

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Радіотехнічний факультет
Кафедра прикладної радіоелектроніки**

«На правах рукопису»
УДК _____

До захисту допущено:
В. о. зав. кафедрою
_____ Михайло Степанов
«__» _____ 20__ р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-професійною програмою «Радіозв'язок і оброблення сигналів»

за спеціальністю 172 «Телекомунікації та радіотехніка» на тему: «Стиснення зображень за допомогою нейронної мережі»

Виконала
студентка 2 курсу, групи РА-01мп
Мен Яцзін

Керівник:
доцент, к.т.н.
Лащевська Наталія Олександрівна

Рецензент:
старший викладач кафедри РТС, PhD

Миرونчук Олександр Юрійович

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студентка _____

Київ – 2021 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Радіотехнічний факультет
Кафедра прикладної радіоелектроніки

Рівень вищої освіти – другий (магістерський)

Спеціальність – 172 «Телекомунікації та радіотехніка»

Освітньо-професійна програма «Радіозв'язок і оброблення сигналів»

ЗАТВЕРДЖУЮ

В.о.зав. кафедрою

_____ Михайло СТЕПАНОВ

«___» _____ 20__ р.

ЗАВДАННЯ
на магістерську дисертацію студента

Мен Яцзін

1. Тема дисертації «Стиснення зображень за допомогою нейронної мережі», науковий керівник дисертації Лащевська Наталія Олександрівна, доцент, к.т.н., затверджені наказом по університету від «15» грудня 2021 р. №3744-с

2. Термін подання студентом дисертації 17 грудня 2021 року

3. Об'єкт дослідження є процес стиснення зображень в сучасних радіотехнічних системах та їх відновлення після стиснення.

4. Предмет дослідження є методи та алгоритми стиснення зображень та їх відновлення за допомогою нейронних мереж.

5. Перелік завдань, які потрібно розробити:

1) Аналіз та класифікація нейронних мереж;

2) Огляд методів використання нейронних мереж для стиснення зображень

6. Орієнтовний перелік графічного (ілюстративного) матеріалу презентація в Microsoft PowerPoint

7. Орієнтовний перелік публікацій _____

9. Дата видачі завдання 1 вересня 2021 року

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
	Отримання завдання та затвердження теми	01.09.2021	
	Аналіз та класифікація нейронних мереж	01.10.2021	
	Вибір НМ для стиснення зображень	20.10.2021	
	Вибір архітектури НМ	05.11.2021	
	Навчання НМ	15.11.2021	
	Написання алгоритму для тестування НМ	01.12.2021	
	Оформлення дисертації	10.12.2021	
	Представлення готової роботи на перевірку	15.12.2021	

Студент

Мен ЯЦЗІН

Науковий керівник



Наталія ЛАЩЕВСЬКА

АНОТАЦІЯ

Мережа ВР на даний момент є найпоширенішою моделлю штучної нейронної мережі. Вона використовує можливість перетворення режимів багаторівневої мережі прямого зв'язку для реалізації кодування даних і безпосередньо забезпечує можливості стиснення даних. На основі введення принципів та алгоритмів ВР мережевий компресор зображень через Комп'ютерний імітаційний експеримент Matlab реалізує стиснення цифрового зображення та аналізує вплив різних параметрів на продуктивність відновленого зображення.

ANNOTATION

The BP network is currently the most commonly used artificial neural network model. It uses the mode transformation capability of the multi-layer feedforward network to realize data encoding and directly provides data compression capabilities. On the basis of introducing the principles and algorithms of the BP network image compressor, through The computer Matlab simulation experiment realizes digital image compression, and analyzes the influence of various parameters on the performance of the reconstructed image.

ЗМІСТ

7

8

РОЗДІЛ 110

10

11

13

15

16

1.6 Стиснення мережі на основі відсікання..... 13

РОЗДІЛ 220

2.1 20

2.2 Принцип роботи нейронної мережі..... 122

2.3 Зміст дослідження штучної нейронної мережі 123

2.4 Напрями дослідження штучної нейронної мережі 124

2.5 25

2.6 31

2.7 Застосування нейронних мереж. 341

2.8 Моделювання стиснення нейронної мережі ВР 342

РОЗДІЛ 3..... 344

3.1 Створення архітектури НМ ВР..... 38

3.2 Навчання нейронної мережі ВР..... 40

3.3 Функція нейронної мережі ВР 45

Висновки 55

Перелік джерел посилань 56

1. ПЕРЕЛІК СКОРОЧЕНЬ

НМ — Нейронна мережа

ШНМ (ANN) – Штучна нейронна мережа

NNC – Neural network compression

ПФ – Передавальна функція

RBF – Мережа радіальних базисних функцій (radial basis function)

Нейронна мережа ВР – Back Propagation

2. ВСТУП

Актуальність теми дослідження. Протягом усього часу існування цифрових технологій задача компактної передачі інформації по каналу зв'язку була, є, та й скоріше за все, буде актуальною. Окремим прикладом такої інформації є візуальна інформація, до якої можна віднести статичні зображення.

З розвитком цифрової обробки інформаційних даних зросли і вимоги щодо розміру цих даних. Якщо мова йде про стиснення зображень, то вимоги пред'являються як швидкості стискання та якості їх передачі по каналам зв'язку, але всі існуючі на сьогодні алгоритми не у змозі задовольнити такі вимоги. Саме тому виникає потреба пошуку нових методів стиснення, які виконують більш інтелектуальний аналіз даних, а відповідно і є ефективнішими в тих чи інших характеристиках стиснення. За таких умов багато перспектив вбачають в нейромержевому підході до стиснення даних.

Суттєвою перевагою нейронних мереж (НМ) в задачі стиснення зображень є висока швидкість обробки, яка забезпечується за рахунок паралельної реалізації. Тому розробка програмного забезпечення для стиснення зображень на основі нейронної мережі є досить актуальною [1].

В роботі була поставлена задача провести дослідження штучних нейронних мереж як засобу для стиснення зображень, провести тестування, за результатами якого зробити висновки, щодо переваг використання нейромержевих методів для вирішення питань конволюції зображень.

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

Магістерська дисертація виконана відповідно до напрямку наукових досліджень кафедри прикладної радіоелектроніки КПІ ім. Ігоря Сікорського плану наукової та навчально-методичної роботи кафедри.

Мета та завдання дослідження. Метою дослідження магістерської дисертації є створення архітектури нейронної мережі та її навчання для підвищення швидкодії стиснення та якісного покращення результатів відновлення зображень після стиснення.

Для досягнення поставленої мети необхідно розв'язати наступні задачі:

1. Виконати огляд існуючих підходів до стиснення цифрових зображень;
2. Виконати аналіз запропонованих раніше методів розв'язання поставленої задачі;
3. Навести обґрунтування вибору методу для стиснення зображень;
4. Провести стиснення зображення шляхом моделювання нейронної мережі в середовищі MATLAB.

Об'єкт дослідження – це процес стиснення зображень комп'ютерними засобами з використанням технологій штучних нейронних мереж.

Предмет дослідження – це методи і програмні засоби стиснення зображень.

РОЗДІЛ 1

1.1 Стан розвитку та дослідницька значимість предмета

Цифрова обробка багатовимірних сигналів є класичною проблемою, яка включає в себе роздільну здатність і покращення зображення, а потім вхідне зображення перетворюється на вихідне зображення з потрібним розширенням, кольором або стилем.

Протягом усього часу існування цифрових технологій завдання компактної передачі інформації через канали зв'язку була, є і, ймовірно, буде актуальною. Конкретним прикладом такої інформації є візуальна інформація, в тому числі статичні зображення. З розвитком цифрової обробки даних зросли вимоги до розміру зображення, швидкості стиснення та якості передачі каналу зв'язку, але не всі існуючі алгоритми можуть задовольнити цю вимогу. Ось чому необхідно знайти нові методи стиснення, щоб виконувати більш розумний аналіз даних і бути ефективнішим у певних характеристиках стиснення. У цьому випадку перспективними виглядають нейромережеві методи стиснення даних.

Під час стиснення графічної інформації використовуються різні прийоми для зменшення кількості байтів, необхідних для представлення зображення. Звичайно, це значною мірою залежить від методу стиснення та вмісту файлу зображення (деякі файли стискаються набагато краще, ніж інші), але коли великий файл зображення стискається п'ять або більше разів, така ситуація дуже поширена. Деякі методи можуть стиснути більше, але знизять якість – інформація буде втрачена під час відновлення зображення. Тому розпаковане зображення може бути трохи розмитим і знебарвленим. Метод втрат забезпечує більш високий рівень стиснення, але не дозволяє скопіювати вихідне зображення до найближчого пікселя. Людське око не може сприймати всі тонкі кольорові відтінки звичайних растрових зображень.

Тому деякі деталі можна опустити без явного порушення інформаційного змісту зображення.

Основною характеристикою нейронної мережі є визначення зв'язку між різними параметрами інформації. Виявивши та відкидаючи зайву інформацію (стиснення без втрат), або відкидаючи неважливу частину інформації та її втрату, ми можемо представити її більш компактно. Не помічено (з втратами стиснення). Стиснення даних нейронної мережі — це проектування та навчання нейронних мереж для отримання меншої кількості шарів, ніж вхідних нейронів, щоб вхідний сигнал можна було відновити (стиск без втрат) або близький до нього (стиск із втратами). Результатом стиснення є зменшення розміру зображення, скорочення часу передачі зображення в мережі та економія місця для зберігання.

1.2 Способи навчання нейронних мереж

Навчання нейронної мережі в основному поділяється на навчання з наглядом (навчання з вчителем) і навчання без нагляду (навчання без учителя).

Навчання з вчителем. Кожен навчальний зразок у навчанні з вчителем відповідає сигналу вчителя, а сигнал вчителя представляє інформацію про навколишнє середовище. Мережа використовує сигнал вчителя як очікуваний вихід, обчислює помилку між фактичним і очікуваним результатом під час навчання, а потім оновлює ваги мережі відповідно до величини та напрямку помилки. Це налаштування повторюється до тих пір, поки помилка не досягне очікуваної точності, і вся мережа не утворить замкнену систему. Похибка може бути виміряна середньоквадратичним значенням помилки кожного вихідного вузла. Таким чином, встановлюється функція продуктивності з вагою мережі як незалежною змінною та кінцевою продуктивністю помилки як значенням функції. Навчання мережі трансформується в задачу

розв'язування точки мінімуму функції. Навчання під наглядом часто може ефективно виконувати такі функції, як класифікація шаблонів та підгонка функцій.

Навчання без нагляду. При навчанні без нагляду мережа отримує лише серію вхідних вибірок і нічого не знає про вихідне значення вибірки. Тому мережа може оновлювати вагові коефіцієнти лише на основі зв'язку між вхідними вибірками. Наприклад, у конкурентній нейронній мережі, що самоорганізується, подібні вхідні вибірки активують той самий вихідний нейрон для досягнення кластеризації зразків або асоціативної пам'яті. Оскільки навчання без нагляду не має очікуваного результату, його не можна використовувати для апроксимації функції [1].

Наведені вище два методи навчання відповідають наступним правилам навчання:

- правило навчання Хебба. Правило навчання Хебба — найстаріше правило навчання в нейронних мережах, вперше запропоноване нейропсихологом Хеббом. Ідея даного методу в наступному: якщо два нейрони, з'єднані на обох кінцях вагового коефіцієнту, активуються синхронно, енергія ваги буде вибірково збільшена; якщо два нейрони, з'єднані на обох кінцях вагового коефіцієнту, активуються асинхронно, енергія ваги буде вибірково зменшена. Математично коригування ваги пропорційне добутку вихідного значення попереднього нейрона на вихідне значення наступного нейрона.

