв'язного шару. Визначення топології мережі орієнтується на завдання, яке вирішується, дані з наукових статей і власний експериментальний досвід. Описана загальна архітектура ЗНМ і особливості побудови її шарів.

Наявність в ЗНМ шарів різного типу призводить до того, що для навчання нейронів цих шарів можуть використовуватися різні функції активації. Розглянуто вимоги, які пред'являються до функції активації при вирішенні задачі класифікації. При синтезі ЗНМ можливе застосування всіх функцій активації, розглянутих у другому розділі, однак внаслідок істотних позитивних властивостей для прихованих шарів використовується ReLu, а для повнозв'язного шару - функція SoftMax (при вирішенні задач класифікації, так як вона дозволяє трактувати вихідні значення нейронів як ймовірність належності до даного класу) або лінійна (для регресійних задач). Єдиною складністю як скористатися цією функцією є обчислення її похідних, необхідних для реалізації алгоритму навчання. Однак ці труднощі не є

принциповими, оскільки для обчислення градієнтів можна використовувати спеціальні бібліотеки, такі як Theano і TensorFlow.

При синтезі ЗНМ в якості опції втрат використовують зазвичай квадратичну функцію або перехресну ентропію. Перша використовується в шарах згортки і пулінгу (підвибірки або субдискретизації), друга – в вихідному шарі, якщо в якості функції активації в цьому шарі використовується функція SoftMax, яка перетворює будь-який вхідний вектор в вектор ймовірностей.

Розглянуто питання навчання ЗНМ, яке аналогічно навчанню ШНМ прямого поширення і складається в корекції її вагових параметрів на основі мінімізації деякої з розглянутих вище функції втрат. Описана градієнтна процедура навчання вихідного (повнозв'язного) шару мережі при виборі в якості опції втрат перехресної ентропії.

Розглянуто особливості навчання ЗНМ на основі стохастичного градієнта. Недоліки цього методу (застрявання в локальних мінімумах, повільна збіжність внаслідок складного ландшафту цільової функції, можливість перенавчання і т.д.), а також обчислювальна складність методів другого порядку привели до розробки методів, заснованих на методі стохастичного градієнта, які володіють в порівнянні з ним цілим рядом переваг і використовують ідею накопичення руху (NAG, Adagrad, RMSProp, Adadelata, Adam, Adamax, Nadam). Відмічені особливості реалізації даних методів.

Описані в даному розділі процедури навчання повнозв'язного і конволюційного шарів використовують градієнтні алгоритми мінімізації функціоналів якості, що представляють собою або квадратичну функцію, або функцію перехресної ентропії, що є оптимальними для гауссівського розподілу, будуть нестійкими при наявності перешкод вимірів, розподіли яких відмінні від нормального. У зв'язку з цим пропонується використовувати для навчання шарів ЗНМ робастний підхід і отримані робастні процедури навчання на основі узагальненого дельта-правила.

4 ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ І РІШЕННЯ ПРАКТИЧНИХ ЗАВДАНЬ

Імітаційне моделювання є необхідним етапом дослідження ефективності як обраної архітектури ШНМ, так і вживаного алгоритму навчання мережі.

В даний час найбільш масовим напрямком нейрокомп'ютингів є моделювання нейронних мереж на персональних комп'ютерах. Успіхи в моделюванні дозволяють створювати навчальні програми, що дають користувачеві повне уявлення про ШНМ і їх можливості.

Однак незалежно від використовуваного пакету результатом експериментального дослідження є визначення областей найбільш ефективного застосування того чи іншого типу ШНМ і розробка рекомендацій щодо їх практичного використання.

В даний час популярні і широко використовуються наступні програмні нейросимулятори: MATLAB NeuralNetworkToolbox, GNU Octave, TensorFlow, PyTorch, Scikit-learn, Scilab5.5.2, NeurophStudio 2.92, Theano.

У даній роботі штучні нейронні мережі прямого поширення були реалізовані з використанням TensorFlow – відкритої програмної бібліотеки для машинного навчання, а також GNU Octave – безкоштовної альтернативи MATLAB

4.1. Моделювання процесів розпізнавання і класифікації ШНМ прямого поширення

Завдання визначення неоднорідності забарвлення можна уявити, як задачу класифікації: клас А – окрас неоднорідний, клас В – окрас однорідний. Для такого завдання, як уже говорилося раніше, PNN мережа являє собою ідеальний класифікатор. Для навчання був обраний датасет, що складається з двох класів шкірних новоутворень: злоякісне шкірне новоутворення (меланома) і доброякісне (невус). За своєю природою, однією з основних відмінних рис цих класів є наявність неоднорідності забарвлення у одного типу – меланоми, і відсутність у іншого – невуся. Навчальна вибірка складається з 10000 зображень, тестова – 3000 [109-112].

У таблиці 4.1 представлені результати моделювання. Можна простежити, як збільшується розмір мережі і підвищується точність класифікації залежно від розмірності вхідного вектору. Також можна побачити, як змінюється точність класифікації залежно від параметра згладжування σ (1.7).

Таблиця 4.1 – Моделювання PNN мережі для вирішення задачі визначення неоднорідності забарвлення

Розмірність вхідного вектора	Параметр згладжування sigma (σ)	Розмір мережі (mb)	Час навчання (sec)	Час класифікації (sec)	Точність класифікації
8*8*3	0.1	2.25	0.003	0.08	0.64
8*8*3	0.5	2.25	0.005	0.075	0.593
8*8*3	1	2.25	0.003	0.027	0.515
16*16*3	0.1	8.85	0.007	0.058	0.666
16*16*3	0.5	8.85	0.013	0.053	0.678
16*16*3	1	8.85	0.006	0.053	0.631
32*32*3	0.1	35.22	0.03	0.338	0
32*32*3	0.5	35.22	0.067	0.435	0.674
32*32*3	1	35.22	0.033	0.318	0.674
64*64*3	0.1	140.67	0.161	1.456	0
64*64*3	0.5	140.67	0.158	1.26	0.735
64*64*3	1	140.67	0.269	1.631	0.647
100*100*3	0.1	343.38	0.39	2.98	0
100*100*3	0.5	343.38	0.6	3.204	0.807
100*100*3	1	343.38	0.367	2.926	0.678
128*128*3	0.1	562.56	0.606	4.754	0
128*128*3	0.5	562.56	0.471	3.637	0.767
128*128*3	1	562.56	0.433	3.466	0.748
156*156*3	0.1	835.56	0.619	5.701	0
156*156*3	0.5	835.56	0.656	5.603	0
156*156*3	1	835.56	0.885	6.522	0.791
184*184*3	0.1	1162.41	1.393	17.392	0
184*184*3	0.5	1162.41	1.314	9.402	0
184*184*3	1	1162.41	1.773	15.764	0.836

Як можна помітити, зі збільшенням розмірності вхідного вектору розмір мережі збільшується (рисунок 4.1). Це відбувається через те, що прихований шар

шаблонів PNN мережі містить по одному нейрону для кожного вхідного вектора з навчальної вибірки, в нашому випадку 10000, а зі збільшенням розмірності вхідного вектора збільшується місце, необхідне для мережі.

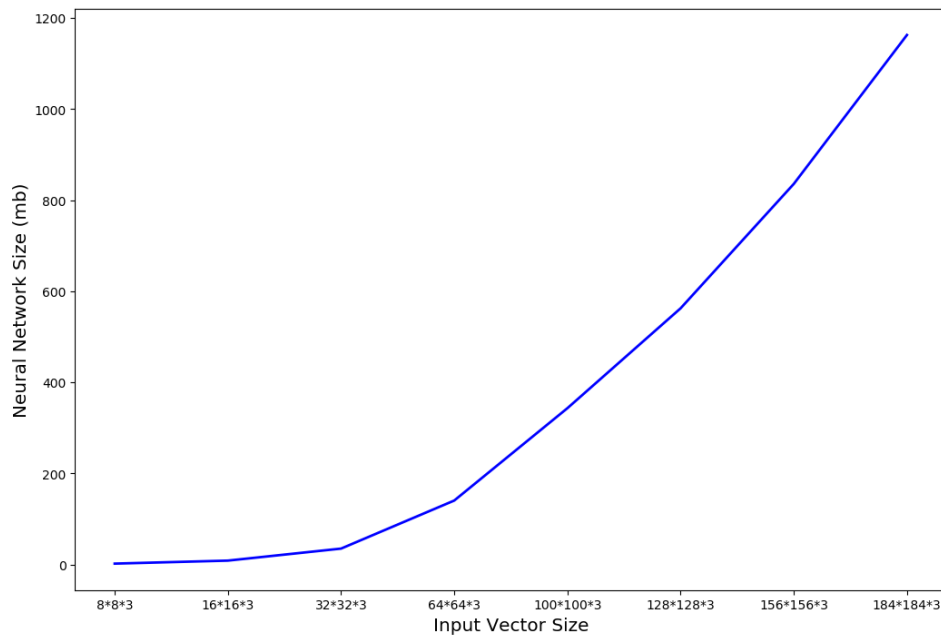


Рисунок 4.1—Залежність розміру PNN мережі від розмірності вхідного вектору

Збільшення розміру мережі, в свою чергу впливає на час, необхідний для класифікації – рисунок 4.2.

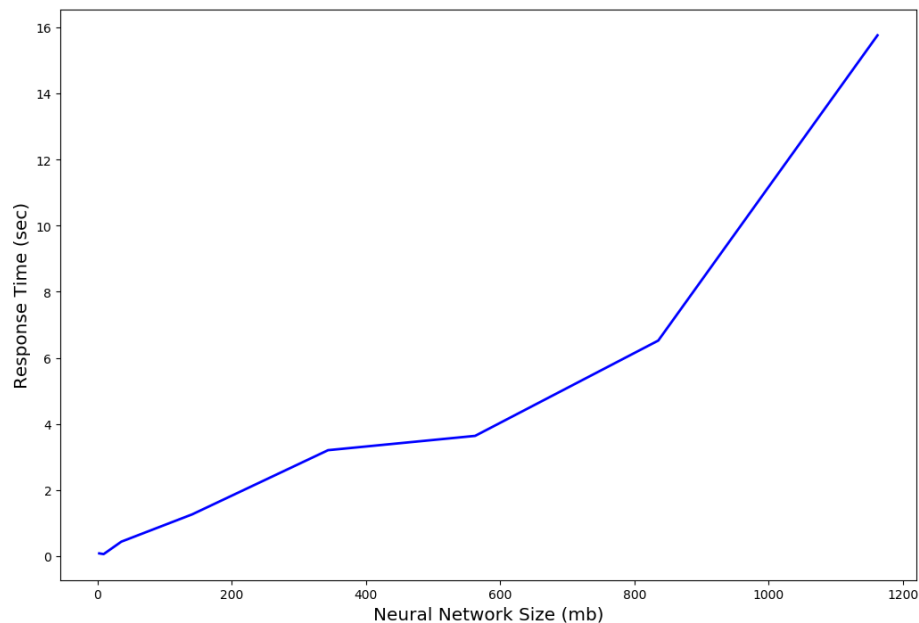


Рисунок 4.2 – Залежність часу, необхідного для класифікації, від розміру PNN мережі

Як можна бачити, час класифікації становить близько 16 секунд в разі, коли розмірність вхідного вектору $184 \times 184 \times 3$, а кількість навчальних прикладів 10000. Підвищити швидкодію можна за рахунок зменшення розмірності вхідного вектору, що не завжди можливо через специфіку предметної області, або за рахунок зменшення розміру навчальної вибірки. В останньому випадку необхідний ретельний аналіз вхідних даних з метою вибору мінімально необхідного навчального набору без втрати репрезентативності.

Великі витрати на місце, що виділяється для функціонування мережею, компенсується підвищенням точності класифікації - зі збільшенням розміру, займаного PNN мережею, точність класифікації зростає (рисунок 4.3).

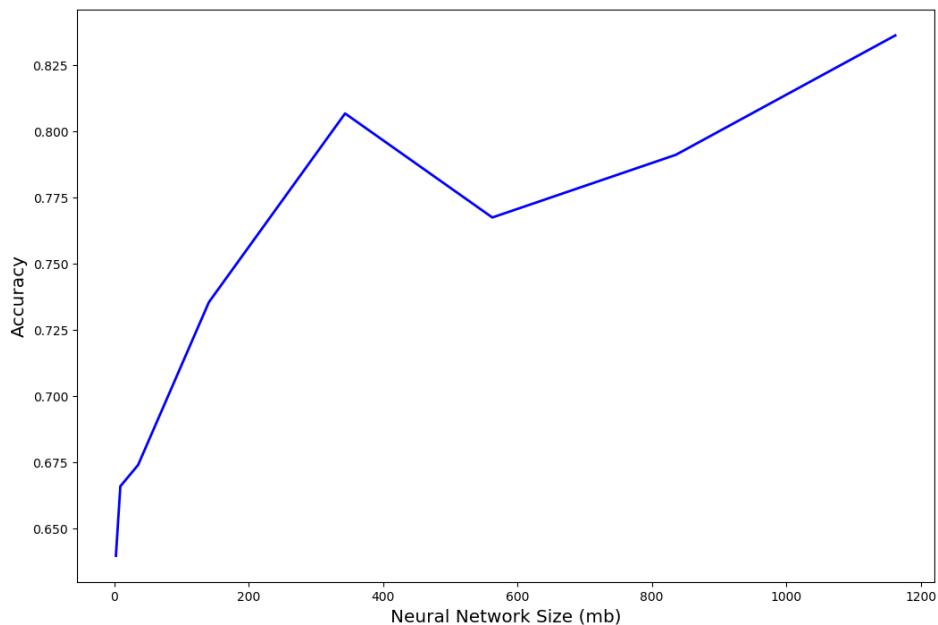


Рисунок 4.3 – Залежність точності класифікації від розміру PNN мережі.

Одним з основних переваг PNN мереж є той факт, що мережа навчається дуже швидко – рисунок 4.4. Для навчання мережі, яка займає 1gb пам'яті, потрібно менше 1.5 секунди. Це пояснюється кардинальною відмінністю процесу навчання мереж даного типу від навчання в мережах прямого поширення з використанням підстроювання ваг на основі методу ЗП. Ітеративний підрядок ваг тут відсутній, а самонавчання зводиться до того, що в шар шаблонів просто додаються додаткові нейрони, що відповідають за нові дані для класифікації.

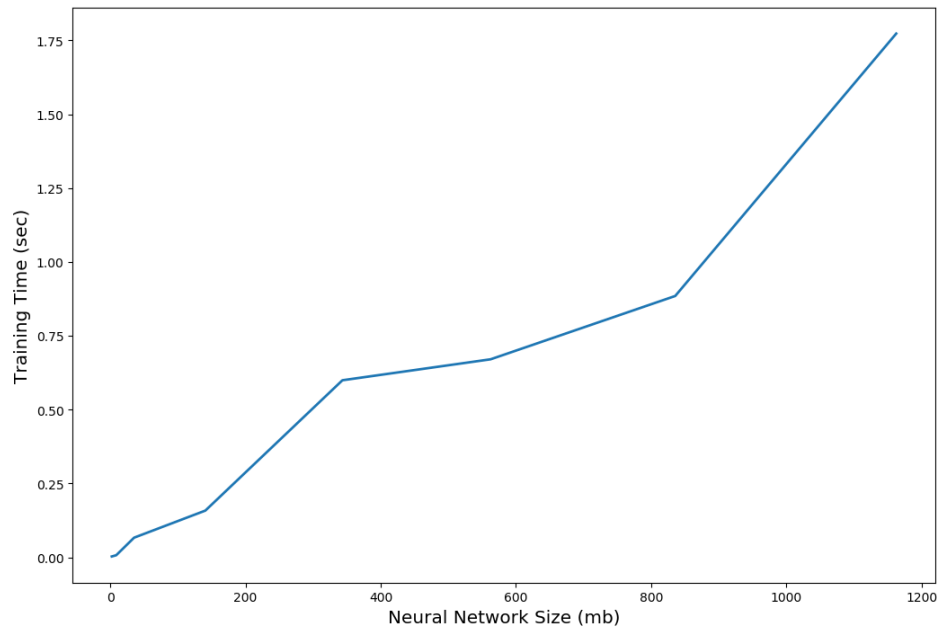


Рисунок 4.4 – Залежність часу, необхідного для навчання PNN мережі, від її розміру.

