# Форма № Н-9.02

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА КОМП’ЮТЕРНОЇ ІНЖЕНЕРІЇ

КАФЕДРА ІНФОРМАТИКИ І ВИЩОЇ МАТЕМАТИКИ

## Пояснювальна записка

до випускної роботи бакалавра

Бакалавр

на тему **АЛГОРИТМИ І МЕТОДИ РОЗПІЗНАВАННЯ З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ.**

Виконав: студент 4 курсу, групи I-13-1

напряму підготовки

6.040302 – «Інформатика»

Парфенюк В. І.

Керівник Славко Г. В.

Рецензет Юдіна Г. Л.

Кременчук – 2017 року

### Форма № Н-9.01

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

Факультет електроніки та комп'ютерної інженерії

Кафедра інформатики і вищої математики

Освітньо-кваліфікаційний рівень – бакалавр

Напрям підготовки 6.040302 – “Інформатика”

**ЗАТВЕРДЖУЮ**

**Завідувач кафедри**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

"\_\_\_" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_р.

**ЗАВДАННЯ**

**НА ВИПУСКНУ РОБОТУ БАКАЛАВРА**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Парфенюк Віталій Іванович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище, ім'я, по батькові)

1. Тема випускної роботи **\_**Алгоритми і методи розпізнавання з використанням нейронних мереж \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_керівник роботи \_\_Славко Генадій Володимирович к.т.н., доцент\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від "\_\_\_" \_\_\_\_\_\_ 20\_ р. № \_\_

2. Строк подання студентом роботи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. Вихідні дані до роботи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| 6. Консультанти розділів роботи |

|  |  |  |  |
| --- | --- | --- | --- |
| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
| Завдання видав | Завдання  прийняв |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

7. Дата видачі завдання\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | Назва етапів випускної роботи | Строк  виконання  етапів роботи | Примітка |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Студент\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Керівник роботи**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**РЕФЕРАТ**

Пояснювальна записка до випускної роботи бакалавра: 54 сторінок, 23 рисунки, 18 літературних джерел, 2 додатка.

Мета роботи – розробка нейронної мережі для розпізнавання образів.

Об’єкт дослідження – розпізнавання образів.

Предмет дослідження – нейронні мережі для розпізнавання.

Методи дослідження і розробки – штучні нейронні мережі із застосуванням процесу розпізнавання без вчителя, програмування на мові Java Script у веб-сервісі GitHub.

Випускна робота складається з вступу, чотирьох розділів, списку використаної літератури та додатків.

Перший розділ описує основні поняття, моделі, методи та алгоритми нейронних мереж.

Другий розділ представляє огляд основних методів розпізнавання образів за допомогою нейронних мереж.

Третій розділ реалізує нейронну мережу розпізнавання образів.

Четвертий розділ розглядає питання з охорони праці та безпеки в надзвичайних ситуаціях.

**Ключові слова:** НЕЙРОННІ МЕРЕЖІ, РОЗПІЗНАВАННЯ ОБРАЗІВ, АЛГОРИТМИ РОЗПІЗНАВАННЯ.

**РЕФЕРАТ**

Пояснительная записка к выпускной работе бакалавра: 54 страниц, 23 рисунка, 18 литературных источников, 2 приложения.

Цель работы – разработка нейронной сети для распознавания образов.

Объект исследования – распознавание образов.

Предмет исследования – нейронные сети для распознавания.

Методы исследования и разработки – искусственные нейронные сети с применением процесса распознавания без учителя, программирование на языке Java Script в веб-сервисе GitHub.

Выпускная работа состоит из вступления, четырех разделов, списка использованной литературы и приложений.

Первый раздел описывает основные понятия, модели, методы и алгоритмы нейронных сетей.

Второй раздел представляет обзор основных методов распознавания образов с помощью нейронных сетей.

Третий раздел реализует нейронную сеть для распознавания образов.

Четвертый раздел рассматривает вопросы охраны труда и безопасности в чрезвычайных ситуациях.