Якщо припустити, що вихідне значення першого нейрона дорівнює a , вихідне значення останнього нейрона дорівнює b , а коефіцієнт навчання дорівнює η , тоді величина коригування ваги дорівнює

$$\Delta w = \eta ab \quad (1.1)$$

- правило навчання W-Н. Правило навчання W-Н також ще називають правилами навчання Delta або правилом навчання з виправленням помилок. Припускаючи, що очікуване вихідне значення дорівнює d , а фактичне вихідне значення дорівнює y , помилка дорівнює $e = d - y$. Метою навчання є мінімізувати помилку, тому коригування ваги є пропорційним похибці, тобто

$$\Delta w = \eta e y \quad (1.2)$$

- правило випадкового навчання. Правило випадкового навчання також називають правилами навчання Больцмана. Ідея походить від статистичної механіки. Нейронна мережа, розроблена на основі цього, називається машиною Больцмана. Машина Больцмана насправді є змодельованим алгоритмом відсікання.

- правило змагального навчання. Вихідні нейрони мережі конкурують між собою. У типовій конкурентній мережі лише один нейрон-переможець може регулювати вагу, а ваги інших нейронів залишаються незмінними, що відображає бічне гальмування між нейронами. Це узгоджується з механізмом роботи біологічних нейронів [2].

1.3 Компресуюча нейронна мережа (Neural network compression)

Модель глибокого навчання зазвичай містить мільйони або навіть десятки мільйонів параметрів і десятки шарів мережі, що зазвичай вимагає дуже великих обчислювальних витрат і місця для зберігання. Стиснення нейронної мережі означає зменшення параметрів мережі або простору для зберігання за допомогою зміни структури мережі або використання кількісних і наближених методів, а також зменшення трудоемності обчислення мережі та простору зберігання без впливу на продуктивність нейронної мережі.

Стиснення нейронної мережі можна грубо розділити на три типи методів: апроксимація, кількісна оцінка та адаптація. Наближені методи в основному використовують ідею розкладання матриці або тензора для відновлення вихідної матриці параметрів мережі або тензора параметрів за допомогою невеликої кількості параметрів для досягнення мети зменшення обчислювальних витрат на мережеве сховище. Зазвичай, коли мережа працює, ці параметри будуть належним чином відновлені, а обчислювальні витрати мережі на час виконання не були ефективно зменшені. Другим типом методу є метод

квантування. Основна ідея методу квантування полягає у відображенні можливих значень параметрів мережі з області реальних чисел у скінченний набір чисел або представлення параметрів мережі з меншою кількістю біт. Метод квантування обмежує початково нескінченні можливі параметри кількома параметрами, а потім повторно використовує ці параметри, щоб зменшити витрати на мережеве сховище. Змінюючи тип даних параметра, наприклад квантування вихідного 64-розрядного типу з плаваючою комою на ціле чи навіть логічний тип, обчислювальні витрати мережі на час виконання також будуть значно зменшені. Третій тип методу – це метод мережевого відсікання. Порівняно з першими двома методами головною особливістю мережевого відсікання є безпосередня зміна структури мережі. Відсікання мережі можна розділити на відсікання рівня, відсікання на рівні нейронів і відсікання на рівні нейронного з'єднання відповідно до деталізації. Об'єктом відсікання ієрархічного є мережевий рівень, а результатом відсікання є отримання більш дрібної мережі [3,4,5].

Стиснення нейронної мережі не тільки необхідно, але й можливе. Перш за все, чим нейронні мережі мають більш глибоку структуру, тим кращий ефект, але для конкретних сценаріїв застосування та вимог може бути задовільною мережа з відповідною глибиною та кількістю параметрів. Слабке підвищення продуктивності, викликане сліпим поглибленням складності мережі, не має великого значення в багатьох програмах. По-друге, нейронні мережі часто мають проблеми з параметризацією. Функції мережевих нейронів мають більшу повторюваність. Навіть у сценаріях, де продуктивність мережі є чутливою, більшість мереж можна «безпечно» стиснути, що не вплине на їх продуктивність. Стиснення нейронної мережі може допомогти зрозуміти механізм дії нейронів і дозволити розгорнути моделі нейронної мережі на простих пристроях.

Хоча глибокі нейронні мережі досягли хороших результатів у багатьох областях, таких як комп'ютерний зір, розпізнавання мови та робототехніка, практичне застосування глибокого навчання часто обмежене його обсягом

зберігання та обчислення. Наприклад, мережа VGG-16 містить приблизно 140 мільйонів параметрів чисел з плаваючою комою. Якщо припустити, що кожен параметр зберігається в 32-розрядному форматі чисел з плаваючою комою, вся мережа повинна займати більше 500 мегабайт пам'яті. Такі розрахунки можуть виконуватися лише за допомогою високопродуктивного паралельного обладнання, але вони не мають гарної продуктивності в реальному часі. Високопродуктивне паралельне обчислювальне обладнання має великі розміри, високе енергоспоживання та високу ціну, тому його не можна використовувати у багатьох випадках. Тому вирішення задачі, як запускати нейронні мережі в умовах обмежених ресурсів, таких як мобільні телефони, планшетні комп'ютери, різні вбудовані та портативні пристрої, є ключовим кроком для глибинного навчання, а також однією з найактуальніших проблем дослідження.

1.4 Стиснення мережі на основі тензорної декомпозиції

Тензор є природним розширенням вектора і матриці. Вектор можна назвати тензором першого порядку, а матрицю — тензором другого порядку. Складаючи матриці для формування «куба», ця структура даних називається тензор третього порядку. Зображення у відтінках сірого представлено в комп'ютері матрицею, яка є тензором другого порядку. Триканальне кольорове зображення RGB зберігається в комп'ютері як тензор третього порядку. Зазвичай, тензори третього порядку також можуть бути складені, щоб утворити тензори вищого порядку. Тензорна декомпозиція є важливою частиною тензорного аналізу. Основний принцип полягає у використанні структурної інформації в тензорних даних для розкладання тензорів на комбінацію кількох тензорів із простішими формами та меншими масштабами зберігання. Типові методи тензорного розкладання включають CP-декомпозицію, розклад Такера тощо. У нейронних мережах параметри зазвичай зберігаються

централізовано у вигляді «тензорів». Повністю зв'язаний шар перетворює вхідний вектор у вихідний через вагову матрицю, а його параметром є тензор другого порядку. Для згорткового шару припустимо, що вхідними даними є тензор третього порядку з каналом C , тоді кожне ядро згортки в шарі згортки також є ядром згортки третього порядку з каналом C тому набір ядер згортки, що міститься в шарі згортки, утворює форму, подібну до $V * M * N * C$ тензор четвертого порядку. Основна ідея стиснення мережі на основі тензорної декомпозиції полягає у використанні техніки тензорної декомпозиції для повторного вираження параметрів мережі у вигляді комбінації малих тензорів. Повторно виражена група тензорів зазвичай може бути приблизно такою ж, як і вихідний тензор з певною точністю, а зайнятий простір значно зменшується, щоб отримати ефект стиснення мережі [6].

1.5 Стиснення мережі на основі кількісної оцінки

Другий тип методу стиснення мережі базується на кількісних методах. Квантування тут в основному має два значення: перше - замінити високоточні параметри на параметри з низькою точністю, і перехопити параметри з точністю, суть - рівномірне квантування. По-друге, розділити ваги та обмежити типи ваг мережі, які можна використовувати. Обмежені типи ваг можуть бути додатково закодовані пізніше, і природа цього методу квантування полягає в нерівномірному квантуванні. Граничний випадок для зниження точності ваг. У цьому випадку вагові коефіцієнти згорткової мережі подвійні $+1$ і -1 . Швидкість обчислень мережі значно покращена, споживання пам'яті значно зменшено, а мережа має потенціал для використання апаратних логічних операцій для досягнення цієї можливості. Квантування розподілу значень — це процес відображення значення ваги мережі з усієї множини дійсних чисел у множину скінченних чисел .

1.6 Стиснення мережі на основі відсікання

Метод стиснення мережі, заснований на тензорній декомпозиції та квантуванні, фокусується на параметрах мережі. Під час процесу стиснення мережі топологія мережі залишається незмінною. Під час стиснення мережі на основі адаптації топологія мережі та методи виведення даних можуть змінюватися. Стиснення мережі на основі відсікання безпосередньо змінить структуру мережі, а його суть полягає у видаленні зайвої частини мережі.

Відповідно до різних об'єктів відсікання мережеве відсікання можна розділити на кілька частин, наприклад відсікання шарів, відсікання на рівні нейронів і відсікання на рівні нейронного з'єднання. Об'єктом відсікання ієрархічного відсікання є весь мережевий рівень, який в основному підходить для моделей з більшою кількістю мережевих шарів. Результатом відсікання є те, що нейронна мережа стає більш «мілкою». Видалення кількох модулів глибокої залишкової мережі насправді є свого роду ієрархічним відсіканням. Об'єктом відсікання на рівні нейронів є один нейрон або фільтр, і результатом відсікання є те, що нейронна мережа стає тоншою. Метою відсікання на рівні нейронного зв'язку є ваги з'єднання однієї нейронної мережі, а результат відсікання – нейронну мережу більш «розріджена». Після того, як нейрон відсікається, усі зв'язки, пов'язані з ним, теж відсікаються. Отже, відсікання нейронів насправді є окремим випадком відсікання нейронного зв'язку.

Завдяки своїй грубій зернистості відсікання рівня має великий вплив на вираження особливостей шару, і досліджень в цьому напрямі порівняно мало. Відсікання на рівні нейронного з'єднання є одним із найбільш досліджених методів стиснення мережі, і воно має більш тонку зернистість. Але його недоліком є те, що збереження розріджених нейронних зв'язків вимагає використання методів зберігання та обчислення розріджених тензорів. Зберігання розріджених тензорів вимагає додаткових обчислювальних витрат для збереження розташування точок даних, а фактично збережений простір для

зберігання менше, ніж кількість зменшених параметрів. Розрахунок розрізаних тензорів потребує спеціальних методів обчислення, що не сприяє паралельному обчисленню. Мова йде про те, що відсікання рівня нейронного зв'язку руйнує «регулярність» мережі [7,8,9].

РОЗДІЛ 2

2.1 Розвиток штучних нейронних мереж

У 1943 році психолог W·McCulloch і математичний логік W·Pitts вперше запропонували математичну модель нейронів, що базується на аналізі та узагальненні основних характеристик нейронів. Ця модель використовується і сьогодні, і безпосередньо впливає на хід досліджень у даній галузі. Тому їх двох можна назвати піонерами досліджень штучних нейронних мереж.

У 1945 році команда дизайнерів на чолі з Фон Нейманом успішно випробувала електронно-обчислювальні машини зі збереженою програмою, що ознаменувало початок ери ЕОМ.

У 1948 році у своїй дослідницькій роботі він порівняв принципову відмінність між структурою мозку людини і комп'ютером із збереженою програмою і запропонував мережеву структуру регенеративних автоматів, що складаються з простих нейронів.

Наприкінці 1950-х років F·Rosenblatt в своїй дослідницькій роботі запропонував поняття «персептрон», який є багатошаровою нейронною мережею. Ця робота вперше переносить дослідження штучної нейронної мережі з теоретичного обговорення в інженерну практику. У той час багато лабораторій світу імітували виробництво персептронів, які використовувалися для вивчення розпізнавання тексту, розпізнавання голосу, розпізнавання сигналів сонара, а також проблем з навчанням і пам'яттю.

Однак цей кульмінаційний період дослідження штучної нейронної мережі тривав недовго. Багато людей один за одним кидали дослідницькі роботи в цій галузі. Це пов'язано з тим, що розвиток цифрових комп'ютерів був у розквіті. Багато людей помилково вважали, що цифрові комп'ютери можуть вирішувати штучні інтелект і закономірності. Усі проблеми розпізнавання, експертної системи тощо змушують не звертати уваги на роботу персептрона.

На той час рівень електронної техніки був відносно відсталим, основними компонентами були електронні лампи або транзистори, нейронна мережа, створена з їх допомогою, була громіздкою і дорогою, зробити її схожою за масштабом на реальну нейронну мережу було абсолютно неможливо.

У книзі під назвою «Персептрон» 1968 року вказувалося, що лінійний персептрон має обмежені функції. Він не може вирішити основні проблеми, такі як відчуження. Більше того, багатосарова мережа не змогла знайти ефективні методи обчислення. Ці аргументи спонукали велику кількість дослідників втратити впевненість у перспективах створення штучних нейронних мереж. Наприкінці 1960-х років дослідження штучних нейронних мереж пішли на спад.