Незважаючи на простоту процесу навчання і відсутність великої кількості гіперпараметрів, необхідних для підстроювання, в мережах PNN присутній параметр згладжування σ , який виконує велику роль на етапі класифікації, а оптимізація точності відповіді мережі безпосередньо пов'язана з місцем знаходження оптимального значення цього параметра (рисунок 4.5). З рисунку видно, що зі збільшенням розмірності вхідного вектору, σ зміщується до 1.

Оптимальне значення параметра згладжування може бути знайдено евристичним шляхом і залежить від розподілу вхідних даних, тому важливо помітити, що вхідний вектор повинен бути заздалегідь оброблений, щоб не було великих розбігів від середнього значення. Так як в даному експерименті оброблялися кольорові зображення, а діапазон значень, які може приймати кожен піксель становить $[0, \dots, 255]$, перед вирівнюванням в вектор і подачею на вхід нейронної мережі цей діапазон перетворювався в діапазон $[0;1]$ шляхом ділення на 255.

У таблиці 4.2 представлені результати ітеративного пошуку параметра σ в діапазоні $[0;2]$ з кроком 0.1 .

Таблиця 4.2 – Залежність точності класифікації від параметра згладжування

	Точність класифікації для вхідного образу із заданою розмірністю			
Параметр згладжування	100*100*3	128*128*3	156*156*3	184*184*3
0.1	0	0	0	0
0.2	0	0	0	0
0.3	0	0	0	0
0.4	0.772	0	0	0
0.5	0.825	0.767	0	0
0.6	0.809	0.816	0.766	0
0.7	0.772	0.839	0.805	0.759
0.8	0.748	0.796	0.839	0.798
0.9	0.711	0.777	0.829	0.814
1	0.678	0.748	0.791	0.846
1.1	0.66	0.727	0.772	0.809
1.2	0.66	0.706	0.742	0.775
1.3	0.655	0.686	0.732	0.772
1.4	0.664	0.669	0.711	0.746
1.5	0.664	0.669	0.686	0.736
1.6	0.664	0.669	0.665	0.72
1.7	0.664	0.664	0.66	0.703
1.8	0.664	0.664	0.651	0.682
1.9	0.664	0.664	0.66	0.665
2.0	0.664	0.664	0.655	0.66

На рисунку 4.6 наведені результати аналізу точності класифікації від вибору параметра згладжування σ . Як можна бачити з графіків, оптимальне значення параметра σ залежить від розмірності вхідного вектора, а максимальна точність досягається в діапазоні $[0.3 \dots 1.1]$

Наступним етапом був перебір значень в діапазоні $[0.3 \dots 1.1]$ з кроком 0.01, що дозволило уточнити параметр σ і ще більше підвищити точність класифікації.

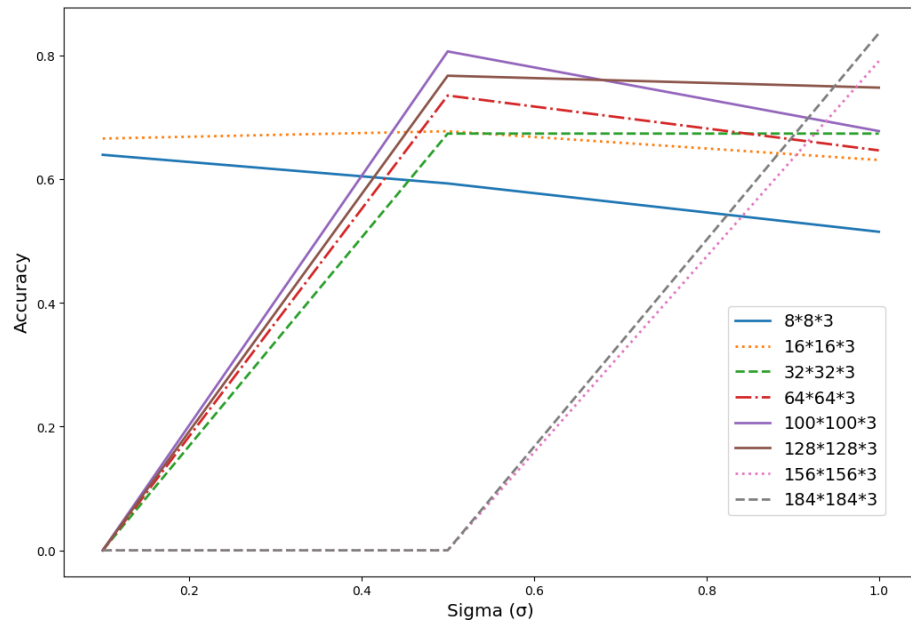


Рисунок 4.5 – Вплив величини параметра згладжування σ на точність класифікації.

Таким чином, PNN мережі відмінно підходять для завдань, в яких потрібно високоточний класифікатор з дуже швидким часом навчання і де не настільки принципова повільна швидкість самої класифікації.

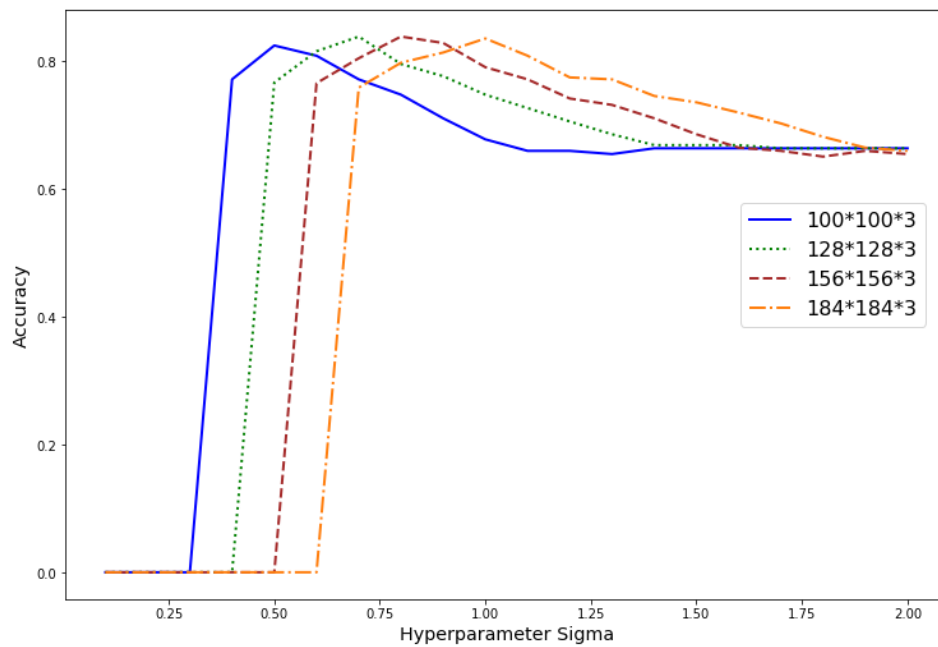


Рисунок 4.6 – Вплив вибору величини параметра згладжування σ на точність класифікації

При цьому потрібно враховувати розмірність вектору вхідних параметрів і кількість прикладів, що навчаються, від яких безпосередньо залежить час відгуку на етапі класифікації, а також місце, що виділяється для ШНМ.

Основні відмінні риси PNN мережі і БП, які дозволяють PNN мережі більш оптимально справлятися з поставленим завданням наступні:

- прихований шар в мережах ймовірнісної нейронної мережі є нелінійним, в той час як вихідний - лінійним. У той же час приховані і вихідний шари багат шарового персептрона, використовуваного в якості класифікатора, є нелінійними;

- аргумент функції активації кожного прихованого вузла мережі PNN є евклидовою нормою (відстанню) між вхідним вектором і центром радіальної функції. У той же час аргумент функції кожного прихованого вузла БП – це скалярний добуток вхідного вектора і синаптичних ваг даного нейрона;

- БП забезпечує глобальну апроксимацію нелінійного відображення. З іншого боку, мережу PNN за допомогою експоненційного зменшення локалізованих нелінійностей (функцій Гаусса) створює локальну апроксимацію нелінійного відображення.

4.2. Моделювання процесу розпізнавання і класифікації загортальними ШНМ

Експеримент1 [113]. Вирішувалося завдання розпізнавання захворювання шкіри – меланоми, за допомогою згортальної нейронної мережі MRCNN [6]. Дана мережа є розширеною версією мережі Faster RCNN, яка в свою чергу є вдосконаленою версією мережі Fast RCNN.

У якості навчальної вибірки використовувався набір фотографій з конкурсу «ISIC 2016: Skin Lesion Analysis Towards Melanoma Detection». В даному наборі міститься 900 фотографій різних пошкоджень шкіри і відповідні їм бітові маски, що дозволяють визначити точний контур пошкодження. Даний набір був розбитий на

600 навчальних фотографій і 300 перевірочних. Приклади зображень з навчальної вибірки наведені на рис.4.7.

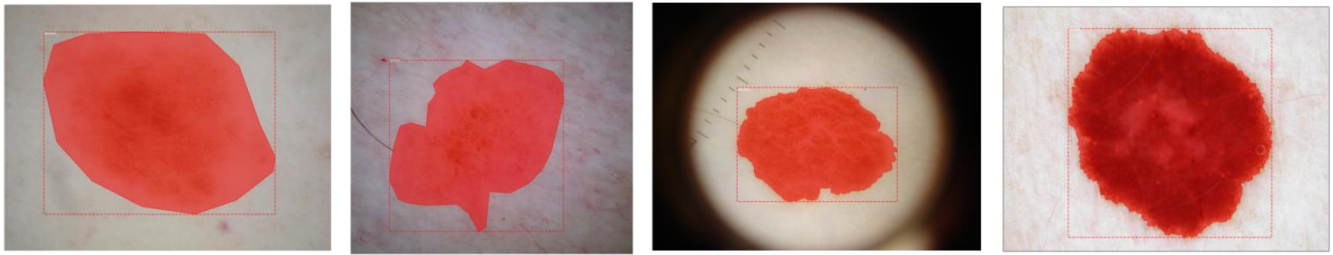
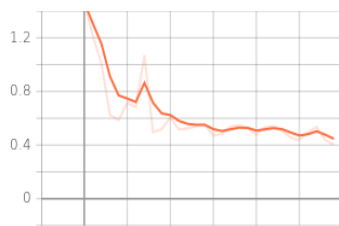
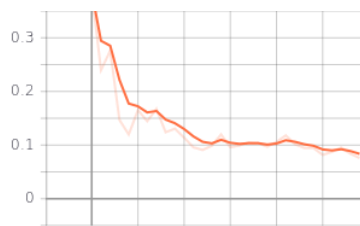


Рисунок 4.7 – Приклади зображень з накладеними масками з навчальної вибірки

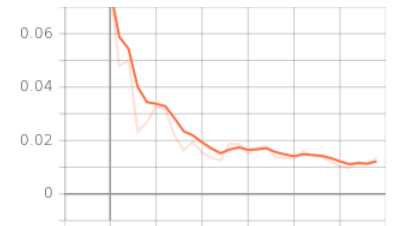
Навчання мережі проводилося з використанням ваг з наперед навченої на вибірці “Coco” (<http://cocodataset.org>) мережі resnet-101. Навчання проводилося протягом 30 епох, і його результати наведені на рис. 4.8.



а) загальна помилка



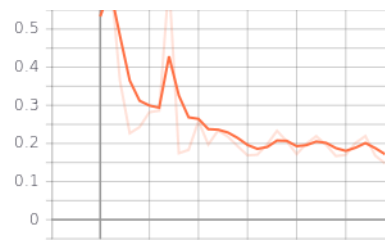
б) помилка модуля
cnn_bbox_loss



в) помилка модуля
cnn_class_loss



г) помилка модуля cnn_mask_loss



д) помилка модуля rpn_bbox_loss

Рисунок 4.8 – Результати навчання мережі.

На рис. 4.8 – а) показаний графік зміни загальної помилки навчання, що складається з помилки детектора «BoundingBox» – рис.4.8 – б), помилки класифікатора 4.8 – с), помилки провісника маски – 4.8 – д) і помилки модуля, що пропонує регіони, що містять об'єкти для розпізнавання 4.8–д).

Результати розпізнавання показані на рис. 4.9. Як видно з результатів, мережа MRCNN здійснює ефективне розпізнавання пошкодження шкіри, вказуючи область пошкодження і пропонуючи точну попиксельну маску. Дана мережа також дозволяє відстежувати зростання пошкодження.

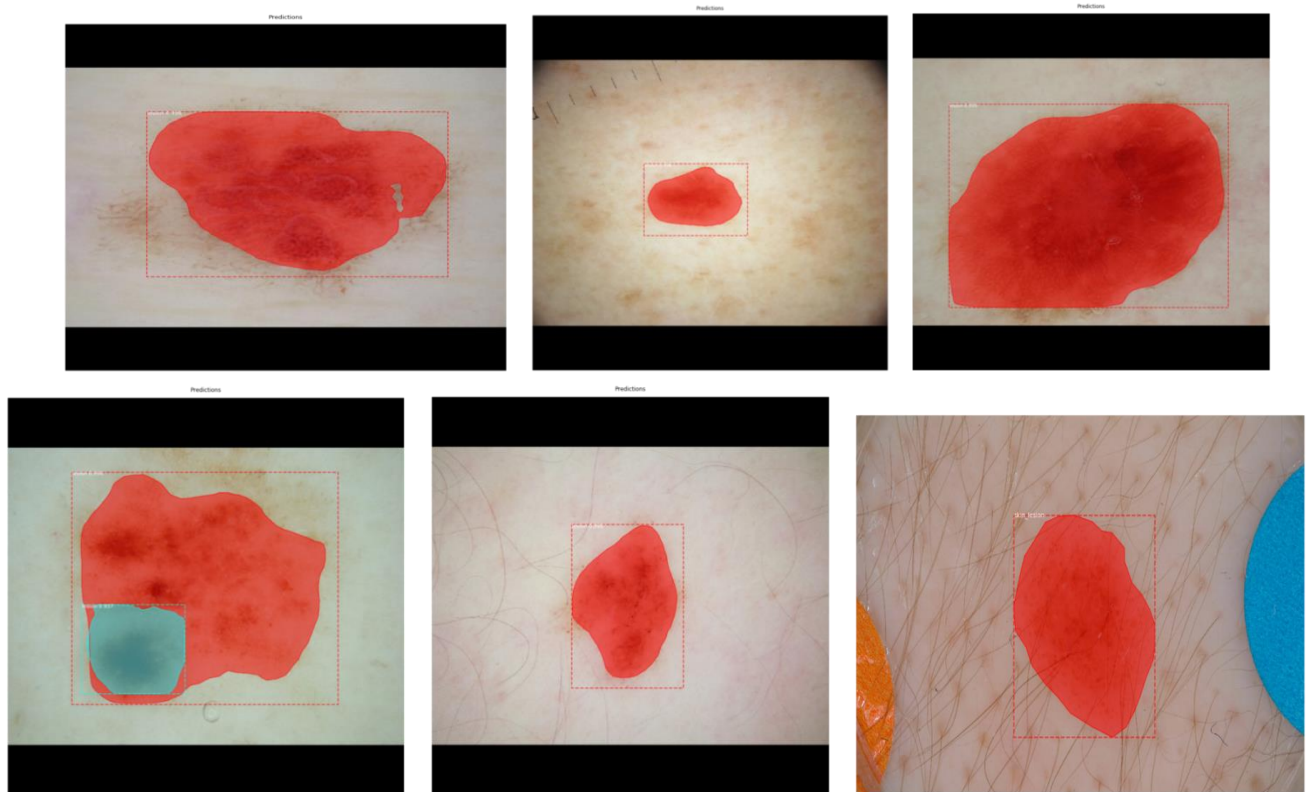


Рисунок 4.9 – Результати розпізнавання

Як видно з рис 4.9, мережа розпізнала два регіони – первинний і вторинний, який є результатом зростання новоутворення.