**Ключевые слова:** НЕЙРОННЫЕ СЕТИ, РАСПОЗНАВАНИЕ ОБРАЗОВ, АЛГОРИТМЫ РАСПОЗНАВАНИЯ.

**ЗМІСТ** Вступ………………………………………………………………………………… 8 Розділ 1 Структура та властивості штучних нейронних мереж.…………….. 9

1.1 Біологічний нейрон і нейронні мережі ……….…….. ………………………..9

1.2 Модель штучного нейрона ………………………………………………… 11

1.3Штучні нейронні мережі**….**………………………………….…….…………12

1.3.1 Одношарові штучні нейронні мережі………………………………………12

1.3.2 Багатошарові штучні нейронні мережі……………………………………..13

1.3.3 Нейронні мережі Хеба………………………………………………………..14

1.3.4 Нейронні мережі Хопфілда……………………………………………..…17

1.3.5 Нейронні мережі Хемінга……………………………………………….…19

1.3.6 Мережа Кохонена…………………………………………………………..22

Розділ 2 Ррозпізнавання образів ……………………………………………… 25

2.1 Поняття образу……………………………………………………………… 25

2.2 Проблема розпізнавання образів ……...................................................... 26

2.3 Геометричний та структурний підходи ……………………………………... 27

2.4 Навчання, самонавчання і адаптація………………………………………..30

2.5 Алгоритмічні побудови……………………………………………………...33

2.5.1 Навчання штучних нейронних мереж…………………………………….33

2.5.2 Навчання з вчителем………………………………………………………..34

2.5.3 Навчання без вчителя…………………………………………………….…34

2.5.4 Процес навчання нейронних мереж………………………………………35

2.6 Алгоритм січних площин…………………………………………………….36

2.7 Алгоритми, засновані на методі потенціалів………………………………...37

2.8 Приклад алгоритму навчання Персептрона з нерекуррентной мережею..38

2.9 Метод навчання Уидроу-Хоффа…………………………………………….39

2.10 Нейро-мережева самоорганізація…………………………………………….39

Розділ 3 Програмна реалізація ………………………….……………………… 42

3.1 Постановка задачі………………………………………………………………42

3.2 Програмне забезпечення……………………………………………………….42

3.3 Робота програми……………………………………………………………..42

Розділ 4 Охорона праці………………………………………………...…………..45 4.1 Шкідливі речовини в повітрі робочої зони………………………………...…45

4.2 Розрахунок штучного освітлення робочого приміщення………….…….…..46

4.3 Електобезпека робочих місць……………………………………..……….. 49

4.4 Пожежна безпека виробничих приміщень…..………………………..………51 Висновки..……………..…………………………………………..………………. 55 Список літератури……………………………………..………………..………… 55

ДОДАТОК А…….……………………………………..………………..………… 57

**ВСТУП**

Основні завдання, які ставляться перед нейронними мережами, належать до задач розпізнавання образів. Вони полягають в тому, щоб класифікувати вхідний образ, тобто віднести його до якого-небудь відомого мережі класу. Спочатку мережі даються еталонні образи - такі образи, належність яких до певного класу відома. Потім на вхід мережі подається деякий невідомий образ, і мережа намагається по певному алгоритму співвіднести його з яким-небудь еталонним образом. Можна сказати, що нейромережі проводять кластеризацію образів. Так як кластерний аналіз застосовується дослідниками ринку цінних паперів, то нейронні мережі можуть бути використані і для прогнозування вартості акцій, що є актуальною задачею, до того ж суворо нерозв'язною на даний момент.

Так само нейронный мережі можуть застосовуватися в криміналістиці (аналіз відбитків пальців) або ж для полегшення роботи правоохоронних органів в затриманні злочинців, тому що зараз вже створені такі програми, які розпізнають обличчя.

Все вищенаведене свідчить про те, що проблеми дослідження можливостей нейронних мереж та їх розвитку є актуальною на даний момент часу.

**Актуальність** програми обумовлена популярністю на ринку програмних продуктів такого типу (для розпізнавання образів) і їх широкого застосування у КПК, смартфонах, ноутбуках, т.д. Програмне забезпечення для розпізнавання образів розробляється багатьма компаніями і є одним із інноваційних продуктів на сучасному ринку.

**Мета роботи** – розробка нейронної мережі для розпізнавання образів.

**Об’єкт дослідження** – розпізнавання образів.

**Предмет дослідження** – нейронні мережі для розпізнавання.

**Методи дослідження і розробки** – штучні нейронні мережі із застосуванням процесу розпізнавання без вчителя, програмування на мові Java Script у веб-сервісі GitHub.

**РОЗДІЛ 1**

**СТРУКТУРА ТА ВЛАСТИВОСТІ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ**

**1.1 Біологічний нейрон і нейронні мережі**

Кожен нейрон володіє багатьма якостями, загальними з іншими елементами тіла, але його унікальною здатністю є прийом, обробка і передача електрохімічних сигналів по нервових шляхах, які утворюють комунікаційну систему мозку.