На початку 1960-х років Уїдроу запропонував мережу з адаптивними лінійними елементами, яка є лінійною зваженою сумою порогової мережі з безперервними значеннями. Пізніше на цій основі була розроблена нелінійна багатосарова адаптивна мережа. У той час, хоча ці роботи не позначали назву нейронної мережі, насправді це була модель штучної нейронної мережі [10,11].

Зі зниженням інтересу людей до персептронів вивчення нейронних мереж на довгий час було призупинене. На початку 1980-х років розробка цифрових комп'ютерів зіткнулася з труднощами в ряді областей застосування.

Американський фізик Хопфілд опублікував дві роботи про дослідження штучних нейронних мереж у *Proceedings of the National Academy of Sciences* у 1982 та 1984 роках, які викликали величезний резонанс. Люди знову визнали силу нейронних мереж і реальність їх застосування. Відразу велика кількість вчених і дослідників провели подальшу роботу навколо методу, запропонованого Хопфілдом, сформувавши дослідницький бум у штучних нейронних мережах.

2.2 Принцип роботи нейронної мережі

Штучні нейронні мережі повинні спочатку навчатися за певними критеріями навчання, перш ніж вони зможуть працювати. Тепер візьмемо для прикладу розпізнавання штучною нейронною мережею двох літер «А» і «В». Обумовлено, що коли «А» вводиться в мережу, вона має виводити «1», а коли вхід — «В», вихід "0".

Отже, критерій для мережевого навчання повинен бути таким: якщо мережа приймає неправильне рішення, за допомогою мережевого навчання мережа повинна зменшити можливість зробити таку ж помилку наступного разу. Спочатку призначте випадкове значення в інтервалі $(0,1)$ кожному з'єднанню мережі та введіть у мережу режим зображення, відповідний "А". Мережа додає вагу вхідного режиму, порівнює його з порогове значення, а потім виконує нелінійну операцію, отримує вихід мережі. У цьому випадку ймовірність того, що вихід мережі буде «1» і «0» становить 50%, що означає, що він є абсолютно випадковим. У цей час, якщо вихід дорівнює «1» (результат правильний), вага з'єднання збільшується, щоб, коли мережа знову зустрічає вхід режиму «А», вона все ще може зробити правильне рішення.

Функція звичайного комп'ютера залежить від знань і вмінь, наведених у програмі. Очевидно, що узагальнити та зібрати програми для розумної діяльності буде дуже важко.

Штучні нейронні мережі також мають можливості попередньої самоадаптації та самоорганізації. Змініть значення синаптичної ваги в процесі навчання або тренувань, щоб адаптуватися до вимог навколишнього середовища. Одна і та ж мережа може виконувати різні функції через різні методи навчання та зміст. Штучна нейронна мережа – це система зі здатністю до навчання, яка може розвивати знання так, щоб перевищувати початковий рівень знань дизайнера [7].

2.3 Зміст дослідження штучної нейронної мережі

Зміст досліджень нейронних мереж є досить широким, що відображає особливості мультидисциплінарних та міждисциплінарних технічних галузей. Нині основна дослідницька робота зосереджена на наступних аспектах:

1) дослідження біологічного прототипу. З біологічних наук фізіології, психології, анатомії, науки про мозок, патології та ін., вивчають структуру біологічного прототипу та функціональний механізм нервових клітин, нейронних мереж і нервової системи;

2) створення теоретичної моделі. На основі дослідження біологічних прототипів встановлюються теоретичні моделі нейронів і нейронних мереж, у тому числі концептуальні моделі, моделі знань, фізико-хімічні моделі, математичні моделі тощо;

3) дослідження мережевої моделі та алгоритму. Створюються конкретні моделі нейронних мереж на основі теоретичних досліджень моделі для реалізації комп'ютерного моделювання або підготовки обладнання, включаючи дослідження алгоритмів мережевого навчання. Цю роботу також називають дослідженням технічної моделі. Алгоритмом, який використовується в нейронних мережах, є векторне множення, широко використовуються символічні функції та їх різноманітні наближення. Паралелізм, стійкість до відмов, апаратна реалізація та характеристики самонавчання є основними перевагами нейронних мереж і відмінністю методів обчислення нейронних мереж від традиційних методів.

4) прикладна система штучної нейронної мережі, заснована на дослідженні мережевої моделі та алгоритму, використовує штучну нейронну мережу для формування фактичної прикладної системи, наприклад, для завершення певної обробки сигналів або функції розпізнавання образів, конструювання експертної системи, створення робота тощо [12].

2.4 Напрями дослідження штучної нейронної мережі

Дослідження нейронної мережі можна розділити на два аспекти: теоретичні дослідження та прикладні дослідження. Теоретичні дослідження можна розділити на дві категорії:

- з використанням нейрофізіологічних та когнітивних наукових досліджень щодо людського мислення та механізмів інтелекту;
- з використанням результатів досліджень базової нейронної теорії, використання математичних методів для дослідження моделей нейронних мереж з більш повними функціями та високою продуктивністю, а також поглиблене вивчення мережевих алгоритмів і продуктивності, таких як стабільність, конвергенція, відмовостійкість, надійність, тощо; розробка нових мережевих математичних теорій, таких як динаміка нейронних мереж, нелінійні нейронні поля тощо [13,14].

Прикладні дослідження також можна розділити на дві наступні категорії:

- дослідження з програмного моделювання та апаратної реалізації нейронної мережі;
- дослідження щодо застосування нейронних мереж у різних галузях: розпізнавання образів, обробка сигналів, розробка знань, експертні системи, оптимізовані комбінації, керування роботами тощо.

З безперервним розвитком самої теорії нейронних мереж, пов'язаних теорій і пов'язаних технологій, застосування нейронних мереж, безсумнівно, стає більш поглибленим.

2.5 Тенденція розвитку штучної нейронної мережі

Унікальні можливості нелінійної адаптивної обробки інформації штучних нейронних дають можливість успішно застосовувати їх для оптимізації комбінацій, прогнозування та для вирішення багатьох інших задач.

В останні роки штучні нейронні мережі все глибше розвиваються на шляху моделювання людського пізнання, поєднуючись з нечіткими системами, генетичними алгоритмами, еволюційними механізмами тощо, для формування обчислювального інтелекту, який став важливим напрямком штучного інтелекту та волі.

Застосування інформаційної геометрії до дослідження штучних нейронних мереж відкрило новий шлях для теоретичного дослідження штучних нейронних мереж. Нейронні мережі добре застосовуються в багатьох областях, але є ще багато аспектів, які потребують вивчення.

Оскільки інші методи також мають свої переваги, нейронну мережу поєднують з іншими методами, щоб вчитися на сильних сторонах один одного, а потім отримати кращі ефекти застосування. Поточна робота в цій галузі включає злиття нейронних мереж і нечіткої логіки, експертних систем, генетичних алгоритмів, вейвлет-аналізу, хаосу, грубої теорії множин, теорії фракталів, теорії доказів і сірих систем. Нижче наведено в основному аналіз злиття нейронної мережі та вейвлет-аналізу, хаосу, грубої теорії множин і теорії фракталів.

2.5.1 Поєднання з вейвлет-аналізом

Вейвлет-перетворення є проривом у методі аналізу Фур'є. Він не тільки має хороші властивості локалізації як у часовій, так і в частотній областях, але також має хорошу роздільну здатність для низькочастотних сигналів у частотній області та для високочастотних сигналів у часовій області, так що

він може зібрати будь-які деталі об'єкт. Вейвлет-аналіз еквівалентний математичному мікроскопу з функціями збільшення, зменшення та панорамування. Він може вивчати динамічні характеристики сигналу, перевіряючи зміни при різних збільшеннях. Тому вейвлет-аналіз став потужним інструментом у багатьох областях, таких як геофізика, обробка сигналів, обробка зображень та теоретична фізика.

Вейвлет-нейронна мережа поєднує в собі хороші характеристики частотно-часової локалізації вейвлет-перетворення з функцією самонавчання нейронної мережі, тому вона має сильну здатність до апроксимації та відмовостійкості. У комбінаційному методі вейвлет-функцію можна використовувати як передавальну функцію для побудови нейронної мережі для формування вейвлет-мережі, або вейвлет-перетворення можна використовувати як інструмент попередньої обробки вхідних даних нейронної мережі з прямим зв'язком, тобто Сигнал стану процесу обробляється характеристиками багаторазової роздільної здатності вейвлет-перетворення Реалізують поділ сигнал-шум і виділяють характеристики стану, які мають найбільший вплив на помилки обробки, як вхід нейронної мережі.

Нейронна мережа Wavelet використовується в діагностиці несправностей двигуна, обробці сигналів несправності високовольтної електромережі та дослідженні захисту, діагностиці підшипників та інших механічних несправностей та багатьох інших програмах. система має хорошу продуктивність контролю відстеження та хорошу надійність, використання нейронної мережі вейвлет-пакетів для інтелектуальної діагностики серцево-судинних захворювань, вейвлет-шару для адаптивного виділення ознак у часовій і частотній області, пряму нейронну мережу для класифікації, правильна швидкість класифікації досягає 94% .

Хоча вейвлет-нейронна мережа використовується в багатьох аспектах, вона все ж має деякі недоліки. Виходячи з вимог точності вилучення та вейвлет-перетворення в реальному часі, необхідно побудувати деякі спеціальні

вейвлет-бази, адаптовані до вимог програми відповідно до фактичної ситуації, щоб досягти кращих результатів у застосуванні. Крім того, вимоги реального часу в додатку також повинні поєднувати розробку DSP для розробки спеціальних чіпів обробки для задоволення цієї вимоги.

2.5.2 Хаотична нейронна мережа

Хаос – це нерегулярний рух, який виникає в певній системі. Хаос – відносно поширене явище, яке існує в нелінійних системах. Хаотичний рух має характеристики ергодичності та випадковості, які можна регулювати в певному діапазоні. Він перетинає всі стани, не повторюючи власного закону. Теорія хаосу визначає хаос нелінійної динаміки. Мета полягає в тому, щоб розкрити прості закони, які можуть ховатися за, здавалося б, випадковими явищами, щоб виявити загальні закони, яким зазвичай слідує великий клас складних проблем.

У 1990 році К. Айхара, Т. Такабе, М. Тойода та інші вперше запропонували хаотичну модель нейронної мережі, засновану на хаотичних характеристиках біологічних нейронів, і внесли хаос в нейронну мережу, зробивши штучну нейронну мережу хаотичною, що ближче до реальної людської мозкової нейронної мережі, тому хаотична нейронна мережа розглядається як одна з інтелектуальних систем обробки інформації, яка може реалізувати її реальні обчислення, і стала одним з основних напрямків дослідження нейронної мережі.

Порівняно зі звичайною дискретною нейронною мережею Хопфілда хаотична нейронна мережа має багатші нелінійні динамічні характеристики.

У реальному застосуванні нейронної мережі, коли мережевий вхід сильно змінюється, властива відмовостійкість мережі прикладних програм часто є недостатньою, і часто виникає амнезія. Динамічна пам'ять хаотичної нейронної мережі належить до детермінованого динамічного руху. Пам'ять

виникає на траєкторії хаотичного атрактора. Завдяки безперервному переміщенню (процесу відкликання) шаблон пам'яті пов'язується один за одним, особливо для тих просторів станів, які тісно розподілені або виникають. Накладаються шаблони пам'яті, хаотичні нейронні мережі завжди можна відтворити та ідентифікувати за допомогою динамічної асоціативної пам'яті без плутанини. Це унікальна продуктивність хаотичних нейронних мереж, яка значно покращить здатність нейронних мереж Хопфілда до пам'яті. Існує домен тяжіння хаотичного атрактора, який утворює властиву відмовостійкої функції хаотичної нейронної мережі. Це відіграє важливу роль у складному розпізнаванні образів, обробці зображень та інших інженерних додатках [15].

Інша причина уваги хаотичних нейронних мереж полягає в тому, що хаос існує в реальних нейронах і нейронних мережах організмів і відіграє певну роль, що підтвердили електрофізіологічні експерименти зоології.