Експеримент 2. В даному експерименті крім вирішення завдання виявлення шкірного ушкодження вирішувалося завдання класифікації захворювання (доброякісна пухлина або злоякісна). Для навчання використовувалася база фотографій з шкірними ушкодженнями з Експерименту 1. До цієї основи додавалася маска, що дозволяє визначити межі пошкодження і опис, що містить інформацію про тип захворювання. Приклад зображень, використаних для навчання представлений на рис. 4.10.

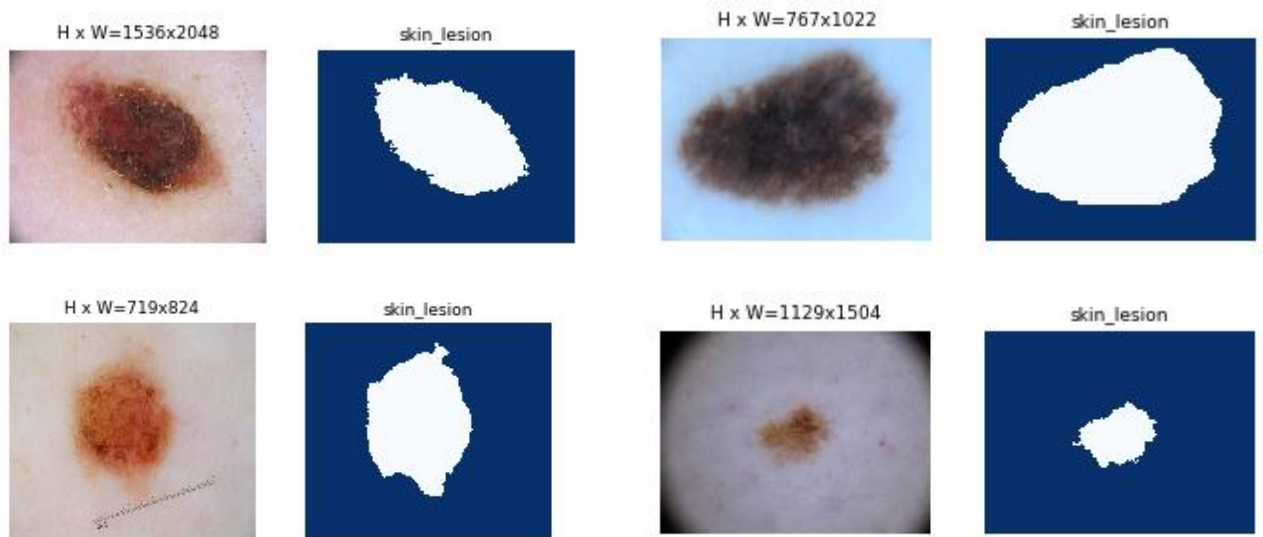


Рисунок 4.10 – Приклади зображень, використаних для навчання

Результати класифікації представлені на рис. 4.11. Зеленим кольором позначалися ушкодження, що не були розпізнані мережею як злоякісні, червоним – позначені пошкодження, які з високою ймовірністю є злоякісними. На деяких зображеннях ймовірність злоякісності новоутворення була близька до порогового рівня, але не перевищувала його. Такі утворення позначалися блакитним кольором. Точність класифікації після 70 епох навчання склала 85,63%.

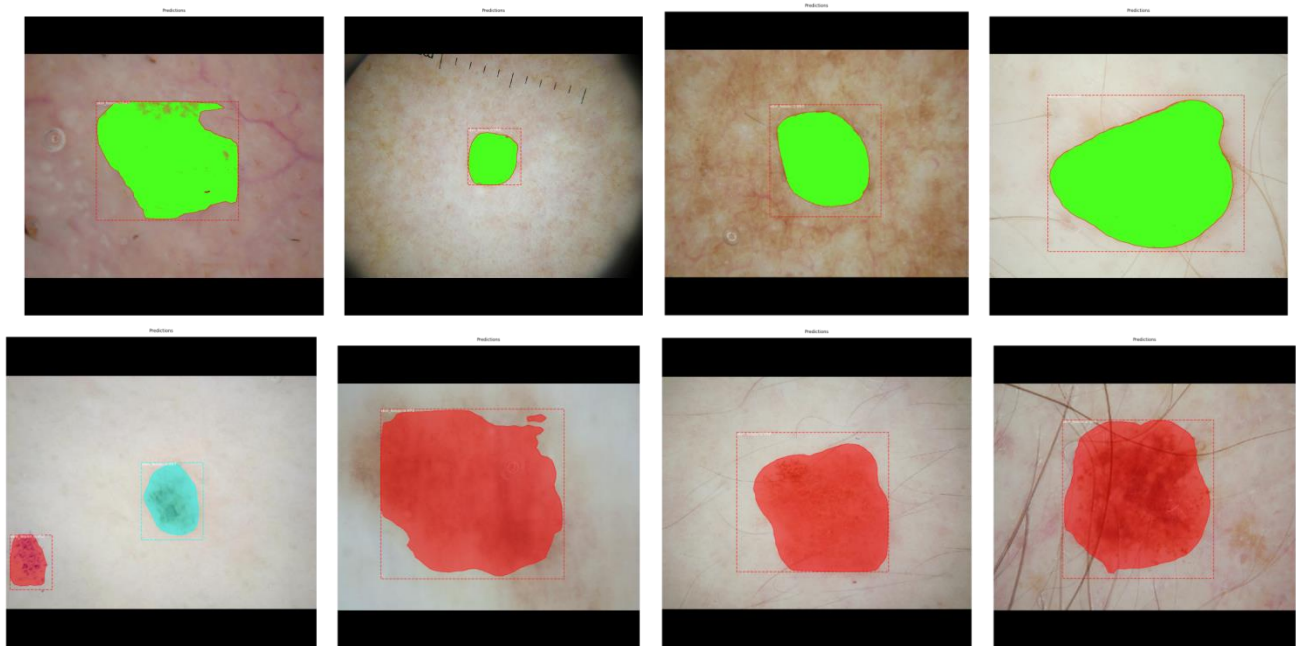


Рисунок 4.11 – Результати класифікації

4.3. Нейромережева експертна система медичної діагностики первинної меланоми шкіри

4.3.1. Завдання діагностування меланоми шкіри

Завдання медичної діагностики полягає у визначенні можливих діагнозів хворого на основі знань предметної області та даних його обстеження, до яких відносять значення ознак (в моменти їхнього спостереження), значення анатомо-фізіологічних особливостей (постійні в часі) і значення подій, що відбулися (в моменти, коли вони відбувалися).

Постановка завдання може бути сформульована таким чином. Дано медичні знання, що задовольняють обмеженням цілісності знань, – знання про спостереження (ознаки, події, анатомо-фізіологічні особливості пацієнта), про захворювання, причинно-наслідкові зв'язки між захворюваннями і спостереженнями, а також результати спостережень хворого – анатомо-фізіологічні особливості, що спостерігалися і результати їх спостереження (значення), ознаки, що спостерігалися, моменти їх спостереження і їх значення в ці моменти, події, що відбулися, моменти, в які вони відбувалися, і їх значення в ці моменти. Потрібно встановити діагноз пацієнта (безліч захворювань, якими хворий пацієнт), а також побудувати пояснення діагнозу – вказати причину кожного захворювання (етіологію або ускладнення) і пояснити всі значення ознак (нормальними реакціями, реакціями на вплив подій, клінічними проявами захворювань і т.д.). Зв'язки між тим, що дано, і тим, що потрібно знайти, визначаються моделлю онтології.

Така модель враховує велику кількість зв'язків між процесами, що відбуваються в організмі хворого. При такому підході очікується, що алгоритм для вирішення сформульованої вище задачі медичної діагностики, що аналізує всі ці зв'язки, матиме високу обчислювальну складність. Розглянемо окрему постановку задачі медичної діагностики, з наступними обмеженнями:

– діагноз пацієнта може включати не більше одного захворювання (пацієнт хворий одним захворюванням або здоровий). В роботі предметну область складають шкірні новоутворення, а саме: меланома, невус і лінтіго;

– пацієнт обстежується лікарем з моменту підозри на виникнення захворювання;

– для спрощення алгоритму не розглядається підзавдання пошуку причини захворювання;

– у кожної хвороби є набір ознак або симптомів.

У таблиці 4.2 стовпці позначені назвами хвороб, а рядки - назвами симптомів. На перетині відповідних стовпців і рядків стоїть знак "+", якщо відповідь на питання стверджувальна, "-" – заперечувальна. Порожня клітинка таблиці говорить про те, що з даного симптому взагалі не можна визначити дану хворобу. Одні і ті ж симптоми можуть бути ознаками різних хвороб, тому деякі з них (симптомів) перекриваються.

Таблиця 4.2 – Аналіз предметної області

Симптом	Тип шкірного новоутворення		
	Невус	Меланома	Лінтіго
А. Пляма з'явилася в дитинстві?	+	-	-
Б. Пляма з'явилася, після того, як зійшов загар?			+
В. Забарвлення неоднорідне?		+	-
Г. Забарвлення з віком темнішає?	+		
Д. Забарвлення світло коричневе?			+
Е. Присутні білі і рожеві тони?	+		
Ж. Межі чіткі?	+		
З. Розмір змінюється?	-	+	-
И. У родичів була меланома?	-	+	
К. Фототип шкіри 1 або 2?		+	+

Завдання діагностування меланоми шкіри зводиться до вирішення задачі класифікації, при цьому вона визначається наступним чином. Є множина об'єктів X , не обов'язково кінцева, а також множина $Y = \{y_1, y_2, \dots, y_Q\}$, що складається з Q класів об'єктів. Кожен клас y_i представлений деяким описом F_i , що має деяку внутрішню структуру. Процедура класифікації f об'єктів $x_i \in X$ полягає у виконанні перетворень над ними, після яких або робиться висновок про відповідність x_i однієї зі структур F_i , що означає віднесення x_i до класу y_i , або висновок про неможливість класифікації x_i . Стосовно до шкірних захворювань, елементами множини Q є описи захворювань меланома, невус і лінтіго. Загальна модель класифікатора представлена алгебраїчною системою такого вигляду

$$K = \langle X, Q, F, K_Q, f \rangle \quad (4.1)$$

де X – множина об'єктів (векторів опитування), що підлягають класифікуванню, Q – множина діагностичних класів, F – множина описів, K_Q – відношення на $Q \times F$, f – операція класифікування виду $x \rightarrow Q$. Відношення K_Q має властивість: $\forall x_i \in Q \exists! F_i \in F : (x_i, F_i) \in K_Q$, тобто класу відповідає єдиний опис. Крім сформульованої задачі класифікації визначається завдання навчання класифікатора, під яким мається на увазі часткове або повне формування Q, F, K_Q , і f на основі деяких апіорних даних. Описи класів F являють собою багатовимірні вектори дійсних чисел, закладені в синаптичних вагах штучних нейронів, а процедура класифікації f характеризується способом перетворення вектора опитування x_i до аналогічного вектору, видом функції активації нейронів, а також топологією мережі.

4.3.2. Архітектура ЕС діагностики первинної меланоми шкіри

В останні роки в якості засобу для зниження обчислювальної складності нейромережових алгоритмів і підбору оптимальної архітектури нейромережі застосовують моделі колективів нейромереж, в рамках яких кілька нейронних мереж об'єднуються і використовуються спільно для вирішення загального завдання. Дані

нейронні мережі так само називають модульними нейронними мережами (МНМ) або гібридними нейронними мережами (ГНМ). У мережах такого типу обчислення, які виконуються, розподіляються на декілька підсистем, які обробляють різні вхідні сигнали і не перетинаються в своїй роботі один з одним [115–124].

Для вирішення завдання діагностики меланоми запропонована модульна нейромережева експертна система, представлена на рис. 4.12

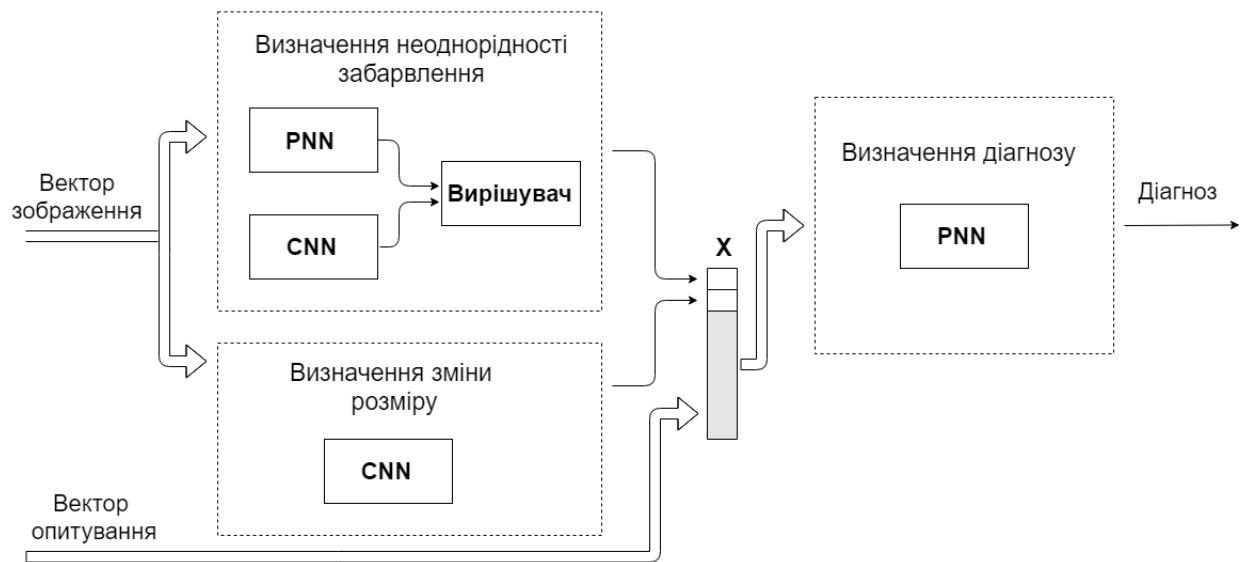


Рисунок 4.12 – Архітектура ЕС діагностики первинної меланоми шкіри

Нейромережева ЕС складається з 3х основних модулів:

- модуль визначення неоднорідності забарвлення;
- модуль визначення зміни розміру;
- модуль постановки діагнозу.