Завдяки своїм складним динамічним характеристикам хаотичним нейронним мережам приділено велику увагу в області динамічної асоціативної пам'яті, оптимізації систем, обробки інформації та штучного інтелекту.

Досліджено застосування хаотичної нейронної мережі в задачі комбінаторної оптимізації. Для того, щоб краще застосувати динамічні характеристики хаотичної нейронної мережі та ефективно керувати наявним хаотичним явищем, все ще необхідно вдосконалювати та коригувати структуру хаотичної нейронної мережі, а також додатково вивчати алгоритм роботи хаотичної нейронної мережі.

2.5.3 На основі грубої теорії множин

Груба теорія (Rough sets) множин була вперше запропонована З. Павлаком, професором Варшавського технологічного університету в 1982 році. Це математична теорія для аналізу даних, вивчення неповних даних, неточ-

ного вираження знань, навчання, індукції та інших методів. Груба теорія множин є новим математичним інструментом для роботи з нечіткими та невизначеними знаннями, головна ідея якого полягає в тому, щоб вивести правила прийняття рішень або класифікації задачі шляхом скорочення знань за умови збереження незмінною класифікаційної здатності. В даний час груба теорія множин успішно застосовується в областях машинного навчання, аналізу рішень, управління процесами, розпізнавання образів та інтелекту. Комбінація грубого набору та нейронної мережі використовується в розпізнаванні мовлення, експертних системах, інтелектуальних аналізах даних, діагностиці несправностей та інших галузях.

2.5.4 Поєднання з фрактальною теорією

Вчений Бенуа Б. Мандельброт, професор кафедри математики Гарвардського університету, ввів поняття фракталу в середині 1970-х років, і фрактальна геометрія перетворилася на наукову методологію-фрактальну теорію. Завдяки швидкому розвитку багатьох дисциплін фрактал став дисципліною, яка описує закономірність багатьох нерегулярних речей у природі та широко використовується в різних областях, таких як біологія, географія, астрономія та комп'ютерна графіка.

Використання фрактальної теорії для пояснення нерегулярних, нестабільних і дуже складних структурних явищ у природі може дати значні результати, а нейронна мережа і фрактальна теорія об'єднані для повного використання нелінійного відображення, обчислювальної потужності та самостійності нейронної мережі. та інші переваги дозволяють досягти кращих результатів. Сфери застосування фрактальних нейронних мереж включають розпізнавання зображень, кодування зображень, стиснення зображень та діагностику несправностей систем механічного обладнання.

Метод стиснення/декомпресії фрактального зображення має переваги високого ступеня стиснення та низького коефіцієнта втрат, але його обчислювальна потужність невелика. Оскільки нейронна мережа має характеристики паралельної роботи, нейронна мережа використовується при стисканні/декомпресії фрактального зображення, що покращує вихідну обчислювальну потужність методу.

2.5.5 Штучні нейронні мережі

Штучна нейронна мережа (ANN), також відома як нейронна мережа, представляє собою математичну модель, в якій для обробки інформації використовується структура, аналогічна синаптичним зв'язкам нервів мозку. .

Нейронна мережа - це обчислювальна модель, яка складається з великої кількості вузлів (або нейронів, або одиниць) і зважених зв'язків між ними. Кожен вузол представляє певну функцію виведення, яка називається функцією активації(activation function) [1].

Кожне з'єднання між двома вузлами є зважене значення для сигналу, що проходить через з'єднання, яке називається вагою(weight), що еквівалентно пам'яті нейронної мережі. Вихідні дані мережі розрізняються залежно від режиму підключення, значення ваг і функції збудження мережі. Сама мережа зазвичай є наближенням певного алгоритму або функції за своєю природою або може бути вираженням логічної стратегії.

Нейронні мережі часто називають нейронними комп'ютерами, але вони відрізняються від сучасних цифрових комп'ютерів, головним чином, в таких аспектах:

1) Нелінійність. Штучні нейрони знаходяться в двох різних станах: активації або гальмування, що математично виражається у вигляді нелінійної залежності. Мережа, що складається з нейронів з граничними значеннями, має кращу продуктивність і може поліпшити відмовостійкість і ємність сховища.

2) Без обмежень. Нейронна мережа зазвичай складається з декількох нейронів, широко пов'язаних між собою. Загальна поведінка системи не тільки залежить від характеристик окремого нейрона, але також може в основному визначатися взаємодією і взаємозв'язком між елементами. Завдяки великій кількості з'єднань між одиницями, імітується необмежену функціонування мозку. Асоціативна пам'ять - типовий приклад необмеженої НМ.

3) Якість. Нейронна мережа має здатність до самоадаптації, самоорганізації і самонавчання. Мало того, що інформація, яка обробляється нейронною мережею, може мати різні зміни, але при обробці інформації постійно змінюється і сама нелінійна динамічна система. Ітераційний процес зазвичай використовується для опису еволюції динамічної системи.

4) Невипуклість. Напрямок еволюції системи буде залежати від конкретної функції стану при певних умовах. Наприклад, енергетична функція, її крайнє значення відповідає щодо стабільного стану системи. Невипуклість означає, що така функція має кілька екстремальних значень, тому система має кілька стійких станів рівноваги, що призведе до різноманітності еволюції системи.

2.6 Модель нейронної мережі та навчання

Штучна нейронна мережа - це обчислювальна модель, яка складається з великої кількості нейронів та їх взаємозв'язків. Кожен вузол представляє певну вихідну функцію, яку називають функцією активації. Кожне з'єднання між двома нейронами являє собою зважене значення для сигналу, що проходить через з'єднання, називається вагою, що еквівалентно пам'яті штучної нейронної мережі [2,3].

Вихід мережі різний залежно від режиму підключення, значення ваги та функції збудження мережі. Сама мережа, як правило, є наближенням певного алгоритму чи функції за своєю природою, або може бути виразом логічної стратегії.

2.6.1 Нейронні клітини

Нейрони є основними одиницями будови і функції нервової системи. Має виступи, які можуть поширюватися на різні органи та тканини по всьому тілу. Випинання поділяються на дендрити та аксони. Нейрон має аксони, які можуть передавати збудження від тіла клітини до іншого нейрона чи інших тканин, але є багато дендритів, які отримують стимуляцію та передають збудження в тіло клітини. Нейронна клітина складається з тіла клітини, синапсу, аксона та дендритів, а структура показана на малюнку рисунку 2.1 [1].

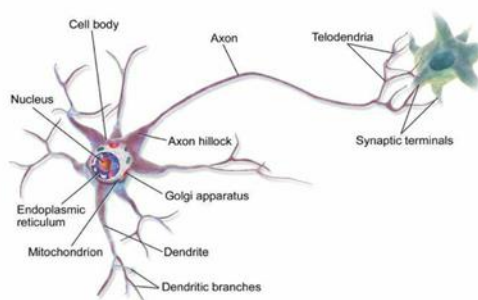


Рисунок 2.1. – Структура біологічного нейрона

Тіло клітини: у сірій речовині та гангліях головного та спинного мозку її морфологія різна, звичайні морфології зірки, піраміди, груші та сферичної форми.

Дендрити: один або кілька виступів з тіла клітини радіальної форми.

Аксон: найдовше трубчасте волокно, яке виступає з тіла клітини.

Синапс: точка зв'язку між нейронами і нейронами. Це ключова структура для передачі інформації між нейронами. Синапси можна розділити на дві категорії: хімічні синапси та електричні синапси.

Нейрони передають інформацію через синапси. Коли нервовий імпульс передається до пресинаптичної мембрани, синаптичні везикули перемі-

щуються до пресинаптичної мембрани, а нейромедіатори у везикулах вивільняються шляхом екзоцитозу Деякі нейромедіатори та відповідні рецептори на постсинаптичній мембрані Комбіновано розподіл іонів на обидві сторони задньої мембрани змінюються, демонструючи збудливі або гальмівні зміни, таким чином впливаючи на активність постсинаптичних нейронів (або ефекторних клітин).

Синапси, що збуджують постсинаптичну мембрану, називаються збудливими, а ті, що викликають гальмування задньої мембрани, — гальмівними.

Збудження або гальмування синапсів залежить від типів нейромедіаторів та їх рецепторів. Оскільки нейрон зазвичай має багато синапсів, деякі з них є збудливими, а деякі гальмівними. Якщо сумарна інтенсивність збудливої синаптичної активності перевищує загальну інтенсивність гальмівної синаптичної активності і досягає певного порогу, то в початковому сегменті аксона нейрона може генеруватися потенціал дії для генерації нервового імпульсу. При появі нервового імпульсу нейрон збуджується, в іншому випадку — гальмується.

Деякі нейрони мають один аксон і один дендрит, звані біполярними нейронами; деякі нейрони мають один аксон і декілька дендритів, званих мультиполярними нейронами; деякі нейрони мають тільки одне волокно, звані уніполярними нейронами. Наприклад, сенсорні нейрони спинномозкових нервів людини та інших хребетних розтягують тільки одне волокно, але діляться на дві гілки недалеко від тіла клітини. Одна гілка йде до рецептора і називається периферичною гілкою, а інша гілка йде в спинний мозок і називається центральною гілкою. Перше представляє собою аферентне волокно, яке передає сенсорний імпульс від рецептора в тіло клітини, тому воно повинно бути дендритним з функціональної точки зору, але має структуру аксона; Останнє має бути аксоном з функціональною і структурною точкою зору.

Нейрони мають такими основними функціональними характеристиками:

- функція інтегрування часу і простору;
- динамічна поляризація нейронів;
- порушений і загальмований стан;
- пластичність конструкції;
- перетворення імпульсного і потенційного сигналу;
- синаптичне розширення і не відкладання;
- навчання, забуття і втома.

Нейронна мережа - це мережева система, що складається з великої кількості нейронних одиниць, пов'язаних один з одним. Однак насправді модель нейронної мережі є лише абстракцією, спрощенням і симуляцією біологічної нейронної мережі і не може повністю відображати функції мозку. Ідея моделі нейронної мережі натхненна дослідженнями біологічних моделей нейронної мережі.

2.6.2 Модель нейронної мережі

Подібно біологічній нервовій системі, штучна нейронна мережа також складається з штучних нейронів в якості основної одиниці. Штучний нейрон - це математична модель, що імітує біологічний нейрон. Це основний блок обробки штучної нейронної мережі. Це також нелінійний компонент з декількома входами і одним виходом, як показано на рисунку 2.2. [1, 10].

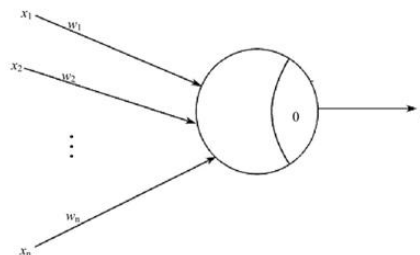


Рисунок 2.2 - Проста модель штучного нейрона

Кожне вхідне з'єднання нейрона має силу синаптичного з'єднання, яка представлена вагою з'єднання. Генерований сигнал посилюється силою з'єднання, і кожен вхід (X_j) має пов'язаний вага (W_{xj}). Блок обробки Квант зважений введення, а потім додає зважене значення для обчислення унікального виведення. Цей висновок (y) є функцією зваженої суми. Ця функція зазвичай називається функцією передачі. Цей процес можна виразити як:

$$y = f\left(\sum_j w_{ij}x_j + b\right) \quad (2.1)$$

де f являє передавальну функцію, яка використовується цим нейроном.

Для базового нейрона персептрона принцип роботи полягає в порівнянні зваженої суми з порогом нейрона. Якщо вона більше порога, нейрон активується. Коли він активований, сигнал передається на нейрони вищого рівня, пов'язані з ним. Передавальна функція, яка використовується в цьому процесі, є функцією жорсткого обмеження.

У різних типах нейронних мереж використовуються різні передавальні функції. Частіше використовуються наступні функції: функція жорсткого обмеження (*hardlim*), лінійна функція (*purelin*), сигмоїдальна функція (*logsig*), радіальна базисна функція Гаусса і т. Д Різниця в передавальній функції також призводить до відмінностей в структурі та функціях різних нейронних мереж, як показано на рисунку 2.3.