Визначення неоднорідності є надзвичайно важливим фактором в постановці діагнозу. Для вирішення даної задачі запропоновано модуль визначення неоднорідності забарвлення, що складається з ансамблю мереж двох типів PNN і CNN. Експерт являє собою окрему ШНМ, що входить в модуль визначення забарвлення.

Одним з основних факторів, що роблять істотний вплив на ефективне формування ансамблю є знаходження оптимального балансу між точністю та

різноманітністю його членів. Ансамблі схильні давати результати краще, якщо є суттєва відмінність моделей (членів ансамблю). Зазвичай під відмінністю розуміють ступінь некорельованості помилок моделей, що входять до складу ансамблю. Ансамбль показує якість не гірше будь-якого з його членів, а якщо члени здійснюють незалежні помилки, то ансамбль працює значно краще за своїх членів. У нашому випадку ця різниця досягається за рахунок принципово різних архітектур НМ входять в ансамбль.

Наступним за значимістю фактором, що впливає на ефективність ансамблю є схема об'єднання голосів моделей, що входять до його складу. Для формування узагальненої відповіді в блоці вирішувача, запропоновано використовувати зважене підсумовування результатів класифікації. Таким чином узагальнений результат знаходиться як зважена сума відповідей всіх членів:

$$Y = \sum_{i=1}^k g_i y_i, \quad (4.2)$$

де y_i – відповідь i -го експерта, $i \in [1, k]$, k – число експертів–нейромереж, що входять в колектив, g_i – внесок i -го експерта в остаточний результат (вага експерта). У нашому випадку, експериментальним шляхом було встановлено такі ваги: 0.41 для PNN і – 0.59 для CNN.

Модуль визначення зміни розміру побудований на базі CNN мережі, які як було показано раніше, оптимально підходять для вирішення подібних завдань. На вхід мережі подаються 2 зображення, зроблені в різні моменти часу, мережа виробляє сегментування області шкірного новоутворення і визначає збільшується пляма чи ні.

Значення отримані в результаті роботи модуля визначення забарвлення і модуля визначення зміни розмірності доповнюють вектор X , отриманий в результаті опитування пацієнта. Цей вектор надходить на вхід модуля визначення діагнозу, який реалізований на базі PNN мережі.

Таким чином, склад і взаємозв'язок компонентів нейромережевої експертної системи медичної діагностики S в загальному вигляді можна виразити як перетворення вхідних значень X у вихідні величини Y :

$$S \subset X \times Y \quad (4.3)$$

де $X(i) = [x_1(i), x_2(i), \dots, x_n(i)]$ – вектор опитування; $Y = \{y_1, y_2, \dots, y_Q\}$ – множина класів.

В результаті роботи, симптоми захворювання пацієнта на основі вектора опитування $X(i)$ і обробці зображень класифікуються завдяки системі S при заданому числі еталонних образів Q , до яких відноситься захворювання.

4.3.3. Функціонування експертної система медичної діагностики первинної меланоми шкіри

Процес визначення діагнозу виглядає наступним чином [124–129]. Спілкування з пацієнтом проходить в формі консультації. Обстежуваному пропонується відповісти на питання. У процесі консультації виявляються симптоми захворювання. На підставі цих відповідей, а також результатів обробки знімків шкірних новоутворень проводиться визначення діагнозу.

Формування вектора опитування відбувається наступним чином. У процесі консультації пацієнт відповідає на питання, які зведені в таблицю 4.1. Під час упорядкування "вектора опитування", якщо при діагностиці отримуємо відповідь "Так", то відповідному компоненту x_i вектора X присвоюється "1", якщо – "Ні", присвоюється "-1", якщо "Не знаю" – "0". Два компонента "вектора опитування", що відповідають за неоднорідність забарвлення і збільшення розміру, формуються в результаті аналізу знімків хворого.

Формування "вектора опитування" в процесі відповіді пацієнта на питання можна представити у вигляді схеми, фрагмент якої зображений на рис.4.13.

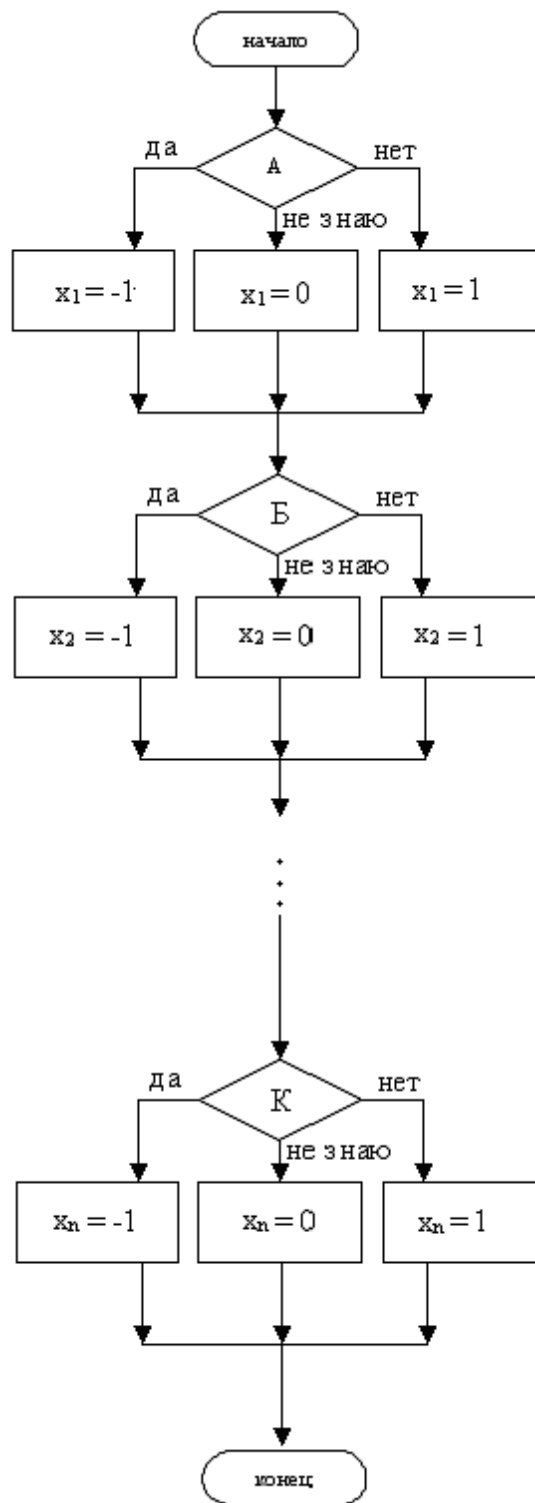


Рисунок 4.13 – Схема формування "вектора опитування"

На схемі буквами позначені питання (див. таб. 4.2):

А. Пляма з'явилася в дитинстві ?;

В. Пляма з'явилася, після того, як зійшов загар?

І т.д.

Таким чином, в процесі консультації формується "вектор опитування" $X = [x_1, x_2, \dots, x_n]^T$, який є вхідним вектором модульної нейронної мережі. Нейронна мережа виробляє класифікацію цього вектора до одного з чотирьох класів - "меланома", "лінтиго", "невус", "хвороба не визначена". Потім проводиться обробка цієї інформації і висновок в зручному для користувача вигляді.

Інтерфейс нейромережевої експертної системи представлений на рисунку 4.14.

Рисунок 4.14–Інтерфейс нейромережевої експертної системи

Для визначення діагнозу захворювання лікар проводить опитування пацієнта. Після чого, після натискання кнопки "Визначити діагноз", у вікні зліва внизу відображається діагноз захворювання (рис. 4.15).

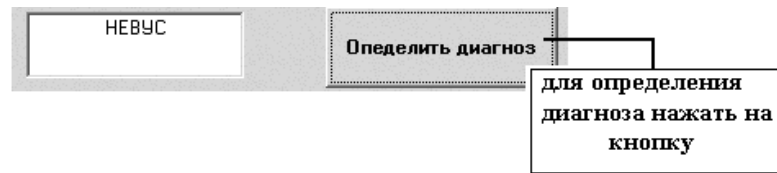


Рисунок 4.15 – Відображення діагнозу захворювання

Якщо в процесі роботи виникають труднощі, то для отримання допомоги по роботі з програмою необхідно натиснути кнопку "Допомога". Так само є можливість отримати довідкову інформацію про захворювання. Для цього потрібно натиснути відповідну кнопку (рис. 4.16).



Рисунок 4.16 – Отримання довідкової інформації про захворювання

Система дозволяє обробляти знімки ділянок ураженої шкіри. Знімок ділянки шкіри зроблений при першому відвідуванні лікаря потрібно завантажити в верхнє вікно відображення зображень, а останній знімок – в нижнє вікно відображення зображень (рис 4.17).

Для завантаження знімка необхідно натиснути кнопку "Відкрити", після чого вибрати необхідний знімок (рис. 4.17). Можна обчислити процентне співвідношення розмірів плями на початковій стадії захворювання і зараз, що дозволяє лікарю визначити прогресує захворювання чи ні.

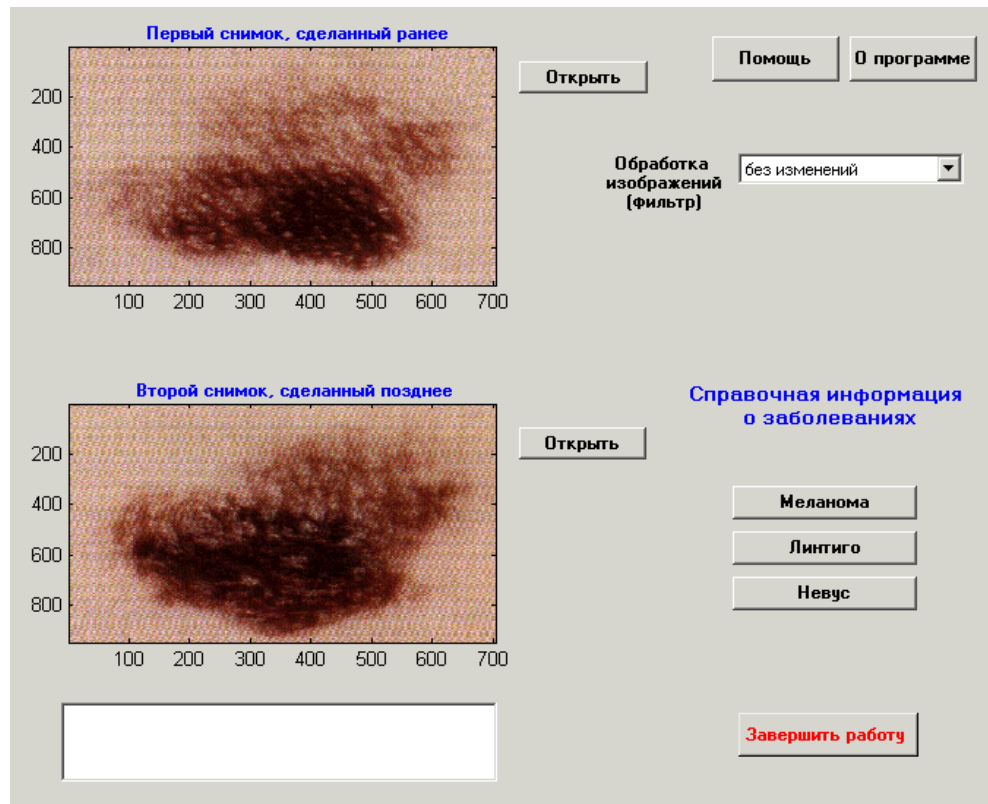


Рисунок 4.17 – Інтерфейс завантаження зображень

В процесі тестування МНМ отримані наступні результати (А – меланома, В – лінтіго, С – невус, Е – новоутворення не розпізнається):

Таблица 4.3 – Результати тестування

Номинальный класс	Кількість тестових прикладів	Результат розпізнавання			
		класс			
		А	В	С	Е
А	3000	2631	0	0	369
В	3000	72	2518	101	309
С	3000	53	0	2947	0

Точність складає близько 89%.

4.4. Висновки по розділу 4

1. Здійснено імітаційне моделювання нейромережових методів розпізнавання зміни параметрів зображень на прикладі меланоми з використанням мереж PNN і ЗНМ, на основі результатів яких визначені вимоги до параметрів мереж. Показано,

що мережа PNN є досить ефективною при класифікації об'єктів, що належать двом класам, а ЗНМ успішно вирішує завдання класифікації при наявності більшої кількості класів.

2. Розроблено узагальнену архітектуру і компоненти нейромережевої експертної системи діагностування меланоми, яка на відміну від існуючих аналогів, використовує модуль, що дозволяє визначити неоднорідність забарвлення, а також модуль дозволяє відстежити збільшення розміру.

3. Здійснено імітаційне моделювання нейромережевих методів розпізнавання зміни параметрів зображень на прикладі меланоми. Програмно реалізована нейромережева система медичної діагностики меланоми, яка дозволяє діагностувати збільшення розміру шкірного новоутворення, визначати неоднорідність забарвлення і допомагає в постановці діагнозу лікаря.

4. Розроблено нейромережевий метод контролю зміни розмірів шкірного утворення на основі мережі CNN, який дозволяє збільшити точність контролю зміни розмірів спостережуваних пігментних новоутворень шкіри для ранньої діагностики меланоми.

5. Розроблено нейромережевий метод розпізнавання неоднорідності забарвлення шкірних змін на основі мережі PNN, який дозволяє підвищити ефективність класифікації поліхромії пігментних новоутворень шкіри.

6. Запропоновано колективна НМ для визначення неоднорідності забарвлення, яка складається з ансамблю PNN і CNN, що дозволяє:

- підвищити точність діагностування за рахунок того, що експерти, яких навчають окремо, будуть сходитися до різних локальних мінімумів поверхні помилок, в результаті чого комбінація їх вихідних сигналів призведе до підвищення ефективності мережі;

- зменшити час навчання за рахунок зменшення параметрів, що настроюються;

- знизити ризик надмірного навчання в разі, коли кількість параметрів, що налаштовуються істотно більше розміру безлічі прикладів, що навчаються.

ВИСНОВКИ

У дисертаційній роботі на основі отриманих теоретичних і експериментальних досліджень вирішена актуальна науково-практична задача побудови нейромережевих методів розпізнавання і класифікації зміни властивостей зображень, які на відміну від існуючих, зменшують помилку розпізнавання і збільшують точність класифікації зображень в умовах апіорної і поточної невизначеності і наявності завад.

При цьому отримано наступні основні результати:

1. Проведено аналіз стану проблем розпізнавання, класифікації, кластеризації та обробки зображень. Розглянуто питання застосування нейронних мереж для рішення цих завдань і описані основні типи ШНМ, сформульована мета і задачі дослідження.

2. Розглянуто методи визначення зміни характеристик зображення і вперше запропоновано метод контролю зміни розмірів об'єкта за допомогою шаблону з використанням властивостей перетворення подібності об'єктів, що дозволяє підвищити точність вимірювань.

3. Одержав подальший розвиток нейромережевий метод визначення неоднорідності забарвлення об'єкта, заснований на використанні розділеного на два класи навчального набору даних, що дозволяє зменшити вимоги до ШНМ і використовувати мережу з двома вихідними нейронами.

4. Вперше розроблено процедуру навчання БШП, що представляє собою матричні варіанти процедури Качмажа (Уідроу-Гоффа), запропонована її модифікація, яка містить зону нечутливості, і розглянуті процедури налаштування цієї зони, що забезпечило отримання робастних оцінок шуканих параметрів мережі.

5. Удосконалено метод адаптації багат шарового персептрона на багатопроцесорні системи, що дозволило отримати оцінку потенційно можливого виграшу в продуктивності в залежності від використовуваних ресурсів, а також розмірності досліджуваного зображення.

6. Отримав подальший розвиток нейромережевий метод класифікації зображень на основі згортальних нейронних мереж з використанням в різних шарах різних функцій активації і робастного навчання параметрів мережі, що дозволило підвищити точність класифікації.

7. Здійснено імітаційне моделювання нейромережевих методів розпізнавання зміни параметрів зображень на прикладі меланоми з використанням мереж PNN і СНС, на основі результатів яких розроблені нейромережевий метод контролю зміни розмірів шкірного утворення і розпізнавання неоднорідності забарвлення шкірних змін, що дозволяють збільшити точність контролю зміни розмірів спостережуваних пігментних новоутворень шкіри для ранньої діагностики меланоми і підвищити ефективність класифікації поліхромії пігментних новоутворень шкіри.

8. Запропонована і програмно реалізована нейромережева система визначення неоднорідності забарвлення, яка складається з ансамблю PNN і CNN, що дозволяє:

- підвищити точність діагностування за рахунок того, що ті експерти, яких навчають окремо, будуть збігатися до різних локальних мінімумів поверхні помилок, в результаті чого комбінація їх вихідних сигналів призведе до підвищення ефективності мережі;

- зменшити час навчання за рахунок зменшення параметрів, що настроюються;

- знизити ризик надмірного навчання в разі, коли кількість параметрів, що налаштовуються, істотно більше розміру безлічі прикладів, що навчаються.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Хайкин С. Нейронные сети: полный курс / С. Хайкин / Пер. с англ. – М.: Вильямс, 2006. – 1104 с.
2. Осовский С. Нейронные сети для обработки информации / С. Осовский / Пер. с польского. – М.: Финансы и статистика, 2002. – 344 с.
3. Руденко О.Г. Основы теории искусственных нейронных сетей / О.Г. Руденко, Е.В. Бодянский. – Харьков: ТЕЛЕТЕХ, 2002. – 317 с.
4. Круглов В.В. Искусственные нейронные сети. Теория и практика. / В.В. Круглов, Борисов В.В.– М.: Горячая линия – Телеком, 2001. – 382 с.
5. Cichocki A. Neural networks for optimization and signal processing. / A. Cichocki, R. Unbehauen – New York: John Wiley & Sons, 1998. – 521p.
6. Cybenko G. Approximation by superposition of a sigmoidal function / G. Cybenko / Math. of controls, Signals & Systems. – 1989. – 2. – P.303-314.
7. Бодянский Е.В. Искусственные нейронные сети: архитектура, обучение, применение / Е.В. Бодянский, О.Г. Руденко. – Харьков: ТЕЛЕТЕХ, 2004. – 372 с.
8. Уоссермен Ф. Нейрокомпьютерная техника / Ф. Уоссермен. – М.: Мир, 1992. – 184 с.
9. Ham F.M. Principles of Neurocomputing for Science and Engineering/ F.M. Ham, I.Kostanic. – N.Y.: Mc Graw-Hill Inc., 2001. – 468p.
10. Patterson D. Artifical Neural Networks, Theory and Application / Patterson D. – Singapur: Prenice Hall Inc., 1996. – 497p.
11. LeCun, Y. Handwritten digit recognition with a backpropagation neural network IY. LeCun, B. Boser, J. S. Denker, D. Henderson, R. Howard, W. Hubbard, L. Jackel //Advances in Neural Information Processing Systems. 1990. № 2. P. 396-404.
12. LeCun, Y. Convolutional networks for images, speech, and timeseries I Y. LeCun, Y. Bengio II The Handbook of Brain Theory and Neural Networks. 1995. P. 255-258.

13. Goodfellow I. Deep learning / I. Goodfellow, Y. Bengio, A. Courville: –The MIT Press, 2016. – 800 p. https://www.deeplearningbook.org/front_matter.pdf
14. Шолле Ф. Глубокое обучение на Python. / Ф. Шолле – СПб.: Питер, 2018. – 400 с.:
15. Горбань А.Н. Нейроинформатика. / А.Н.Горбань, В.Л.Дунин-Барковский, А.Н.Кирдин, Е.М.Миркес и др.– Новосибирск: Наука. Сибирское предприятие РАН, 1998. - 296с.
16. Анисимов Б. В. Распознавание и цифровая обработка изображений: Учеб. пособие для студентов вузов. / Б. В. Анисимов, В. Д. Курганов, В. К. Злобин — М.: Высш. шк., 1983. — 295с.
17. Вороновский Г.К. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности. / Г.К. Вороновский и др – Х.: Основа, 1997. – 112с.
18. Барский А.Б. Нейронные сети: распознавание, управление, принятие решений. / А.Б. Барский – М.: Финансы и статистика, 2004. – 176с.
19. Вятчинин Д.А. Нечеткие методы автоматической классификации. / Д.А. Вятчинин – Мн. УП Технопринт, 2004. – 219с.
20. Яхьяева Г.Э. Нечеткие множества и нейронные сети. / Г.Э. Яхьяева – М.: БИНОМ, 2006. – 316с.
21. Круглов В.В. Искусственные нейронные сети. Теория и практика. 2-е изд., / В.В. Круглов, В.В. Борисов – М.: Горячая линия – Телеком, 2002. – 382с.
22. Миркес Е.М. Нейроинформатика: Учеб. пособие для студентов / Е.М. Миркес – Красноярск: ИПЦ КГТУ, 2002. - 347 с.
23. Hornik K. Multilayer feedforward networks are universal approximators / K. Hornik, M. Stinchcombe, H. White // Neural Networks. – 1989. – N2. – P.359-366.
24. Zhu Q.M. Fast orthogonal identification of nonlinear stochastic models and radial function neural networks / Q.M. Zhu, S.A. Billings // Int.J.Control. – 1996. – Vol.64. – N5. – P.871-886.

25. Li Y. Analysis of minimal radial basis function network algorithm for real-time identification of nonlinear dynamic systems / Y. Li, N. Sundararajan, P. Saratchandran // IEEE Proc. – Control Theory Appl. – 2000. – v.147. – №4. – P. 476 - 484.
26. Powell M.J.D. Radial Basis Functions for Multivariable Interpolation: A review.// Proc. of IMA Conf. on Algorithms for the Approximation of Functions and Data, Shrivenham, UK. – 1985. – P.143-167.
27. Broomhead D.S. Multivariable Function Interpolation and Adaptive Networks / D.S. Broomhead, D. Lowe / Complex System. – 1988. – 2. – P.321-355.
28. Lowe D. Adaptive Radial Basis Function Nonlinearities and the Problem of Generalization / D. Lowe // Proc. of IEE Int. Conf. of Artificial Neural Networks, London, UK. – 1989. – P.171-175.
29. Poggio T. Networks for Approximation and Learning/ T. Poggio, F. Girosi // Proc. of IEEE. – 1990. – 78(9). – P.1481-1497.
30. Hartman E. J. Layered neural networks with Gaussian hidden units as universal approximations/ E. J. Hartman, J. D. Keeler, J. Kowalski J. / Neural Computation. – 1990. – 2. – P. 210-215.
31. Park J. Universal approximation using radial-basis-function networks / J. Park, I. W. Sandberg / Neural Computation. – 1991. – 3. – P. 246-257.
32. Leonard J. A. Using radial basis functions to approximate a function and its error bounds/ J. A. Leonard, M. A. Kramer, L.H. Ungar / IEEE Trans. on Neural Networks. – 1992. – 3. – P. 614-627.
33. Sunil E. V. T. Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems/ E. V. T. Sunil, C. Sh. Yung // IEEE Trans. on Neural Networks. – 1994. – 5. – P. 594-603.
34. Kohonen T. Self-Organizing Maps. / T. Kohonen - Berlin. Heidelberg: Springer, 1995. - 501 p.
35. Fukushima K. Cognitron: A self-organizing multilayered neural network / K. Fukushima / Biolog. Cybernetics. -1975.- vol. 20.- Pp. 121-136.

36. Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position / K. Fukushima / *Biolog. Cybernetics*.- 1980.- vol. 36.- Pp. 193- 202.

37. Fukushima, K. Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. I K. Fukushima, S. Miyake II *Pattern recognition*. 1982. № 15(6). P. 455-469.

38. Fukushima, K. Neocognitron: A model for visual pattern recognition. / In M. Arbib, editor // *The Handbook of Brain Theory and Neural Networks*. MIT Press. 1995.

39. Vincent, P. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion / P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P. A. Manzagol / *The Journal of Machine Learning Research*. – 2010. – Vol. 11. – P. 3371–3408.

40. Hinton G.E. Reducing the dimensionality of data with neural networks / G.E. Hinton, R.R. Salakhutdinov / *Science*. 2006. - VOL 313.- P/504-507.

41. Specht D.F. Probabilistic neural networks for classification, mapping or associative memory/ D.F. Specht // *Proc. of the IEEE Conf. on Neural Network*, San Diego, 1988.-v.1.-2.- P. 568-576.

42 Specht D.F. A general regression neural network / D.F. Specht / *IEEE Trans. on Neural Networks*. – 1991. – Vol.2. – N6. – P.568–576.

43. Parzen E. On estimation of a probability density function and mode / E. Parzen / *Ann. Math. Stat.* –1962. –33. –P.1065–1076.

44. Cacoullos T. Estimation of multivariable density/ T. Cacoullos / *Ann. Math. Stat.* – 1966. –18. – P.179 – 189.

45. Leshem Z. Tuvia Apparatus and method for measurement and temporal comparison of skin surface images / Z. Leshem / *United States Patent No 6,215,893*.- April 10, 2001.

46. Декларационный патент Украины № 52992 АпоМПКГ01С 11/00 от 15.01.2003

47. Craine E.R. Digital skin lesion imaging system and method / . E.R.Craine, L. Brian /United States Patent No 7,162,063.-January 9, 2007.
48. АксакН.Г. Нейросетевая медицинская система принятия решений / Н.Г. Аксак, И.В. Новосельцев, А.П. Лучко // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: Матеріали восьмої міжнародної науково-технічної конференції. Полтава -Баку – Харків - Жиліна. – 2018.-С.13.
49. АксакН.Г. Спосіб контролю зміни розмірів об'єкта за допомогою шаблону / Н.Г. Аксак, М.В. Кушнар'ов, І.В. Новосельцев, А.Ю. Тихун // Патент на винахід № 84380 АпоМПК (2006) G01C 11/00 від 10.10.2008. Бюл. №19.
50. Solla S.A. Accelerated learning in layered neural networks / S.A. Solla, E. Levin, V. Fleisher / Complex Systems. – 1988. –Vol.2. – N2. – P.625-640.
51. Lisboa P.J.G. Complete solution of the local minima in the XOR problem / P.J.G. Lisboa, S.J. Perantonis / Network. – 1991. – Vol.1. – N2. – P.119-124.
52. Gori M. On the problem of local minima in backpropagation / M. Gori, F. Tesi / IEEE Trans. on Pattern Analysis and Machine Learning. – 1992. – Vol.1. –N14. – P.76-86.
53. Poston T. Local minima and backpropagation / T. Poston, C.N. Lee, Y. Choie, Y. Kwon // Proc. of IJCNN '91, Seattle, WA, July, 1991. – II – P.173-176.
54. Yu X. Can backpropagation error surface not have local minima?/ X. Yu / IEEE Trans. Neural Networks. – 1992. – Vol.3. –N6. – P.1019-1021.
55. Moller M. Efficient training of feed-forward neural networks / M. Moller // Ph.D. Thesis, Daimi PB-464, Computer Science Department, Aarhus University, 1993. – 151 p.
56. Rumelhart, D. E. Learning internal representations by error propagation. / D. E. Rumelhart, G. E. Hinton, R. J Williams / Parallel distributed processing, MIT Press.- 1986. - Vol. 1.- P. 318-362.
57. Rumelhart, D. Learning representations of back-propagation errors. / D. E. Rumelhart, G. E. Hinton, R. J Williams /Nature.- 1986.- №323. -P. 533-536.

58. Krizhevsky A., Sutskever I., Hinton G.E.. ImageNet Classification with Deep Convolutional Neural Networks. — 2012. — URL: <https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>.
59. He K., Mask R-CNN / K He, G. Gkioxari, P Dollár, R Girshick // IEEE International Conference on Computer Vision (ICCV), 2017.- 2980-2988
60. Sadowski P. Notes on backpropagation .
<https://www.ics.uci.edu/~pjsadows/notes.pdf>, 2016. Online; accessed 11 June 2016.
61. Shepherd A.J. Second-order methods for neural networks: Fast and reliable training methods for multi-layer perceptrons / A.J. Shepherd. — New York: Springer, 1997. — 146 p.
62. Поляк Б.Т. Введение в оптимизацию / Б.Т. Поляк. — М.: Наука, 1983. — 384с.
63. Wilson C. The Marginal Value of adaptive gradient methods in machine learning / 2017: <https://arxiv.org/abs/1705.08292>.
64. Sutton R.S. Adapting bias by gradient descent: An incremental version of delta-bar-delta / R.S. Sutton // Proceedings of the 10th National Conference on Artificial Intelligence.- MIT Press, 1992. -P. 171-176
65. S. Ruder. An overview of gradient descent optimization algorithms. CoRR, abs/1609.04747, 2016. <http://ruder.io/optimizing-gradient-descent/>
66. Аксак Н. Г. Распознавание изменения размера и цвета изображения на основе сверточной нейронной сети/ Н. Г. Аксак, А.А. Бессонов, И.В. Новосельцев, О.Г. Руденко // Бионика интеллекта: Научн.-техн. журнал. — 2018. — №2 (74). — С.131-136.
67. Le T. A variational approach to reconstructing images corrupted by Poisson noise/ T. Le, R. Chartrand, T. J. Asaki / J. Math. Imaging Vision, 2007. To appear.
68. Vogel C. R. Iterative methods for total variation denoising / C. R. Vogel, M. E. Oman / SIAM J. Sci. Comput.- 1996.- vol. 17.- no. 1.- P. 227–238.
69. Dobson D. Convergence of an iterative method for total variation denoising / D. Dobson, C. Vogel / SIAM J. Numer. Anal.- 1997.- Vol. 34.- P. 1779–1791.