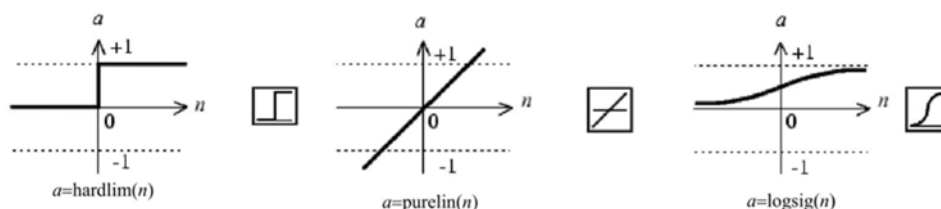


Рисунок 2.3 - Принципова діаграма поширених передавальних функцій

Нейронна мережа складається з одиниць обробки, розташованих пошарово. Нейронний шар, який приймає вхідний сигнал, називається вхідним шаром, нейронний шар, який виводить сигнал, називається вихідним шаром, а

шар нейронів, який безпосередньо не пов'язаний із входом/виходом називається середнім шаром або прихованим шаром.

Якщо мережа отримує набір вхідних даних, кожен нейрон у вхідному шарі мережі отримує частину вхідного шаблону, а потім вхідний шар передає вхідний сигнал до прихованого шару через з'єднання. Прихований шар отримує весь шаблон введення. Завдяки функції передачі вихід блоку прихованого шару сильно відрізняється від вхідного шару.

Вихідний блок приймає всі режими активності на виході з блоку прихованого шару, але передача сигналу з блоку прихованого шару на вихідний шар з'єднана вагами. Тому блок вихідного шару збуджується або гальмується для створення відповідного вихідного сигналу. Вихідний режим блоку вихідного шару - це загальна реакція мережі на збудження режиму входу.

Основна структура нейронної мережі показана на рисунку 2.4.

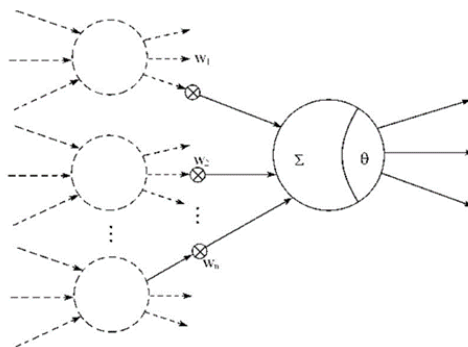


Рисунок 2.4 - Принципова діаграма структури нейронної мережі

Для більшості нейронних мереж, як тільки функція передачі буде обрана під час роботи мережі, вона залишається незмінною. Динамічна модифікація ваг є самим базовим процесом у навчанні, подібним до «інтелектуального процесу». Можна зробити висновок, що найважливіша інформація мережі зберігається у скоригованих вагах.

2.6.3 Навчання нейронної мережі

Після визначення модельної структури нейронної мережі, наступним кроком є навчання та навчання. Мережа не завершує процес навчання, змінюючи сам процесор, а навчається, змінюючи вагу з'єднань у мережі. Тому, якщо блок обробки навчиться правильно відображати шаблон даних, єдиний елемент, який використовується для зміни продуктивності процесора, - це ваги з'єднання [7].

Через різницю в моделі підключення нейронної мережі, дискретність або безперервність вхідної інформації, наявність чи відсутність тренування під наглядом та динамічні характеристики нейрона відповідний алгоритм навчання також відрізняється.

Керований алгоритм навчання. Потрібно подавати як вхідний, так і правильний вихідний сигнал. Мережа налаштовує мережу відповідно до різниці між поточним виходом та необхідним цільовим виходом, щоб мережа відповідала правильно.

Алгоритм навчання без нагляду. Враховуючи лише набір вхідних даних, мережа може поступово еволюціонувати, щоб відповідати певним моделям вхідних даних. Тобто у навчальних зразках є лише вхідні вектори, а нейронна мережа коригує ваги, порівнюючи співвідношення між вхідними векторами, тим самим відображаючи співвідношення між вхідними векторами у навчальних вибірках до ваг мережі для досягнення асоціативних функція стиснення пам'яті або даних. Наприклад, людський мозок використовує типове тренування без нагляду. Оскільки цей тип мережі не має вихідного вектора, його не можна використовувати для моделювання функцій.

Правила навчання нейронних мереж відповідають різним стратегіям регулювання ваг нейронів, включаючи наступне:

- Правила навчання Хебба.
- Правила навчання з виправленням помилок, також відомі як правила навчання Widrow-Hoff.
- Правила навчання на основі пам'яті.
- Правила випадкового навчання.

- Конкурентні правила навчання.

2.6.4 Класифікація нейронних мереж

Нейронні мережі зазвичай класифікують мережі по різним структурам, функцій і алгоритмам навчання, які можна розділити на:

- Нейронна мережа персептрона.

Найпростіший тип нейронної мережі має тільки одношарову структуру нейронної мережі та використовує функцію жорсткого обмеження в якості передавальної функції мережі, яка підходить для простих лінійних задач поділу на два класи.

- Лінійна нейронна мережа.

Нейронна мережа з одношарової структурою використовує лінійну функцію в якості передавальної функції мережі, яка підходить для таких додатків, як лінійна апроксимація і апроксимація даних.

- Нейронна мережа ВР

Найбільш широко використовувана мережа має багаторівневу мережеву структуру з одним або декількома прихованими шарами. Його структура показана на рисунку 2.5.

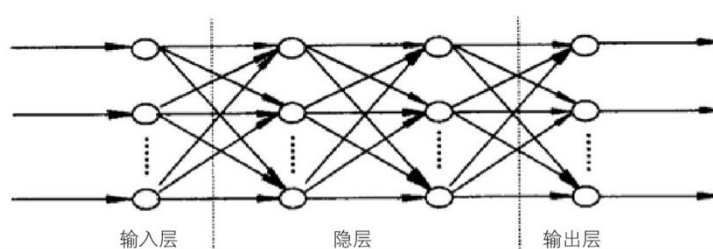


Рисунок 2.5 - Структура мережі ВР

Мережа ВР використовує алгоритм навчання Уїдрой-Хоффа і нелінійну ПФ. Типова мережа ВР використовує алгоритм градієнтного спуску, який передбачений алгоритмом Уїдрой-Хоффа.

BP, або зворотне поширення, відноситься до методу прямого поширення сигналу і зворотного поширення помилки при обчисленні градієнта для навчання нелінійної багатошарової мережі.

Для навчання мережі BP існує безліч базових алгоритмів оптимізації, таких як алгоритм змінного масштабу і алгоритм Ньютона. MATLAB Neural Network Toolbox надає безліч таких алгоритмів.

Використовуючи нелінійну ПФ, мережа BP може апроксимувати будь-яку нелінійну функцію з довільною точністю, що є дуже важливим показником.

Завдяки прихованій структурі середнього рівня мережа BP може витягувати статистичні властивості вищого порядку, особливо при великому масштабі вхідних даних здатність мережі витягувати статистичні властивості вищого порядку стає дуже важливою.

- Нейронна мережа з радіальної базисної функцією.

Нейронна мережа з радіальної базисної функцією також називається мережею RBF. Це така ж багатошарова пряма мережа, що і мережа BP. Вона також може апроксимувати будь-яку нелінійну функцію з довільною точністю. Просто передавальна функція, яка використовується цими двома мережами, відрізняється. Мережа BP зазвичай використовує сигмоїдну функцію або лінійну функцію в якості передавальної функції, в той час як мережа RBF використовує радіальну базисну функцію в якості передавальної функції.

Завдяки використанню радіальної базової передавальної функції мережа RBF має більш високу швидкість навчання, ніж мережа BP, що дозволяє уникнути громіздких та незручних обчислень в процесі зворотного поширення помилки, більше підходить для нових даних і має відносно гарну збіжність.

- Конкурентні нейронні мережі.

Характерною властивістю конкурентної нейронної мережі є те, що її різні нейрони знаходяться в конкуруючих відносинах. Багато нейронів кон-

курують один з одним, щоб визначити переможця. Нейрон-переможець визначає, який режим прототипу найкраще представляє режим введення. Отже, конкурентна нейронна мережа дуже підходить для завдання класифікації патернів.

2.7 Застосування нейронних мереж.

Історія розвитку нейронних мереж налічує 50 років, але фактичне застосування з'явилося в останні 15 років. В даний час нейронні мережі все ще швидко розвиваються, а типи додатків охоплюють всі сфери життя. Наведемо декілька прикладів областей застосування нейронних мереж:

- Медичні додатки

З постійною появою нових теорій, технологій і методів нейронних мереж, ступінь, в якій вони імітують людський інтелект, продовжує зростати, і було отримано безліч комерційних додатків. У медицині необхідно отримувати великий обсяг інформації під час операції, інтенсивної терапії та лікування, включаючи чисті дані, біологічні сигнали, зображення, текст та ін.

Технології нейронних мереж і інші технології штучного інтелекту використовуються в обробці сигналів, допоміжних експертних системах прийняття рішень на основі динамічних даних, інтелектуальному аналізі даних, прогнозуванні різних клінічних станів, інтелектуальному прикроватному моніторингу, телемедицині і навчанні персоналу для більш ефективного, безпечного та економічного обслуговування пацієнтів, медичних роботах і т.д.

- Обробка зображень

Самонавчання, самоадаптація і інші потужні можливості нейронної мережі по синтезу інформації визначають, що вона може добре імітувати функцію людського ока і виконувати виявлення кордонів, що важко для звичайних операторів кордону. У той же час алгоритм виявлення країв нейронної мережі може розпізнавати різницю між шумом і крайовим зображенням, і має

хороші антишумові характеристики, за допомогою навчання, можна видалити помилковий край, викликаний шумом. Використання нейронних мереж для виявлення меж цифрових зображень в останні роки стало гарячою точкою досліджень, з'явилася велика кількість теорій і методів, а використовувані алгоритми майже охоплюють всі типи нейронних мереж.

На мою думку, з постійними інноваціями в технології нейронних мереж технологія обробки зображень з використанням нейромережевої технології неминуче буде швидко розвиватися.

- *Економіка і фінанси*

В області економіки і фінансів нейромережеві методи в основному використовуються для прогнозування інвестицій, що в останні роки стало предметом досліджень.

2.8 Моделювання стиснення нейронної мережі ВР

Принцип стиснення нейронної мережі ВР такий: нейронна мережа ВР містить принаймні один прихований шар, тому в цілому формується тришарова мережа.

Потрібно виконати співставлення набору шаблонів введення з набором шаблонів виводу через невелику кількість прихованих шарів і зробити вихідні шаблони рівними з вхідними шаблонами, наскільки це можливо. Тому значення нейрона прихованого шару та відповідного вагового вектора можуть виводити вектор, такий самий як вихідний шаблон введення. Коли кількість нейронів у прихованому шарі відносно невелика, це означає, що прихований шар може використовувати меншу кількість чисел для представлення шаблону введення, а це насправді стиснення.

Перший шар - це вхідний шар, середній - прихований шар, функція відображення мережі залежить від прихованого шару. Перетворення від вхід-

ного шару до прихованого шару еквівалентно процесу кодування стисненням, а перетворення від прихованого шару до вихідного рівня еквівалентно процесу декодування.

РОЗДІЛ 3

3.1 Створення архітектури НМ ВР

Мережа ВР може вивчати та зберігати велику кількість вхідно-вихідних зв'язків відображення шаблонів, не розкриваючи математичного рівняння, що описує це співвідношення відображення заздалегідь. Його правило навчання полягає у використанні методу найкрутішого спуску, щоб безперервно коригувати ваги та пороги мережі за допомогою зворотного поширення, щоб мінімізувати суму квадратів помилок мережі. Топологічна структура моделі нейронної мережі ВР включає вхідний шар (input), прихований шар (hide layer) і вихідний шар (output layer) [4].

Модель нейрона ВР показана на рисунку 3.1.