70. Rudin L. Nonlinear total variation based noise removal algorithms/ L. Rudin, S. Osher, E. Fatemi / *Physica D.*- 1992.-Vol. 60.-P. 259–268.
71. Green M. Statistics of images, the TV algorithm of Rudin-Osher-Fatemi for image denoising and an improved denoising algorithm,/ M. Green / *CAM Report 02-55*, UCLA, 2002.
72. Jonsson E. Total variation regularization in positron emission tomography / E. Jonsson, S.-C. Huang, and T. Chan // *Tech. Rep. 98-48*, UCLA Group in Computational and Applied Mathematics, 1998.-86 p.
73. Nikolova M. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers / M. Nikolova / *SIAM J. Numer. Anal.*- 2002.- Vol. 40.- P. 965–994.
74. Nikolova M. A variational approach to remove outliers and impulse noise / M. Nikolova / *J. Math. ImagingVision.*- 2004.- Vol. 20.- P. 99–120.
75. Хьюбер П. Робастность в статистике / П. Хьюбер – М.: Мир, 1984, 304 с.
76. Цыпкин Я.З. Основы информационной теории идентификации/ Я.З. Цыпкин – М.: Наука, 1984. – 320 с.
77. Цыпкин Я.З., Поляк Б.Т. Огрубленный метод максимального правдоподобия / Я.З. Цыпкин, Б.Т. Поляк // В кн. «Динамика систем». – Горький, 1977. – Вып. 12. – С. 22-46.
78. Hampel F.R. The influence curve and its role in robust estimation / F.R. Hampel // *J. Amer. Statist. Assoc.* – 1974. – June. – 69. – P. 383-393.
79. Hampel F.R. Robust Statistics. The Approach Based on Influence Functions / F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, W.A. Stahel– N.Y.: John Wiley and Sons, 1986. – 526 p.
80. Chambers J. A Robust Mixed-Norm Adaptive Filter Algorithm / J. Chambers, A. Avlonitis // *IEEE Signal Processing Letters.* – 1997. – Vol. 4. – №2. – P.46-48.
81. М.В. Ульянов. Архитектуры процессоров. – М.: МГАПИ, 2002. – 68 с.

82. В.П. Гергель, Р.Г. Стронгин. Основы параллельных вычислений для многопроцессорных вычислительных систем: Учебное пособие Нижегородский госуниверситета. – Нижний Новгород, 2003 – 184с.
83. В.В. Воеводин, Вл.В. Воеводин. Параллельные вычисления. – С.Петербург: БХВ-Петербург, 2002. – 608 с.
84. И.В.Новосельцев, Н.Г. Аксак. Реализация многослойного персептрона с помощью стандарта OpenMP / 8-я международная научно-практическая конференция "Современные информационные и электронные технологии": Сб. трудов конференции - Одесса, 2007.-С.79
85. Аксак Н.Г., Новосельцев И.В., Солдатов А.С. Адаптация нейросетевых моделей на высокопроизводительные системы с общей памятью - Системні технології. Збірник наукових праць.- Випуск 3 (56).- Том 2.-Дніпропетровськ, 2008.- С.117-122
86. Новосельцев И.В., Аксак Н.Г. Применение многопроцессорных систем для реализации многослойного персептрона - Радиоэлектроника. Информатика. Управление.- №1(19) , Запорожье.-2008.-С.108-110.
87. И.В.Новосельцев, Н.Г. Аксак. Использование параллельных технологий для распознавания образов. // Материалы 11-го юбилейного международного форума «Радиоэлектроника и молодежь в XXI веке». – Харьков: ХНУРЕ, 2007. – с.306.
88. Аксак Н.Г., Верчиков Р.С., Лола О.Р., Новосельцев И.В., Олищук С.А. Распознавание образов на основе нейронных сетей с использованием технологии MPI.- Высокопроизводительные параллельные вычисления на кластерных системах. Материалы шестого Международного научно-практического семинара. Том 1. Санкт-Петербург, 2007.-С.21-25
89. И.В.Новосельцев, Н.Г. Аксак. Использование технологий MPI и OpenMP при реализации многослойного персептрона для решения задачи распознавания образов. // VII международная конференция “Интеллектуальный анализ информации ИАИ-2007”. – К.: Просвіта, 2007. – с.251-258.

90. Аксак Н. Г. Система предоставления медицинских услуг на основе архитектуры cloud-fog-dew / Н.Г. Аксак, И.В. Новосельцев, Д.Н. Росинский // Наука та виробництво: міжвуз. темат. зб. наук. пр. Вип. 19 / ДВНЗ «ПДТУ». –Мариуполь, ПДТУ, 2018. – С. 150-159.
91. Axak N. Cloud-fog-dew Architecture for Personalized Service-oriented Systems / N. Axak, D. Rosinskiy, O. Barkovska, I. Novoseltsev // The 9th IEEE International Conference on Dependable Systems, Services and Technologies, DESSERT'2018, Kyiv, Ukraine – 2018. – P.80 -84.
92. Аксак Н.Г., Модели производительности синтеза нейроалгоритма на высокопроизводительных архитектурах / Н.Г. Аксак, А.Ю. Тихун, И.В. Новосельцев // Труды 8-й Международной конференции. Высокопроизводительные параллельные вычисления на кластерных системах (НПС-2008), изд-во КГТУ, Казань. – 2008. – С.199-202.
93. J. Secretan, M. Georgiopoulos, I. Maidhof, P. Shibly, J. Hecker. Methods for parallelizing the probabilistic neural network on a beowulf cluster computer. // Neural Networks, 2006. – pp.2378 – 2385.
94. Гергель В.П. Теория и практика параллельных вычислений: учебное пособие/В.П. Гергель. – Интернет-университет информационных технологий, 2010. – 423с.
95. Springenberg J.T. Striving for simplicity / J.T.Springenberg, A.Dosovitskiy, T.Brox, M. Riedmiller: ICLR 2015.-14 p. <https://arxiv.org/pdf/1412.6806.pdf>
96. Stutz D. Understanding Convolutional Neural Networks Seminar Report : Fakultät für Mathematik, Informatik und Naturwissenschaften Lehr- und Forschungsgebiet Informatik VIII Computer Vision.-2014.- 23 p.
97. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S.Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467, 2016.
98. Towards Better Analysis of Deep Convolutional Neural Networks
Mengchen Liu, Jiabin Shi, Zhen Li, Chongxuan Li, Jun Zhu, Shixia Liu

99. Kingma, D. P., & Ba, J. L. (2015). Adam: a Method for Stochastic Optimization. International Conference on Learning Representations, 1–13.
100. Нестеров Ю.Е. Метод минимизации выпуклых функций со скоростью сходимости // Докл. АН СССР.— 1983.— Т. 269.- Вып. 3.— С. 543-547.
101. Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. Journal of Machine Learning Research, 12, 2121–2159. Retrieved from <http://jmlr.org/papers/v12/duchi11a.html>
102. Zeiler M. D. (2012). ADADELTA: An Adaptive Learning Rate Method. Retrieved from <http://arxiv.org/abs/1212.5701>
103. Kingma, D. P., & Ba, J. L. (2015). Adam: a Method for Stochastic Optimization. International Conference on Learning Representations, 1–13.
104. Dozat, T. (2016). Incorporating Nesterov Momentum into Adam. ICLR Workshop, (1), 2013–2016.
105. <https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4>
106. Deep Learning for Image Processing Applications //Ed. Hemanth D.Y., Estrella V.V. 2017.-273 p. <http://ebooks.iospress.nl/volume/deep-learning-for-image-processing-applications>
107. Deep learning tutorial, Stanford University.Autoencoders. Available: <http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders/>
108. Сверточная сеть на python. Часть 2. Вывод формул для обучения модели <https://habr.com/ru/company/ods/blog/344116/>
109. Новосельцев И.В. Вероятностные нейронные сети для решения задач медицинской диагностики / И.В. Новосельцев, Н.Г. Аксак, О.Ю.Барковская // Бионика интеллекта: Научн.-техн. журнал. – 2006. – №2(65). – С. 62-65.
110. Барковская О.Ю. Обобщенная модель распознавания антропоморфных объектов / О.Ю. Барковская, И.В. Новосельцев, Н.Г. Аксак // Системні технології. Збірник наукових праць.- Випуск 6 (53).- Дніпропетровськ, 2007.- С. 11-16.

111. Барковская О.Ю. Сравнительный анализ алгоритмов бинаризации полутонового изображения / О.Ю.Барковская, Н.Г. Аксак, И.В. Новосельцев // Автоматизированные системы управления и приборы автоматики, 2005. – Вып. 133. – С. 40-46.
112. Барковская О.Ю. Обобщенная модель распознавания антропоморфных объектов / О.Ю. Барковская, И.В. Новосельцев, Н.Г. Аксак // Системні технології. Збірник наукових праць.- Випуск 6 (53).- Дніпропетровськ, 2007.- С. 11-16.
113. Аксак Н. Г. Распознавание изменения размера и цвета изображения на основе сверточной нейронной сети/ Н. Г. Аксак, А.А. Бессонов, И.В. Новосельцев, О.Г. Руденко // Бионика интеллекта: Научн.-техн. журнал. – 2018. – №2 (74). – С.131-136. 113. ISIC 2016: SkinLesionAnalysisTowardsMelanomaDetection
114. Новосельцев И. В. Система первичной диагностики меланомы кожи / И.В. Новосельцев, Н.Г. Аксак // Бионика интеллекта: Научн.-техн. журнал. – 2010. – №3(74). – С.94-98.
115. Новосельцев І.В. Система віддаленого нейромережного діагностування меланоми / І.В. Новосельцев, Н.Г. Аксак // Збірник наукових праць, ХНУРЕ.– Харків: «Друкарня Мадрид», 2018. – 116-118.
116. Аксак Н.Г. Система интеллектуального анализа медицинских данных/ Н.Г. Аксак, И.В.Новосельцев, Д.Н. Росинский, В.О. Лебедев // Комп'ютерні та інформаційні системи і технології: Матеріали другої міжнародної науково-технічної конференції. Харків. – 2018. – С.128-129.
117. T. Kondo, A.S. Pandya, J.M. Zurada. An application of self-selecting algorithm for optimum neural network architecture to medical image recognition. // Proceeding of the 44th SICE annual conference, Aug. 2005 – pp.1657-1661.
118. N. Kourentzes, K. Barrow, F. Crone. Neural network ensemble operators for time series forecasting.// ExpertSystemswithApplications. – July 2014. – Vol.41, Iss. 9. – ISSN: 0957-4174. – P. 4235–4244

119 Аксак Н.Г. Агентно-ориентированный медицинский Интернет-ресурс / Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Вычислительный интеллект (результаты, проблемы, перспективы): Материалы 2-й Международной научно-технической конференции. – Черкассы: Маклаут, 2013. – С.321.

120. R. Mizuno, N. Aibe, M. Yasunaga, I. Yoshihara. Reconfigurable architecture for probabilistic neural network system. // Field-Programmable Technology (FPT), 2003. – pp.367 – 370.

121. Аксак Н.Г. Мультиагентная модель адаптации медицинского Интернет-ресурса на основе информационного портрета пользователя /Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Системи обробки інформації. Збірник наукових праць. – Х.: ХУПС, 2013. – Вип.2(109). – С. 242-246. (Входит до міжнародних наукометричних баз Index Copernicus, BASE).

122. Новосельцев И.В. Гибридные нейронные сети для распознавания образов / И.В. Новосельцев, Н.Г. Аксак, О.Ю. Барковская // Системы обработки информации. Збірник наукових праць.-2007.- Вып. 3(61).-С.60-64. (Входит до міжнародних наукометричних баз BASE).

123. W. N. Street,Y. Kim. A streaming ensemblealgorithm (SEA) for large-scale classification. // KDD'01 – NY, USA, ACMPress2001, – pp. 377–382.

124. J. C. Giarratano, G. D. Riley. Expert Systems: Principles and Programming. FourthEdition, CourseTechnology, Boston, MA, 2004 -856 pages.

125. Новосельцев И.В. Применение нейронных сетей для построения медицинской экспертной системы / И.В. Новосельцев, Н.Г. Аксак // Международная научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов по материалам Междун. Научн. конференции.- Харьков: ХНУРЭ, 2003,- С.341-342.

126. Новосельцев И.В. Применение нейронной сети при фильтрации СПАМ сообщений / И.В. Новосельцев, Н.Г. Аксак // 10-я Юбилейная международная

научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов. Ч.2. - Харьков: ХНУРЭ, 2004.- С.295-296.

127. Новосельцев, И.В. Модель обработки больших данных для систем здравоохранения / И.В. Новосельцев, А.Н. Тарасенко //Материалы 22-го международного молодежного форума «Радиоэлектроника и молодежь в 21 столетии». – 17-19 апреля 2018. – Харьков. – С. 57-58.

128. D. Piccolo, A. Ferrari, K. et al. Peris. Dermoscopic diagnosis by a trained clinician vs. a clinician with minimal dermoscopy training vs. computeraided diagnosis of 341 pigmented skin lesions: a comparative study. Br J Dermatol 2002. - 481—486p.

129. T. Kondo, A.S. Pandya. Medical image recognition by using GMDH-type neural networks with sigmoid functions. // Proceedings of the international technical conference on circuits and systems, computers and communications, July 2001 – pp.259-264.

130. M. Zeiler and R. Fergus Visualizing and Understanding Convolutional Networks, ECCV 2014

131. Szegedy C., Liu W., Jia Y., Sermanet P., Reed S., Anguelov D., Erhan D., Vanhoucke V., Andrew Rabinovich A.. Going Deeper with Convolutions.— 2014.— URL: <http://arxiv.org/abs/1409.4842>.

132. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proc. International Conference on Learning Representations <http://arxiv.org/abs/1409.1556> (2014).

133. Szegedy C., Ioffe S., Vanhoucke V., Alemi A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. — 2016.— URL: <https://arxiv.org/abs/1602.07261>.

134. Ross Girshick, Jeff Donahue, Trevor Darrell and Jitendra Malik R-CNN: Region-based Convolutional Neural Networks . <https://github.com/rbgirshick/rcnn>

135. Region-based Convolutional Networks for Accurate Object Detection and Segmentation R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.
136. Dai, Y. Li, K. He, and J. Sun. R-FCN: Object detection via region-based fully convolutional networks. In NIPS, 2016.
137. Girshick. Fast R-CNN. In ICCV arXiv:1504.08083, 2015.
138. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks Shaoqing Ren * Kaiming He Ross Girshick Jian Sun / Advances in Neural Information Processing Systems 28 (NIPS 2015)
139. Hui J. Image segmentation with Mask R-CNN.-2018.
https://medium.com/@jonathan_hui/image-segmentation-with-mask-r-cnn-ebe6d793272
140. MaskR-CNN: архитектура современной нейронной сети для сегментации объектов на изображениях <https://habr.com/ru/post/421299/>
141. Waleed A. Splash of Color: Instance Segmentation with Mask R-CNN and TensorFlow <https://engineering.matterport.com/splash-of-color-instance-segmentation-with-mask-r-cnn-and-tensorflow-7c761e238b46>

ДОДАТОК А

Доказ збіжності процедури (2.48).

Запишемо (2.48) відносно помилок навчання $\tilde{\theta}(i) = \theta(i) - \hat{\theta}(i)$ (тут $\theta(i)$ - вектор оптимальних значень параметрів, $\hat{\theta}(i)$ - вектор оцінок параметрів на i -му такті)

$$\tilde{\theta}(k) = \tilde{\theta}(k-1) - \gamma [2\lambda e(k) + (1-\lambda)\text{sign } e(k)] x(k). \quad (\text{A.1})$$

Помноживши обидві частини (A.1) зліва на $\tilde{\theta}^T(k)$, з урахуванням того, що $e(k) = \tilde{\theta}^T(k-1)x(k)$, маємо

$$\begin{aligned} \|\tilde{\theta}(k)\|^2 &= \|\tilde{\theta}(k-1)\|^2 - 4\gamma\lambda e^2(k) - 2\gamma(1-\lambda)e(k)\text{sign } e(k) + \\ &+ 4\gamma^2\lambda^2 e^2(k)\|x(k)\|^2 + 4\gamma^2\lambda(1-\lambda)e(k)\text{sign } e(k)\|x(k)\|^2 + \\ &+ \gamma^2(1-\lambda)^2\|x(k)\|^2. \end{aligned} \quad (\text{A.2})$$

Введемо функцію Ляпунова $V(k) = \|\tilde{\theta}(k)\|^2$ і розглянемо її прирощення $\Delta V(k) = \|\tilde{\theta}(k)\|^2 - \|\tilde{\theta}(k-1)\|^2$.

З (3.48) видно, що так як $\gamma > 0$, прирощення функції Ляпунова буде від'ємним, тобто процедура (2.48) буде сходиться, якщо

$$2(2\lambda e^2(k) + (1-\lambda)|e(k)|) > \gamma(2\lambda e(k) + (1-\lambda)|e(k)|)^2\|x(k)\|^2. \quad (\text{A.3})$$

Таким чином, умови збіжності процедури (2.48) будуть виконуватися, якщо параметр γ задовольняє нерівності

$$0 < \gamma < \frac{2e(k)}{(\lambda e(k) + (1-\lambda)\text{sign } e(k))\|x(k)\|^2}. \quad (\text{A.4})$$

Оптимальне значення параметра γ визначаємо з рівняння, що одержане шляхом диференціювання (A.2) по γ і прирівнювання похідної нулю. Таким чином,

$$\gamma^{onm} = \frac{e(k)}{(\lambda e(k) + (1-\lambda)\text{sign } e(k))\|x(k)\|^2}. \quad (\text{A.5})$$

ДОДАТОК Б

Розвиток архітектури ЗНМ

LeNet

LeNet [11,12,58] - перша сучасна згортальна нейромережа, розроблена в 90-х роках Яном ЛеКуном, навчалася для розпізнавання рукописних символів. Остання версія мережі LeNet-5 демонструвала на той момент вражаючі результати за рахунок реалізації кількох новаторських ідей. Основною відмінністю від неокогнітрона було включення субдискретизуючого maxpooling шару після кожного згортального шару і включення повнозв'язних шарів на виході мережі. З цього моменту згортальні і maxpooling-шари стають серцем сучасних глибоких мереж.

Робота А. Крижевського, І. Суцкевера і Д. Хінтона, яка зіграла істотну роль в популяризації CNN в області комп'ютерного зору. ЗНМ AlexNet в 2012 році вперше в історії виграла конкурс за класифікацією зображень бази ImageNet на 1000 класів з перевагою по точності розпізнавання майже в два рази. Архітектура включала згортальні шари, maxpooling-шари, повнозв'язні шари на виході мережі. Використовувалися методи dropout і локальної нор-малізації. Як функція активації використовувалася функція ReLU. Навчання мережі проходило на двох потужних графічних прискорювачах протягом тижня.

ZF Net.

ЗНМ М. Зеллера і Р. Фергюса (ZF Net) [130] стала переможцем ILSVRC 2013. Дана архітектура була поліпшеною версією AlexNet: тут збільшили розміри середніх згортальних шарів і зменшили крок і розмір фільтра на першому шарі.

GoogLeNet, Inception

2014 році ImageNet Recognition Challenge виграла нейромережа GoogleGoog LeNet [131], на основі першої Inception-архітектури, в якій був зменшений розмір

згортки, паралельно включені невеликі згортки різного масштабу, видалені повнозв'язні шари на виході мережі і замість них включені шари, названі global average pooling. Скорочення параметрів відбувається також завдяки заміні повнозв'язних шарів у верхній частині мережі шарами середнього пулінгу.

Нова архітектура отримала активний розвиток завдяки своїй ефективності. Inception-v3 – остання реалізація, що включає кілька десятків шарів.

VGGNet.

Відразу за GoogLeNet на ILSVRC 2014 розташувалася мережа К. Симоняна і Е. Циссермана VGGNet [132]. Розробникам вдалося наочно продемонструвати, що глибина є ключовим фактором для продуктивності. Їх мережа містить 16 згортальних і повнозв'язних шарів і має надзвичайно однорідну архітектуру, яка виконує згортання 3×3 і пулінг 2×2 від початку до кінця. Вихідна модель доступна в режимі PlugandPlay у фреймворку для глибинного навчання Caffe.

ResNet, Inception-ResNet

В остаточній мережі (Residual Network) [133], розробленої К. Хе і іншими і стала переможцем ILSVRC, інтенсивно використовувалися пакетна нормалізація і спеціальні скіп-з'єднання. Крім того, в кінці архітектури були відсутні повнозв'язні шари. ResNet станом на сьогоднішній день є справжнім витвором мистецтва в світі згортальних нейронних мереж і використовується найбільш часто.

Архітектура отримала подальший розвиток, наприклад, у вигляді високоефективної гібридної мережі Inception-ResNet.

R-CNN

Мета R-CNN (RegionsWithCNNs) [59,134-136] полягає в тому, щоб взяти зображення і правильно визначити, де знаходяться основні об'єкти (через прямокутник, що обмежує) на зображенні.

Архітектура мережі R-CNN була розроблена для застосування Convolution Neural Networks до задачі object detection.

CNN добре показували себе в класифікації зображень, і в даній мережі вони по суті були застосовані для того ж самого. Для цього на вхід CNN подавалося не все зображення повністю, а попередньо виділені іншим способом регіони, на яких імовірно є якісь об'єкти. На той момент таких підходів було кілька, автори обрали SelectiveSearch, хоча вони вказують, що особливих причин для переваги саме його немає.

Як CNN-мережі використовувалася готова архітектура – CaffeNet (AlexNet). Такі нейромережі, як і інші для набору зображень ImageNet, проводять класифікацію на 1000 класів. R-CNN розроблялася для детектування об'єктів меншої кількості класів ($N=20$ або 200), тому останній класифікаційний шар CaffeNet був замінений на шар з $N+1$ виходами (з додатковим класом для фону).

Selective Search видавав близько 2000 регіонів різного розміру і співвідношень сторін, однак CaffeNet приймає на вхід зображення фіксованого розміру 227×227 пікселів, тому перед подачею регіонів на вхід мережі їх доводилося модифікувати. Для цього зображення з регіону укладалося в найменший квадрат, що його охоплює. Уздовж тієї (меншої) сторони, по якій утворювалися поля, додавалося кілька «контекстних» (оточуючих регіон) пікселів зображення, решта поля нічим не заповнювалась. Отриманий квадрат масштабувався під розмір 227×227 і подавався на вхід CaffeNet.

Незважаючи на те, що CNN тренувалася на розпізнавання $N+1$ класів, в результаті вона використовувалася лише для вилучення фіксованого 4096-розмірного вектора ознак. Безпосереднім визначенням об'єкта на зображенні займалися N лінійних SVM, кожен з яких проводив бінарну класифікацію за своїм типом об'єктів, визначаючи чи є такий в переданому регіоні чи ні.

Слід зазначити, що отримані за допомогою SelectiveSearch регіони тільки можуть містити якісь об'єкти, і не факт, що містять їх повністю. Чи вважати регіон

таким, що містить об'єкт чи ні, визначалося по метриці IntersectionoverUnion (IoU), що представляє собою відношення площі перетину прямокутного регіону-кандидата з прямокутником, що насправді охоплює об'єкт, до площі об'єднання цих прямокутників. Якщо відношення перевершує задану граничну величину, регіон-кандидат вважається таким, що містить потрібний об'єкт.

Таким чином, процедуру детектування об'єктів мережею R-CNN можна розділити на наступні кроки:

1. Виділення регіонів-кандидатів за допомогою SelectiveSearch.
2. Перетворення регіону в розмір, який приймає CNN CaffeNet.
3. Отримання за допомогою CNN 4096-розмірного вектора ознак.
4. Проведення N бінарних класифікацій кожного вектора ознак за допомогою N лінійних SVM.
5. Лінійна регресія параметрів рамки регіону для більш точного охоплення об'єкта.

R-CNN працює дуже добре, але дуже повільно з декількох простих причин:

1. Потрібно прямий прохід CNN (AlexNet) для кожної пропозиції регіону для кожного окремого зображення (це близько 2000 прямих проходів на зображення!).
2. Необхідно навчання трьох різних моделей окремо - CNN для генерації об'єктів зображення, класифікатор, який передбачає клас, і модель регресії для посилення обмежувальних рамок. Це робить процес надзвичайно важким для навчання.

Автори відзначали, що розроблена ними архітектура також непогано показувала себе в завданні semantic segmentation.

Fast R-CNN

Незважаючи на високі результати, продуктивність R-CNN була все ж невисока, особливо для більш глибоких, ніж CaffeNet мереж (таких як VGG16). Крім того,

навчання boundingboxregressor і SVM вимагало збереження на диск великої кількості ознак, тому воно було дорогим з точки зору розміру сховища.

Автори Fast R-CNN [137] запропонували прискорити процес за рахунок модифікацій:

- пропускати через CNN не кожен з 2000 року регіонів-кандидатів окремо, а все зображення цілком. Запропоновані регіони потім накладаються на отриману загальну карту ознак;
- замість незалежного навчання трьох моделей (CNN, SVM, bboxregressor) поєднати всі процедури тренування в одну, тобто Fast R-CNN об'єднав CNN, класифікатор і обмежує регресорів в одну єдину мережу.

Перетворення ознак, які потрапили в різні регіони, до фіксованих розмірів проводилося за допомогою процедури RoIPooling (об'єднання областей інтересів). Вікно регіону шириною w і висотою h поділялося на сітку, що має $H \times W$ комірок розміром $h/H \times w/W$ (автори використовували $W=H=7$). По кожній такій комірці проводився MaxPooling для вибору тільки одного значення, задаючи таким чином результуючу матрицю ознак $H \times W$. За своєю суттю RoIPool розділяє пряму передачу CNN для зображення по всьому його субрегіону.

Бінарні SVM не використовувалися, замість цього вибрані ознаки передавалися на повнозв'язний шар, а потім на два паралельних шари: softmax з $K+1$ виходами (по одному на кожен клас $+1$ для фону) і boundingboxregressor.

Faster R-CNN

Після поліпшень, зроблених в Fast R-CNN, найвужчим місцем нейромережі виявився механізм генерації регіонів-кандидатів. У 2015 команда з Microsoft Research змогла зробити цей етап значно швидшим [139]. Вони запропонували обчислювати регіони не по первісному зображенню, а знову ж таки по карті ознак, отриманих з CNN. Для цього був доданий модуль під назвою Region Proposal Network (RPN).

В рамках RPN по витягнutih CNN ознаках ковзають «міні-нейромережу» з невеликим (3x3) вікном. Отримані з її допомогою значення передаються в два паралельних повнозв'язних шари: box-regressionlayer (reg) і box-classificationlayer (cls). Виходи цих шарів базуються на так званих anchor: k рамок для кожного положення ковзного вікна, що мають різні розміри і співвідношення сторін. Reg-шар для кожного такого anchor видає по 4 координати, що коректують положення рамки, що його охоплює; cls-шар видає по два числа - ймовірності того, що рамка містить хоч якийсь об'єкт або не містить.

Процес навчання reg і cls шарів об'єднаний; функцію втрат вони мають загальну, що представляє собою суму функцій втрат кожного з них, з балансуючим коефіцієнтом.

Обидва шари RPN видають тільки пропозиції для регіонів-кандидатів. Ті з них, які мають високу ймовірність вмісту будь-якого об'єкта, передаються далі в модуль детектування об'єктів і уточнення охоплюваної рамки, який як і раніше реалізований як Fast R-CNN.

Для того щоб розділяти ознаки, одержувані в CNN, між RPN і модулем детектування, процес навчання всієї мережі побудований ітераційно, з використанням декількох кроків:

1. Ініціалізується і навчається на визначення регіонів-кандидатів RPN-частина.
2. З використанням запропонованих RPN регіонів заново навчається Fast R-CNN частина.
3. Навчена мережа детектування використовується, щоб ініціалізувати ваги для RPN. Загальні convolution-шари, однак, фіксуються і проводиться доналаштування тільки шарів, специфічних для RPN.
4. З зафіксованими convolution-шарами остаточно доналаштовувати Fast R-CNN.

Mask R-CNN

Mask R-CNN [140,141] розвиває архітектуру Faster R-CNN шляхом додавання ще однієї гілки, яка передбачає положення маски, що покриває знайдений об'єкт, і, таким чином вирішує вже завдання instance segmentation. Маска являє собою просто прямокутну матрицю, в якій 1 на деякій позиції означає приналежність відповідного пікселя об'єкту заданого класу, 0 - що піксель об'єкту не належить.

Автори документа умовно поділяють розроблену архітектуру на CNN-мережу обчислення ознак зображення, звану ними backbone, і head – об'єднання частин, що відповідають за пророкування рамки, що охоплює, класифікацію об'єкта і визначення його маски. Функція втрат для них загальна і включає три компонента:

Виділення маски відбувається в class-agnostic стилі: маски передбачаються окремо для кожного класу, без попереднього знання, що зображено в регіоні, і потім просто вибирається маска класу, який переміг в незалежному класифікаторі. Стверджується, що такий підхід більш ефективний, ніж той, що спирається на апріорне знання класу.

Одна з основних модифікацій, що виникли через необхідність передбачати маску – заміна процедури RoIPool (обчислює матрицю ознак для регіону-кандидата) на так звану RoIAlign. Справа в тому, що карта ознак, отримана з CNN, має менший розмір, ніж вихідне зображення, і регіон, який охоплює на зображенні цілочисленну кількість пікселів, не виходить відобразити в пропорційний регіон карти з цілочисельною кількістю ознак.

У RoIPool проблема вирішувалася просто округленням дрібних значень до цілих. Такий підхід нормально працює при виділенні охоплюваної рамки, але обчислена на основі таких даних маска виходить занадто неточною.

На противагу цьому, в RoIAlign не використовується округлення, всі числа залишаються дійсними, а для обчислення значень ознак використовується білінійна інтерполяція по чотирьом найближчим цілочисельним точкам.

ДОДАТОК С

Акти впровадження дисертаційної роботи

«ЗАТВЕРДЖУЮ»

Генеральний директор ТОВ «Ипра-софт»



В.Б. Романчук

2018 р.

АКТ

про використання результатів науково-дослідної роботи
**«Методи швидкої обробки великих даних у розподіленому
 гетерогенному середовищі для сервіс-орієнтованих систем»**

Цим актом підтверджується, що розроблений співробітниками кафедри електронних обчислювальних машин Харківського національного університету радіоелектроніки старшим викладачем Росінським Дмитром Миколайовичем та здобувачем Новосельцевим Ігорем Валерійовичем під керівництвом кандидата технічних наук, доцента Аксак Наталії Георгіївни програмний продукт «CloudServicePlanner», в рамках виконання науково-дослідної роботи «Методи швидкої обробки великих даних у розподіленому гетерогенному середовищі для сервіс-орієнтованих систем», використовується для поліпшення взаємодії користувачів з постачальником інформаційних послуг ТОВ «Ипра-софт».