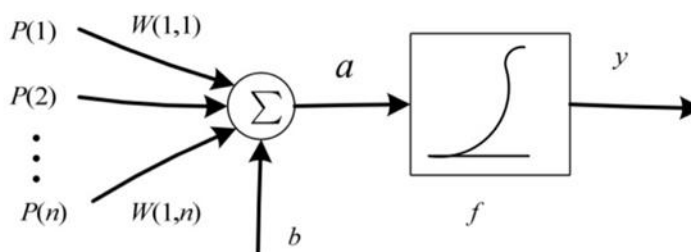


Рисунок 3.1 – Модель нейрона ВР

Вхід нейрона ВР дорівнює P , вага та поріг — w і b відповідно, а вихід моделі лінійного нейрона — y . Нейрон ВР подібний до інших нейронів, за винятком того, що ПФ нейрона ВР є нелінійною функцією. Найчастіше використовувані функції передачі — це функції logsig і tansig , як показано на рисунку 4.2. Іноді ПФ нейронної мережі ВР також використовує пурелін.

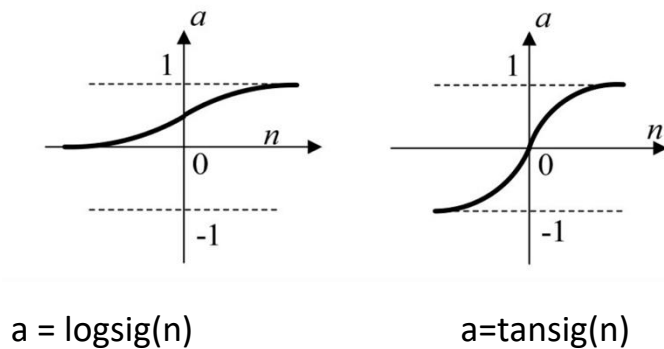


Рисунок 3.2 - Загальні ПФ нейронної мережі ВР

Мережа ВР є різновидом багаторівневої нейронної мережі з прямим зв'язком, яка складається з вхідного шару, прихованого шару та вихідного шару. На рисунку 4.3 показана структурна схема типової тришарової нейронної мережі ВР. Шари повністю взаємопов'язані. Між шарами немає взаємозв'язку. Прихований шар може мати один або кілька шарів.

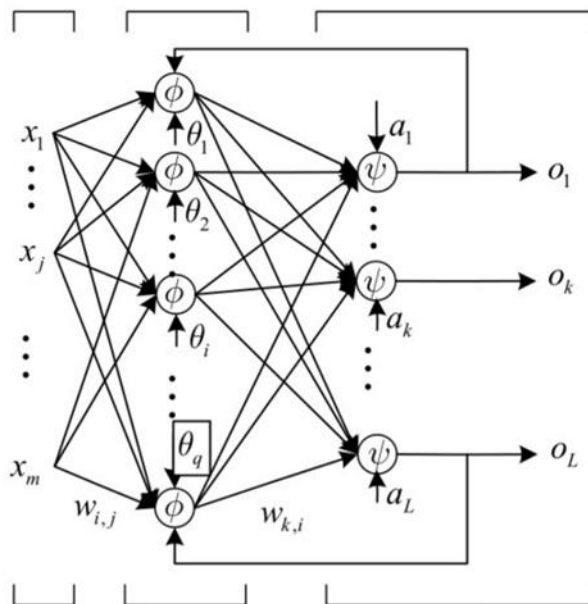


Рисунок 3.3 - Структурна схема типової тришарової нейронної мережі ВР

Між шарами діють два види сигналів. Одним з них є робочий сигнал (позначений суцільною лінією), який є сигналом, який поширюється вперед після застосування вхідного сигналу, доки на вихідному кінці не буде створено

фактичний вихід, і є функцією входу та ваги. Іншим сигналом є сигнал помилки (позначений пунктирною лінією) Різниця між фактичним виходом мережі та очікуваним виходом є помилкою, яка починає поширюватися назад шар за шаром від вихідного кінця. Процес навчання мережі ВР складається з процесу прямого обчислення та процесу зворотного поширення помилки. У процесі прямого обчислення вхідні дані обчислюються шар за шаром від вхідного шару через прихований шар і передаються до вихідного. Стан кожного шару нейронів впливає лише на стан наступного шару нейронів. Якщо вихідний рівень не може отримати потрібний результат, він увійде в процес зворотного поширення помилки, сигнал помилки повернеться по вихідному шляху з'єднання, а ваги та пороги кожного рівня мережі будуть коригуватися послідовно, поки не досягне вхідного шару, а потім обчислення напрямку буде повторено. Ці два процеси повторюються за раз, постійно коригуючи ваги та пороги кожного шару, щоб помилка мережі була мінімізована або час очікування був задовільним, і тоді процес навчання буде вважатися закінченим.

3.2 Навчання нейронної мережі ВР

Існує два різні методи навчання мережі ВР: метод збільшення та пакетний метод.

У методі збільшення ваги мережі та зміщення оновлюються щоразу, коли подаються вхідні дані. При пакетній обробці ваги мережі та зміщення оновлюються лише після того, як були подані всі вхідні дані.

Метод збільшення (застосування та адаптивна мережа та інші мережі) - хоча метод збільшення частіше застосовується до динамічних мереж, таких як адаптивна фільтрація, насправді його можна застосовувати і до статичних мереж також [5,6].

Метод збільшення в статичній мережі. Починаємо проводити навчання мережі поетапно. Щоразу, коли надходять вхідні дані, ваги мережі та зміщення оновлюються. Припустимо, що вхід нейронної мережі такий.

$p=[1\ 2; 2\ 1; 2\ 3; 3\ 1];$

Цільовий вихід наступний.

$t=[4\ 5\ 7\ 7];$

Встановимо лінійну функцію: $t=2p_1+p_2$.

Спочатку ініціалізуємо ваги та зміщення 0. Щоб показати ефект методу збільшення, швидкість навчання також встановлюється на 0. Конкретна процедура полягає в наступному:

$net = newlin([-1\ 1; -1\ 1], 1, 0, 0);$

$net.IW\{1,1\} = [0\ 0];$

$net.b\{1\} = 0;$

Щоб використовувати метод збільшення, вхідний вектор і цільовий вектор виражаються таким чином:

$P = \{[1; 2]\ [2; 1]\ [2; 3]\ [3; 1]\};$

$T = \{4\ 5\ 7\ 7\};$

Зазвичай, незалежно від того, чи вводиться вхідний вектор як синхронна векторна матриця чи як асинхронний векторний масив комірок, змодельоване вихідне значення є однаковим. Але під час навчання мережі імітаційне вихідне значення буде іншим.

При використанні функції адаптації для навчання нейронної мережі, якщо вхід є асинхронним вектором, тоді вага буде оновлюватися, коли буде подано кожен набір вхідних даних (тобто спосіб додавання); якщо вхід є синхронним вектором, тоді вага буде оновлюватися лише тоді, коли будуть подані всі вхідні дані (тобто пакетна обробка).

Почнемо навчати мережу поетапно:

$[net, a, e, pf] = \text{адаптувати}(net, P, T);$

Оскільки швидкість навчання дорівнює 0, вихід мережі також дорівнює 0, а ваги не оновлюються. Значення помилки та цільове значення не змінюються.

$a = [0 \ 0 \ 0 \ 0]$

$e = [4 \ 5 \ 7 \ 7]$.

Якщо швидкість навчання встановлена на 0,1, тоді, коли буде подано кожен набір вхідних даних, результати мережі будуть змінюватися.

```
net.inputWeights{1,1}.learnParam.lr=0,1;
```

```
net.biases{1,1}.learnParam.lr=0,1;
```

```
[net,a,e,pf] = адаптувати (net,P,T);
```

Результати навчання такі:

$a = [0 \ 2 \ 6,0 \ 5,8]$

$e = [4 \ 3 \ 1,0 \ 1,2]$.

Оскільки перші вхідні дані не були оновлені перед подачею, перше вихідне значення в цей час таке ж, як і перше вихідне значення, коли швидкість навчання дорівнює 0. Під час переходу до другого кроку ваги оновлюються, а другий вихід змінюється. Кожного разу, коли обчислюється помилка, ваги постійно змінюються. Якщо мережа можлива, а швидкість навчання встановлено належним чином, помилка продовжуватиме прямувати до нуля.

Метод збільшення в динамічній мережі.

Навчальні динамічні мережі також можуть бути розширені. Побудуємо нейронну мережу відповідно до наступної процедури, зробимо початкову вагу 0, а швидкість навчання встановимо 0,1.

```
net = newlin([-1 1],1,[0 1],0.1);
```

```
net.IW{1,1} = [0 0];
```

```
net.biasConnect = 0;
```

Щоб використовувати метод збільшення, вхідний вектор і цільовий вихід виражаються таким чином.

$P_i = \{1\}$;

$P = \{2 \ 3 \ 4\}$;

$T = \{3 \ 5 \ 7\}$;

Використовуємо adapt для навчання мережі.

```
[net,a,e,pf] = адаптувати (net,P,T,Pi);
```


Результати навчання такі:

$a = [0 \ 2,4 \ 7,98]$

$e = [3 \ 2,6 \ -1,98]$

Оскільки вага не оновлюється, перший вихід дорівнює 0, але кожен наступний крок послідовності змінюється один раз.

При *пакетній обробці* ваги та зміщення мережі оновлюються лише після того, як усі вхідні дані були подані. Це також можна застосувати до статичних та динамічних мереж.

Метод пакетної обробки в статичній мережі. Пакетна обробка може бути реалізована за допомогою функцій адаптації або навчання. Завдяки використанню більш ефективних алгоритмів навчання, `train`, як правило, є найкращим вибором. Метод збільшення можна реалізувати лише за допомогою адаптації, а функцію `train` можна використовувати лише в пакетній обробці.

Застосуємо мережу, яка використовується в статичній мережі з методом збільшення, і швидкість навчання встановимо на 0,1.

```
net = newlin([-1 1;-1 1],1,0,0.1);
```

```
net.IW{1,1} = [0 0];
```

```
net.b{1} = 0;
```

Функція адаптації використовується для реалізації пакетної обробки статичних мереж, а вхідний вектор повинен бути розміщений у вигляді синхронної векторної матриці.

```
P = [1 2 2 3; 2 1 3 1];
```

```
T = [4 5 7 7];
```

Виклик функції адаптації запустить функцію `adaptwb`, яка є функцією лінійного налаштування мережі за замовчуванням. `learnwh` — це функція навчання ваги та зміщення за замовчуванням. Тому буде використовуватися метод навчання Уідроу-Хоффа, а код MATLAB такий.

```
[net,a,e,pf] = adapt(net,P,T);
```

Результати навчання такі:

```
>> a
```

```
a = 0 0 0 0
```

```
>> e
```

```
e = 4 5 7 7
```

Оскільки вагові коефіцієнти не оновлюються до того, як будуть подані всі дані, які підлягають навчанню, на виході мережі всі сигнали 0. Якщо вага відображається в програмі, буде отримано наступний результат:

```
>> net.IW{1,1}
```

```
ans =
```

```
4,9000 4,1000
```

```
>> net.b{1}
```

```
ans =
```

```
2,3000
```

Далі використовуємо функцію `train` для реалізації пакетної обробки. Оскільки правило Відрю-Хоффа можна застосувати в режимі збільшення та пакетного режиму, його можна запустити за допомогою адаптації та навчання. У нас є кілька алгоритмів, які можна використовувати лише в пакетній обробці (особливо алгоритм Левенберга-Марквардта). Використовуючи функцію `train`, будь-який асинхронний векторний масив комірок буде перетворено в синхронну векторну матрицю. Це пояснюється тим, що мережа є статичною і тому, що запуск завжди використовується в пакетному режимі. Оскільки *MATLAB* є більш ефективним для реалізації синхронного режиму, тому, коли це можливо, завжди використовуємо синхронний режим для обробки.