Програмний продукт «CloudServicePlanner» виконано у повній відповідності до технічного завдання побудови розподілених систем, які в реальному часі підтримують виконання сервіс-орієнтованих завдань, що дозволяє обробляти великі обсяги різномірної інформації, отриманої з різноманітних джерел різними пристроями, для надання сервісу якомога більшому числу користувачів. Розроблена експериментальна система добре узгоджується із запропонованою архітектурою Cloud-Fog-Dew для сервіс-орієнтованих систем.

Аксак Н.Г. розробила методологію організації адаптивного сервіс-орієнтованого розподіленого середовища, яка забезпечує ефективний доступ до сервіс-орієнтованих обчислювальних середовищ з використанням різноманітних датчиків та мобільних пристроїв та дозволяє створити мультиагентну систему оперативного реагування на критичне змінення стану досліджуваного об'єкту. Також було розроблено модель інформаційного супроводу процесу надання спеціалізованих послуг, яка передбачає наявність постійної підтримки експертів у даній проблемній області та консультативний супровід їхніх дій у діагностично складних випадках. Запропоновано організаційну структуру сервіс-орієнтованої системи, яка дозволяє сформувати агрегаційні рівні для динамічного розподілу ролей між агентами розробленої моделі з метою управління різними інцидентами.

Росінський Д.М. розробив елементи інфраструктури інформаційної системи, яка містить комплекс програмних та апаратних компонентів через інтеграцію систем з нейромережною обробкою та розподілених технологій, що

дозволяє організувати взаємодію віддалених компонентів системи в режимі реального часу.

Новосельцевим І.В. розроблено метод, що перетворює інформацію про користувача для діагностичного модуля, який дозволяє здійснити автоматизований аналіз введеної інформації для надання сервісу, оснований на співставленні існуючих ознак з отриманими зображеннями. Метод здатен ефективно працювати як із числовими даними, так і з наборами цифрових зображень, які мають ідентичний формат та були попередньо оброблені й відфільтровані.

Проведені експериментальні дослідження розробленого програмного продукту «CloudServicePlanner» показали його високу ефективність. Технічний ефект від впровадження програмного, інформаційного та методичного інструментаріїв для вирішення задачі надання спеціалізованих послуг у сервіс-орієнтованій системі полягає у зменшенні часу адаптації локального сайту на 7,2%, зменшенні часу обробки вхідних показників про користувача на 48,3%, зменшенні часу реакції системи на 3,5%, підвищенні ступеня релевантності інформації, що відображається, на 14,1%.

Даний акт не є підставою для фінансових розрахунків.

Технічний директор



О.Ф. Божок

УТВЕРЖДАЮ

Директор Центра репродукции человека Сана-мед

д.мед.н. А.М. Феськов

«01» июля 2009 г.



АКТ

внедрения в эксплуатацию программного продукта
«Адаптивный медицинский интернет-ресурс»

Данным актом подтверждается, что программный продукт «Адаптивный медицинский интернет-ресурс», разработанный сотрудниками кафедры ЭВМ Харьковского национального университета радиоэлектроники канд. техн. наук, доц. Аксак Натальей Георгиевной, аспирантами Коргутом Сергеем Анатольевичем и Новосельцевым Игорем Валерьевичем, выполнен в полном соответствии с техническим заданием по созданию адаптивного медицинского интернет-ресурса клиники и внедрен в Центре репродукции человека Сана-мед.

Программный продукт представляет медицинский интернет-ресурс на языке PHP с использованием базы данных MySQL и состоит из следующих подсистем:

1. Мультиагентная система (МАС).
2. Подсистема обработки данных пользователя.
3. База данных.
4. База знаний.
5. Система поддержки принятия решений.

В мультиагентной системе Аксак Н.Г. предложила методы взаимодействия агентов, Коргут С.А. разработал архитектуру МАС.

В подсистеме обработки данных пользователя Коргутом С.А. предложен метод обработки на основе информационного портрета пользователя.

Новосельцевым И.В. разработаны методы определения множества экспертов и способы извлечения экспертных знаний, на основе которых построены База данных и База знаний.

Разработанный программный комплекс позволяет предоставлять актуальную информацию по различным аспектам диагностики и лечения бесплодия, осуществлять медицинские консультации в режиме реального времени, осуществлять информационное сопровождение пациентов в процессе лечения и используется на этапах предварительной диагностики и лечения различных видов бесплодия.

Медицинский директор
Центра репродукции человека Сана-мед



д.мед.н. Феськова И.А.

ЗАТВЕРДЖУЮ
Проректор з наукової роботи Харківського
національного університету радіоелектроніки

« » 2007 р.
проф. Сіпченко М.І.

ЗАТВЕРДЖУЮ
Директор Інституту дерматології та
венерології АМН України

« » 2007 р.
проф. Мавров І.І.

АКТ

про впровадження результатів науково-дослідної роботи
«Експертна система диференційної діагностики алергічного контактного
і atopічного дерматитів, екземи та нейродерміту»

Цим актом підтверджується, що згідно з договором про наукове співробітництво між Інститутом дерматології та венерології АМН України (м. Харків) та Харківським національним університетом радіоелектроніки (ХНУРЕ) співробітниками кафедри електронних обчислювальних машин ХНУРЕ кандидатом технічних наук, доцентом Корабльовим Миколою Михайловичем, кандидатом технічних наук, доцентом Аксак Наталією Георгіївною та аспірантом Новосельцевим Ігорем Валерійовичем була розроблена експертна система у вигляді програмного комплексу, який орієнтований на допомогу дерматологам, лікарям загального профілю та студентам медичних вищих навчальних закладів.

Експертна система складається з таких підсистем:

1. Підсистема обробки зображень.
2. Підсистема взаємодії користувача з експертною системою.
3. Підсистема моделювання.
4. Підсистема прийняття рішень.

В підсистемі обробки зображень Корабльов М.М. запропонував і реалізував методи, моделі та алгоритми обробки зображень з використанням штучних імунних систем, Аксак Н.Г. запропонувала і реалізувала методи та архітектури високопродуктивних обчислювальних систем для паралельної обробки зображень.

В підсистемі взаємодії користувача з експертною системою Новосельцев І.В. розробив інтерфейс користувача.

В підсистемі моделювання Аксак Н.Г. запропонувала обчислювальні моделі нейроалгоритмів, Новосельцев І.В. запропонував способи, алгоритми та інструментальні засоби паралельної реалізації нейроалгоритмів.

В підсистемі прийняття рішень Корабльов М.М. запропонував методи та моделі надання рекомендацій лікарю, які реалізовані у вигляді правил нечіткого виводу з імунною настройкою.

В процесі тестування експертної системи в Інституті дерматології та венерології АМН України підтверджено її працездатність та доцільність використання для формування обґрунтованих рекомендацій лікарю-дерматологу щодо вибору процедури лікування алергодерматозів. Розроблені програмні засоби включено до складу експертної системи Інституту дерматології та венерології АМН України.

Комісія:

Завідувач лабораторії алергології
Інституту дерматології та венерології
АМН України, доктор медичних наук, професор

Е.М. Солошенко

Старший науковий співробітник відділення
дерматології Інституту дерматології та венерології
АМН України, кандидат медичних наук

О.М. Стулій

Завідувач кафедри ЕОМ Харківського
національного університету радіоелектроніки,
доктор технічних наук, професор

О.Г. Руденко

Доцент кафедри ЕОМ Харківського
національного університету радіоелектроніки,
кандидат технічних наук, доцент

М.О. Волк

ДОДАТОК Д

Список публікацій здобувача за темою дисертації

в яких опубліковані основні наукові результати дисертації:

1. Аксак Н. Г. Распознавание изменения размера и цвета изображения на основе сверточной нейронной сети / Н.Г. Аксак, А.А. Бессонов, И.В. Новосельцев, О.Г. Руденко // Бионика интеллекта: Научн.-техн. журнал. – 2018. – №2 (91). – С.114-119. (Входить до міжнародних наукометричних баз Google Scholar, Index Copernicus, BASE, Research Gate).
2. Новосельцев И. В. Система первичной диагностики меланомы кожи / И.В. Новосельцев, Н.Г. Аксак // Бионика интеллекта: Научн.-техн. журнал. – 2010. – №3(74). – С.94-98. (Входить до міжнародної наукометричної бази Google Scholar).
3. Новосельцев И.В. Применение многопроцессорных систем для реализации многослойного персептрона / И.В. Новосельцев, Н.Г. Аксак // Радиоэлектроника. Информатика. Управление. – №1(19). – 2008. – С.108-110. (Входить до міжнародних наукометричних баз РІНЦ, Google Scholar).
4. Новосельцев И.В.Вероятностные нейронные сети для решения задач медицинской диагностики / И.В. Новосельцев, Н.Г. Аксак, О.Ю.Барковская // Бионика интеллекта: Научн.-техн. журнал. – 2006. – №2(65). – С. 62-65.
5. Барковская О.Ю. Сравнительный анализ алгоритмов бинаризации полутонового изображения / О.Ю.Барковская, Н.Г. Аксак, И.В. Новосельцев // Автоматизированные системы управления и приборы автоматики, 2005. – Вып. 133. – С. 40-46.
6. Аксак Н. Г. Система предоставления медицинских услуг на основе архитектуры cloud-fog-dew / Н.Г. Аксак, И.В. Новосельцев, Д.Н. Росинский // Наука та виробництво: міжвуз. темат. зб. наук. пр. Вип. 19 / ДВНЗ «ПДТУ». –Маріуполь, ПДТУ, 2018. – С. 150-159.

7. Новосельцев І.В. Система віддаленого нейромережного діагностування меланоми / І.В. Новосельцев, Н.Г. Аксак // Збірник наукових праць, ХНУРЕ.– Харків: «Друкарня Мадрид», 2018. – 116-118.

8. Аксак Н.Г. Мультиагентная модель адаптации медицинского Интернет-ресурса на основе информационного портрета пользователя /Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Системи обробки інформації. Збірник наукових праць. – Х.: ХУПС, 2013. – Вип.2(109). – С. 242-246. (Входить до міжнародних наукометричних баз IndexCopernicus, BASE).

9. Новоосельцев И.В. Показатели эффективности нейросетевой системы медицинской диагностики / И.В. Новосельцев, Н.Г. Аксак // Системи управління, навігації та зв'язку. Збірник наукових праць. Випуск 4(16), Київ, 2010. - С.142-149.

10. Аксак Н.Г. Адаптация нейросетевых моделей на высокопроизводительные системы с общей памятью / Н.Г. Аксак, И.В. Новосельцев, А.С. Солдатов // Системні технології. Збірник наукових праць. – Випуск 3 (56).- Том 2.-Дніпропетровськ, 2008.- С. 117-122.

11. Новосельцев И.В. Гибридные нейронные сети для распознавания образов / И.В. Новосельцев, Н.Г. Аксак, О.Ю. Барковская // Системы обработки информации. Збірник наукових праць.-2007.- Вип. 3(61).-С.60-64. (Входить до міжнародних наукометричних баз BASE, WorldCat).

12. Барковская О.Ю. Обобщенная модель распознавания антропоморфных объектов / О.Ю. Барковская, И.В. Новосельцев, Н.Г. Аксак // Системні технології. Збірник наукових праць.- Випуск 6 (53).- Дніпропетровськ, 2007.- С. 11-16.

Публікації, які засвідчують апробацію матеріалів дисертації:

13. AxakN. Cloud-fog-dew Architecture for Personalized Service-oriented Systems / N. Axak, D.Rosinskiy, O. Barkovska, I. Novoseltsev // The 9thIEEEInternationalConferenceonDependableSystems, ServicesandTechnologies, DESSERT'2018, Kyiv, Ukraine – 2018. – P.80 -84. (Входить до міжнародної наукометричної бази Scopus, PІНЦ).

14. Аксак Н.Г. Спосіб контролю зміни розмірів об'єкта за допомогою шаблону / Н.Г. Аксак, М.В. Кушнар'ов, І.В. Новосельцев, А.Ю. Тихун // Патент на винахід № 84380 А по МПК (2006) G01C 11/00 від 10.10.2008. Бюл. №19.

15. Аксак Н.Г. Система интеллектуального анализа медицинских данных/ Н.Г. Аксак, И.В.Новосельцев, Д.Н. Росинский, В.О. Лебедев // Комп'ютерні та інформаційні системи і технології: Матеріали другої міжнародної науково-технічної конференції. Харків. – 2018. – С.128-129.

16. Аксак Н.Г. Агентно-ориентированный медицинский Интернет-ресурс / Н.Г. Аксак, С.А.Коргут, И.В.Новосельцев // Вычислительный интеллект (результаты, проблемы,перспективы): Материалы 2-й Международной научно-технической конференции. –Черкассы: Маклаут, 2013. – С.321.

17. Аксак Н.Г. Анализ способов ускорения нейроалгоритма на основе технологий параллельного программирования / Н.Г. Аксак, А.Ю. Тыхун, И.В. Новосельцев // Матеріали Х Міжнародної науково-технічної конференції "Системний аналіз та інформаційні технології". – К., 2008. – С.284.

18. Аксак Н.Г., Модели производительности синтеза нейроалгоритма на высокопроизводительных архитектурах / Н.Г. Аксак, А.Ю. Тихун, І.В. Новосельцев // Труды 8-й Международной конференции. Высокопроизводительные параллельные вычисления на кластерных системах (НРС-2008), изд-во КГТУ, Казань. – 2008. – С.199-202.

Публікації, які додатково відображають наукові результати дисертації:

19. Аксак Н.Г. Распознавание образов на основе нейронных сетей с использованием технологии МРІ / Н.Г. Аксак, Р.С. Верчиков, О.Р. Лола, И.В. Новосельцев, С.А. Олищук // Высокопроизводительные параллельные вычисления на кластерных системах. Материалы шестого Международного научно-практического семинара. Том 1. Санкт-Петербург, 2007. –С.21-25.

20. Новосельцев, И.В. Модель обработки больших данных для систем здравоохранения / И.В. Новосельцев, А.Н. Тарасенко //Материалы 22-го

международного молодежного форума «Радиоэлектроника и молодежь в 21 столетии». – 17-19 апреля 2018. – Харьков. – С. 57-58.

21. Аксак Н.Г. Нейросетевая медицинская система принятия решений / Н.Г. Аксак, И.В. Новосельцев, А.П. Лучко // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: Матеріали восьмої міжнародної науково-технічної конференції Полтава -Баку – Харків - Жиліна. – 2018.- С.13.

22. Новосельцев И.В. Реализация многослойного персептрона с помощью стандарта OpenMP / И.В. Новосельцев, Н.Г. Аксак // 8-я международная научно-практическая конференция "Современные информационные и электронные технологии": Сб. трудов конференции - Одесса, 2007.-С.79.

23. Новосельцев И.В. Применение нейронных сетей для построения медицинской экспертной системы / И.В. Новосельцев, Н.Г. Аксак // Международная научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов по материалам Междун. Научн. конференции.- Харьков: ХНУРЭ, 2003,- С.341-342.

24. Новосельцев И.В. Применение нейронной сети при фильтрации спам сообщений / И.В. Новосельцев, Н.Г. Аксак //10-я Юбилейная международная научная конференция "Теория и техника передачи, приема и обработки информации": Сб. тезисов докладов.Ч.2. - Харьков: ХНУРЭ, 2004.- С.295-296.

25. И.В.Новосельцев, Н.Г. Аксак. Использование параллельных технологий для распознавания образов. // Материалы 11-го юбилейного международного форума «Радиоэлектроника и молодежь в XXI веке». – Харьков: ХНУРЕ, 2007. – с.306.