Метод збільшення в динамічній мережі. Навчання статичної мережі є відносно простим. Якщо функція `train` використовується для навчання мережі, навіть якщо вхід є асинхронним вектором, його буде перетворено в синхронний вектор і використано пакетну обробку. Якщо ми використовуємо функцію адаптації, формат введення визначає метод навчання мережі. Якщо передача є послідовністю, мережа використовує метод збільшення, якщо пе-

редача є вектором синхронізації, мережа використовує метод пакетної обробки. У динамічних мережах, особливо коли існує лише одна навчальна послідовність, пакетну обробку можна виконувати лише за допомогою функції `train`. Обираємо мережу, яка використовується в статичній мережі з методом збільшення, і встановлюємо швидкість навчання на 0,02.

```
net = newlin([-1 1],1,[0 1],0.02);
```

```
net.IW{1,1}=[0 0];
```

```
net.biasConnect=0;
```

```
net.trainParam.epochs = 1;
```

```
ni = {1};
```

```
P = {2 3 4};
```

```
T = {3 5 6};
```

Проводимо тренування з набором даних, навченими попереднім методом збільшення, але оновлювати вагові коефіцієнти потрібно лише після того, як усі дані будуть подані (пакетна обробка). Оскільки вхідні дані є послідовністю, мережа буде моделюватися в асинхронному режимі, а ваги будуть оновлюватися в пакетному режимі.

```
net=train(net,P,T,Pi);
```

Після одного тренування значення ваг наступні:

```
>>net.IW{1,1}
```

```
ans = 0,9000 0,6200
```

Значення ваг тут відрізняється від того, яке ми отримуємо за допомогою методу збільшення. У режимі збільшення, через налаштування тренування, одне тренування може оновлювати ваги 3 рази. У пакетному режимі кожне навчання можна оновити лише один раз.

3.3 Функція нейронної мережі `VR`

Переважна більшість моделей нейронних мереж використовує нейронну мережу ВР та її варіації, які також є основною частиною прямої мережі і втілюють суть штучної нейронної мережі. Мережа ВР в основному використовується в наступних чотирьох аспектах.

- апроксимація функції: потрібно навчити мережу з вхідним вектором і відповідним вихідним вектором для апроксимації функції.
- розпізнавання образу: використовуємо вихідний вектор, що очікує на розгляд, щоб пов'язати його з вхідним вектором.
- класифікація: класифікуємо відповідний метод, визначений вхідним вектором.
- стиснення даних: зменшуємо розмірність вихідного вектору для передачі або зберігання зображень.

3.4 Приклад та результати програмної реалізації навчання мережі ВР в пакеті Matlab

Matlab і Simulink містять багато функцій панелі інструментів для проектування та аналізу додатків нейронної мережі. На даний момент останній інструментарій нейронних мереж майже повністю узагальнює нові досягнення існуючих нейронних мереж.

Лістинг програми:

```
%% 使用 bp 神经网络对图像进行不同程度的压缩

clc
clear

I=imread('image.png'); %% 读入图像

I=imresize(I,[128,128]);

figure(1)

imshow(I),title('Original image');

P=[];
```

```

%% 将原始图像分成 4*4 的互不重叠的像素块，16*1 的列向量。

%% 并将原始数据转化 16*1024 的矩阵。
for i=1:32
    for j=1:32
        I2=I((i-1)*4+1:i*4,(j-1)*4+1:j*4);
        i3=reshape(I2,16,1);
        II=double(i3);

                %% 矩阵归一化

        P_1=II/255;
        P=[P,P_1];
    end
end

%% 神经网络输入与输出采用同一矩阵

T=P;

%% 构建 BP 网络，隐含层为 8，采用 trainlm 算法

net=newff(minmax(P),[8,16],{'tansig','logsig'},'trainlm'
);
net.trainParam.goal=0.001;
net.trainParam.epochs=500;

%% 训练网络

tic
net=train(net,P,T);
toc

%% 编码仿真结果

Y_chonggou=sim(net,P);

%% 图像重建

```

```

Ychonggou_ceshi=[];
for k=1:1024
    Ychonggou_ceshi1=reshape(Y_chonggou(:,k),4,4);
    Ychonggou_ceshi=[Ychonggou_ceshi,Ychonggou_ceshi1];
end
YYchonggou_ceshi=[];
for k=1:32
    YYchonggou_ceshi1=Ychonggou_ceshi(:,(k-
1)*128+1:k*128);

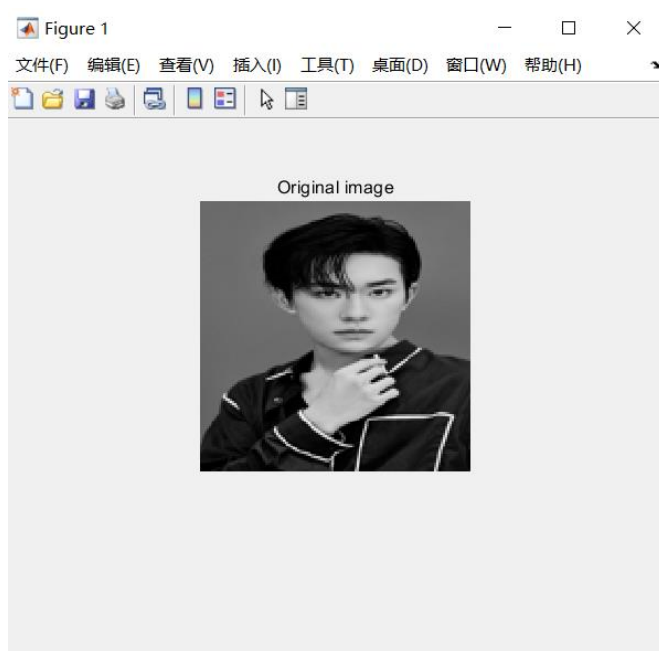
YYchonggou_ceshi=[YYchonggou_ceshi;YYchonggou_ceshi1];
end

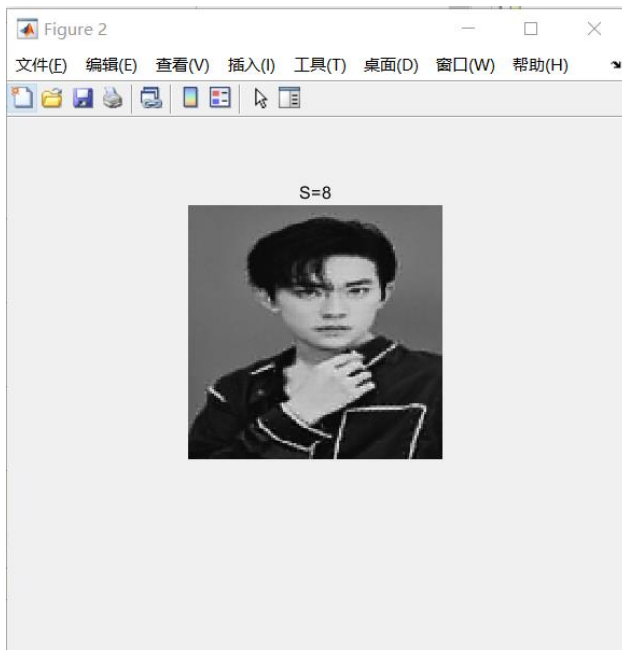
Ychonggou_ce=uint8(YYchonggou_ceshi*255);
figure(2)

imshow(Ychonggou_ce),title('S=8'); %修改

```

S=8





Neural Network Training (nntraintool)

Neural Network

Algorithms

Training: Levenberg-Marquardt (trainlm)
 Performance: Mean Squared Error (mse)
 Calculations: MEX

Progress

Epoch:	0	22 iterations	500
Time:	0:00:01		
Performance:	0.285	0.000994	0.00100
Gradient:	0.0522	0.000371	1.00e-07
Mu:	0.00100	1.00e-07	1.00e+10

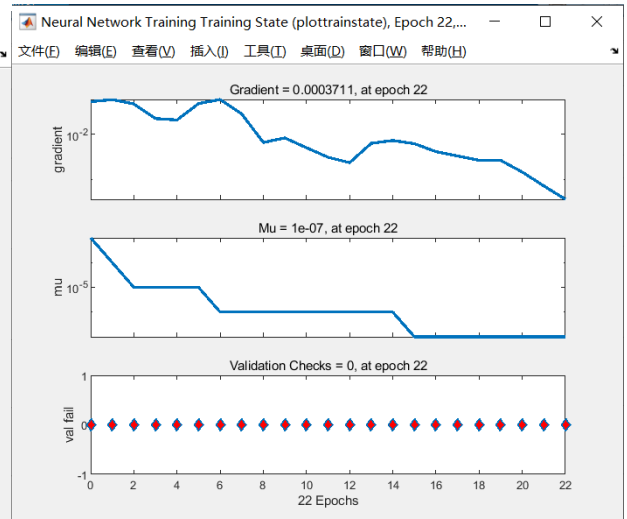
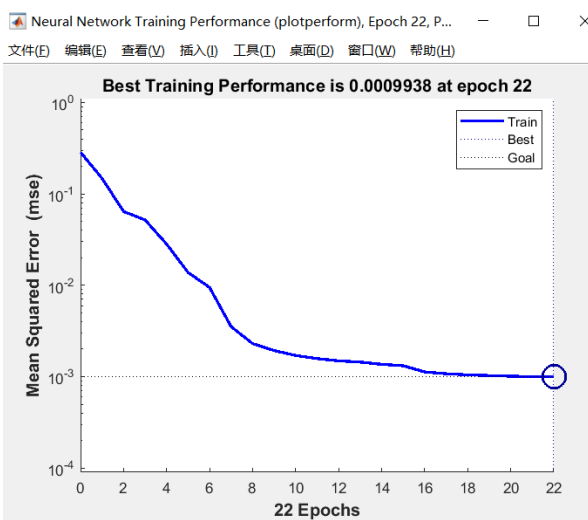
Plots

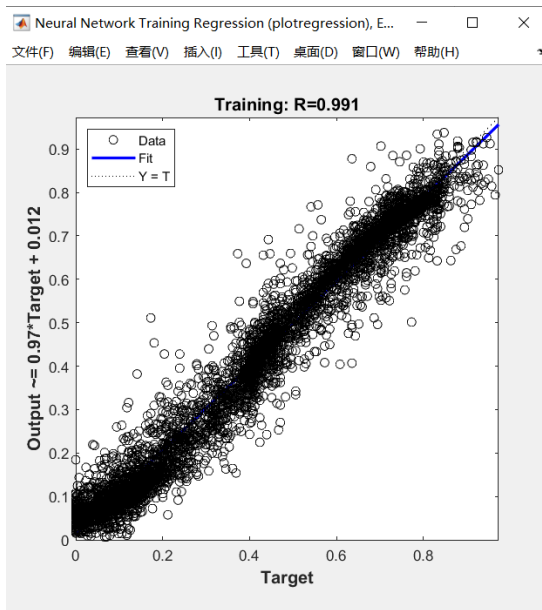
Performance (plotperform)
 Training State (plottrainstate)
 Regression (plotregression)

Plot Interval: 1 epochs

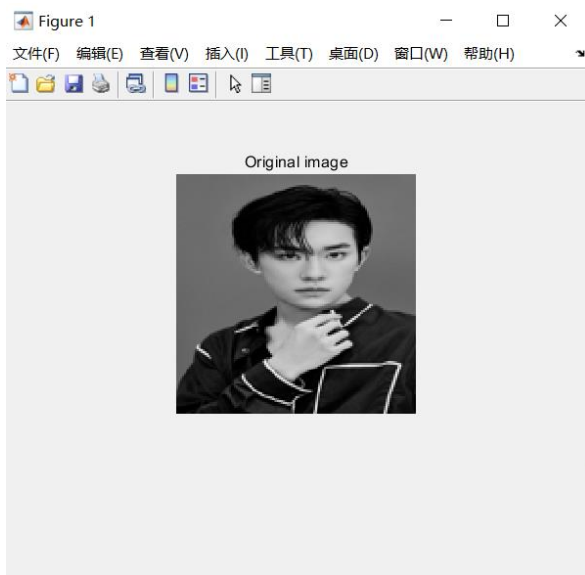
✓ Performance goal met.

Stop Training Cancel





S=2



Neural Network Training (nntraintool) — □ ×

Neural Network

Algorithms

Training: Levenberg-Marquardt (trainlm)
 Performance: Mean Squared Error (mse)
 Calculations: MEX

Progress

Epoch:	0	500 iterations	500
Time:		0:00:07	
Performance:	0.316	0.00641	0.00100
Gradient:	0.0378	3.83e-05	1.00e-07
Mu:	0.00100	1.00e-11	1.00e+10

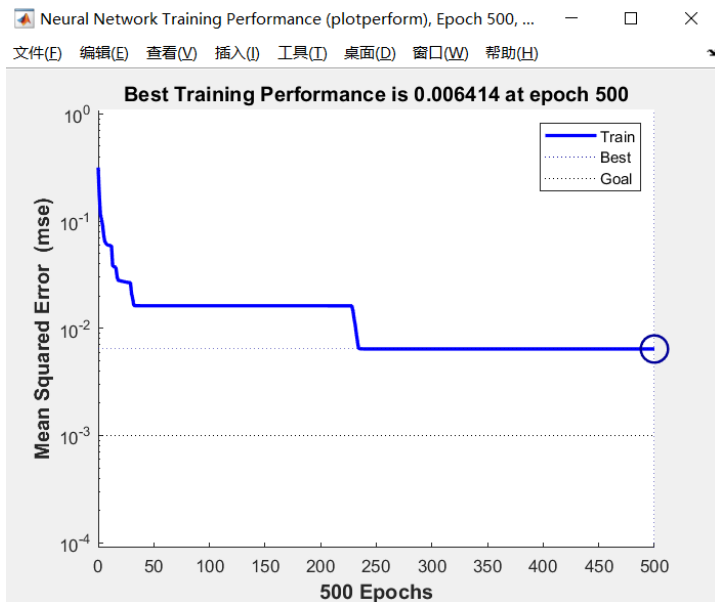
Plots

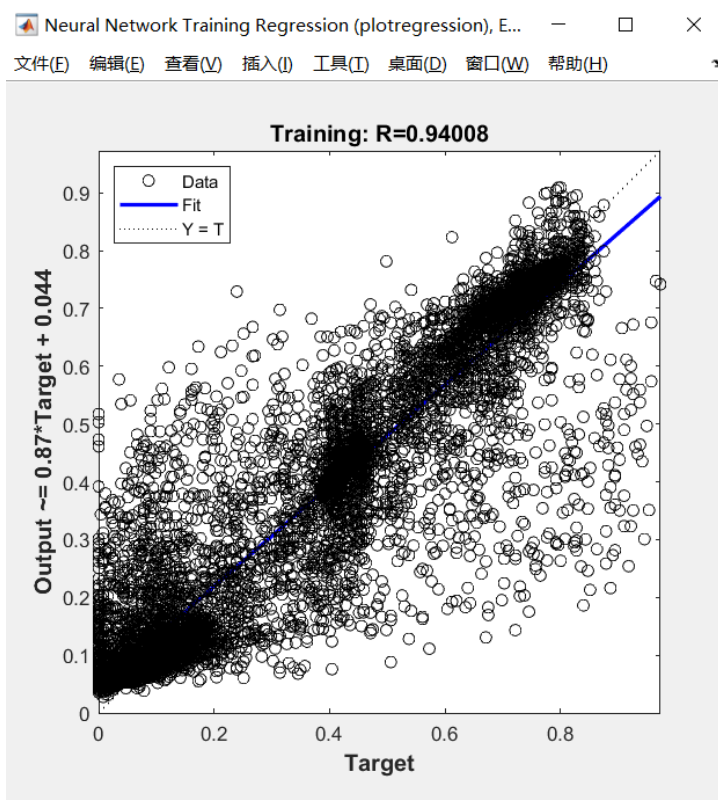
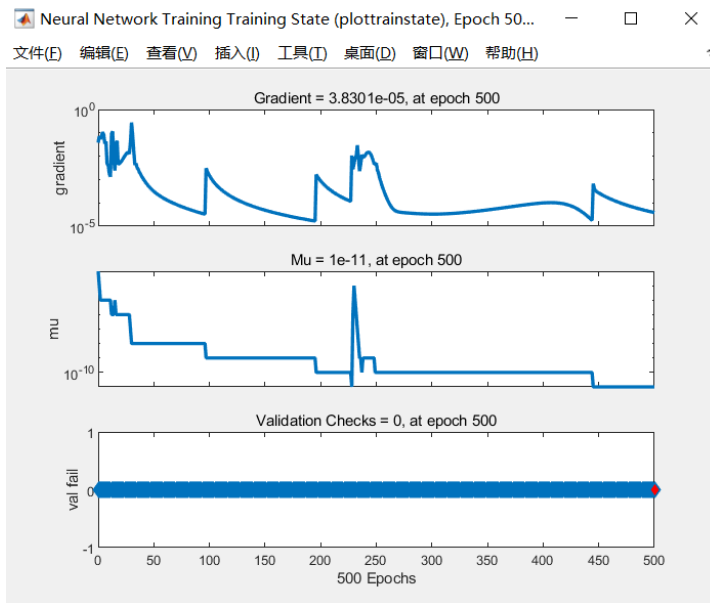
Performance (plotperform)
 Training State (plottrainstate)
 Regression (plotregression)

Plot Interval: 1 epochs

✔ Opening Regression Plot

⏹ Stop Training ⏹ Cancel





Нейронна мережа BP зараз є відносно зрілою мережевою моделлю, оскільки нейронна мережа має невід'ємні переваги при обробці цифрових зображень, особливо при стисненні зображень. Отже, з погляду візуального ефекту після стиснення, кількість вузлів нейронної мережі у прихованому шарі впливає на ефект стиснення зображення у певному діапазоні.

При великій кількості вузлів прихованого шару ступінь стиснення нижче та якість стисненого відновленого зображення краща, але цей ефект не є лінійним.

Якість навчання мережі також істотно впливає на стиснення зображення, яке можна розглядати як стиснення із втратами.

В цілому, чим більше шарів, тим менше помилка всієї мережі, але це ускладнює всю мережу і збільшує час навчання, мережа також може бути "перенасиченою".

Підсумовуючи можна сказати, що один або два приховані шари можуть вирішити багато проблем. Якщо обсяг даних великий, кількість шарів можна відповідним чином збільшити, не допускаючи надмірного припасування.

Для різних моделей кількість прихованих шарів і кількість вузлів прихованого шару різняться, а такі фактори, як розміри введення та виведення, складність мережі, кількість навчальних вибірок та розмір помилки повинні також розглядатися в обов'язковому порядку.

Якщо кількість вузлів невелика, модель не має здатності до узагальнення та не має жодного ефекту; якщо вузлів більше, легко потрапити до локального оптимуму.

ВИСНОВКИ

Розглянута в роботі та змодельована нейронна мережа ВР на відміну від інших архітектур НМ (Хопфілда та ін.) має розподілене зберігання та паралельну обробку інформації, вона має велику перевагу в швидкості для характеристик масивних даних зображення;

Тестування в пакеті Matlab довело її здатність до одночасної роботи з кількома блоками, а також її відмовостійкість і надійність можуть подолати недоліки, які виникають в інших мережах при зберігання даних зображення. Під час передачі зберігається якість стиснення і декомпресії зображення.

Нейронна мережа ВР — це багаторівнева мережа з прямим зв'язком. Дослідження та тестування показали, що продуктивність мережі тісно пов'язана з її вхідним сигналом, кількістю шарів, кількістю нейронів у кожному шарі, передавальною функцією, алгоритмом навчання, початковою вагою та очікуваною помилкою.

В подальших роботах планується при реальному проектуванні вище зазначені умови детально розглянути відповідно до конкретних вимог (для якої цілі, застосунку, радіотехнічного пристрою відбувається проектування НМ).

Використовуючи вдосконалений алгоритм стиснення зображень мережі ВР, можна покращити якість і швидкість стиснення стиснутих зображень.

3. ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Володіння нейронною мережею MATLAB. Чжу Кай, Ван Чженлін. 2010 рік.
2. Лі Чонг. Дослідження алгоритму стиснення зображень у реальному часі на основі нейронної мережі [D]. Пекін: Пекінський університет пошти та телекомунікацій. 2019 рік.
3. Аналіз типового випадку нейронної мережі MATLAB R2020a. Чжан Дефен. 2021 рік.
4. Посібник із супер навчання нейронних мереж MATLAB. MATLAB Technology Alliance, Лю Бін, Го Хайся. 2014 рік.
5. Аналіз 43 випадків нейронної мережі MATLAB. Під редакцією Ван Сяочуань, Ши Фен, Ю Лей та Лі Ян. 2011 рік.
6. Чжуге Музи, Лі Вень, Чжан Сінхуей. Дослідження методу стиснення цифрових зображень на основі нейронної мережі BP [A]. Відділення індустріалізації інтелектуальної обробки інформації Китайської асоціації досліджень високотехнологічної індустріалізації. Спеціальний випуск 10-ї Національної конференції з обробки та застосування сигналів та інтелектуальної інформації [C]. Відділення індустріалізації інтелектуальної обробки інформації Китайської дослідницької асоціації високотехнологічної індустріалізації: Китайська дослідницька асоціація високотехнологічної індустріалізації, 2016, 5
7. Реалізація стиснення зображення нейронною мережею BP. [Електронний ресурс]. – 2017. – Режим доступу до ресурсу: <https://www.jianshu.com/p/f129d1d73a1d>
8. Реалізація нейронної мережі BP та matlab. – 2012. – Режим доступу до ресурсу: <https://blog.csdn.net/gongxq0124/article/details/7681000>
9. Як навчити нейронну мережу? [Електронний ресурс]. – 2017. – Режим доступу до ресурсу: https://blog.csdn.net/Jaster_wisdom/article/details/78018653
10. Навчання та моделювання нейронної мережі MATLAB.

[Електронний ресурс]. – 2014. – Режим доступу до ресурсу:
<https://blog.csdn.net/hzq20081121107/article/details/17755271>

11. Знайомство з нейронними мережами. [Електронний ресурс]. – 2017.
– Режим доступу до ресурсу: <https://www.ibm.com/cn-zh/cloud/learn/neural-networks>

12. Стиснення зображень на основі глибокого навчання. [Електронний ресурс]. – 2017. – Режим доступу до ресурсу:
<https://blog.csdn.net/intflojx/article/details/85383560>

13. Вступ до штучної нейронної мережі. [Електронний ресурс]. – 2017.
– Режим доступу до ресурсу:
<https://baike.baidu.com/item/%E4%BA%BA%E5%B7%A5%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C/382460>

14. Технологія стиснення зображень на основі нейронної мережі. [Електронний ресурс]. – 2018. – Режим доступу до ресурсу:
<https://cloud.tencent.com/developer/article/1033691?from=15425>

15. Реалізація стиснення зображення через нейронну мережу VP. [Електронний ресурс]. – 2018. – Режим доступу до ресурсу:
<https://cloud.tencent.com/developer/article/1091433?from=15425>

16. Застосування та розвиток нейронної мережі. [Електронний ресурс]. – 2018.
– Режим доступу до ресурсу:
<https://blog.csdn.net/lewyu521/article/details/85226541>

17. Як стиснути картинки. [Електронний ресурс]. – 2017. – Режим доступу до ресурсу:
<https://www.jianshu.com/p/5ade0e1dd279>

18. Метод стиснення нейронної мережі. [Електронний ресурс]. – 2021. – Режим доступу до ресурсу:
https://m.sohu.com/n/478682331/?wscrid=95360_6

19. Використовуйте згорткові нейронні мережі (CNN) для обробки складних зображень. [Електронний ресурс]. – 2019. – Режим доступу до ресурсу:

<https://developers.google.com/codelabs/tensorflow-5-compleximages?hl=zh-cn#0>

20. Hamdy S.Soliman.Mohammed Omari .A neural networks approach to image data compression.[J].Applied Soft Computing.2006.6 : 258~271.

21. Лі Ліфень, Чжао Хуайю, Лю Шуган Застосування нейронної мережі в стисканні зображень Дослідження.[J].Journal of North China Electric Power University.2002.29.2:91~93.

22. Чжан Цзяньмін, Чжан Чжицай, Ченг Кеян та ін Дослідження та розвиток глибокого навчання [J] Journal of Jiangsu University (Видання природничих наук), 2015, 36(2):191-200.

23. Лі Юнджун Дослідження алгоритму стиснення зображень і відео низької складності [D]. Сіань: Університет Сідянь, 2017.