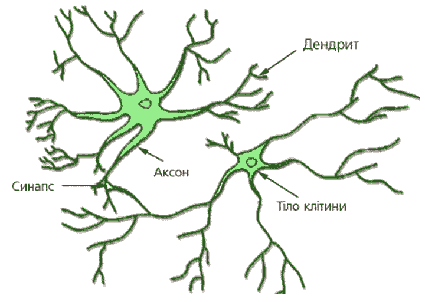


Рисунок 1.1 – Структура типових біологічних нейронів

Нейрон складається з трьох частин: тіла клітини, дендритів і аксона, кожна частина зі своїми, але взаємопов'язаними функціями. Дендрити йдуть від тіла нервової клітини (сома) до інших нейронів, де вони приймають сигнали в точках з'єднання, які називаються синапсами (рис. 1.1).

Прийняті синапсом вхідні сигнали підводяться до тіла нейрона. Тут вони підсумовуються, причому одні входи прагнуть порушити нейрон, інші – перешкодити його порушення. Коли загальне збудження в тілі нейрона перевищує деякий поріг, нейрон збуджується, посилаючи по аксону сигнал іншим нейронам. У цієї основної функціональної схеми багато ускладнень і винятків, тим не менш більшість штучних нейронних мереж моделюють лише прості властивості.

Нервові клітини, або нейрони, являють собою особливий вид клітин у живих організмах, що мають електричною активністю, основне призначення яких полягає в оперативному управлінні організмом. Сома, як правило, має поперечний розмір кілька десятків мікрон. Довжина дендритів може досягати 1 мм, дендрити сильно гілкуються, пронизуючи порівняно великий простір в околиці нейрона.

Довжина аксона може досягати сотень міліметрів. На сомі і на дендритах розташовуються закінчення (колатералі) аксонів, що йдуть від інших нервових клітин. Кожне таке закінчення має вигляд потовщення, званого синаптичної бляшкою, або синапсом.

Поперечні розміри синапсу, як правило, не перевищують декількох мікрон, найчастіше ці розміри становлять близько 1 мкм.[5]

Вхідні сигнали дендритної дерева (постсинаптичні потенціали) зважуються і підсумовуються на шляху до аксонному горбку, де генерується вихідний імпульс-спайк) або пачка імпульсів. Його наявність (або інтенсивність), отже, є функцією зваженої суми вхідних сигналів. Вихідний сигнал проходить по гілках аксона і досягає синапсів, які з'єднують аксони з дендритными деревами інших нейронів. Через синапси сигнал трансформується в новий вхідний сигнал для суміжних нейронів. Цей вхідний сигнал може бути позитивним і негативним (збуджують або гальмують) в залежності від виду синапсів. Величина вхідного сигналу, генерованого синапсом, може бути різною навіть при однаковій величині сигналу, що приходить в синапс. Ці відмінності визначаються ефективністю або вагою синапсу. Синаптична вага може змінюватися в процесі функціонування синапсу.

Нейрони можна розбити на три великі групи: рецепторні, проміжні і ефекторні. Рецепторні нейрони забезпечують введення в мозок сенсорної інформації. Вони трансформують сигнали, що надходять на органи почуттів (оптичні сигнали в сітківці ока, акустичні у вушний равлику або нюхові в хеморецепторах носа), в електричну імпульсацію своїх аксонів. Ефекторні нейрони передають приходять на них сигнали виконавчим органам. На кінці їх аксонів є спеціальні синаптичні з'єднання з виконавчими органами, наприклад м'язами, де збудження нейронів трансформується в скорочення м'язів. Проміжні нейрони здійснюють обробку інформації, одержуваної від рецепторів, і формують керуючі сигнали для ефекторів. Вони утворюють центральну нервову систему.[2]

**1.2 Модель штучного нейрона**

Штучний нейрон імітує в першому наближенні властивості біологічного нейрона.

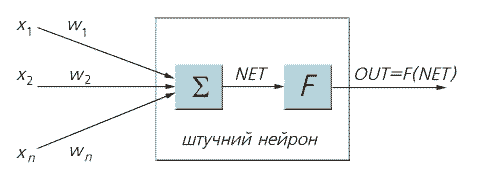


Рисунок 1.2 – Штучний нейрон

На вхід штучного нейрона надходить деяка множина сигналів, кожен відповідну вагу, аналогічну синаптичної силі, і всі добутки підсумовуються, визначаючи рівень активації нейрона. Хоча мережеві парадигми досить різноманітні, в основі майже всіх їх лежить ця конфігурація. Тут множина вхідних сигналів, позначених x1, x2,..., xn, надходить на штучний нейрон. Ці вхідні сигнали, в сукупності, позначаються вектором X, відповідають сигналам, що приходять в синапси біологічного нейрона. Кожен сигнал помножується на відповідну вагу w1, w2,..., wn, і надходить на підсумовуючий блок, позначений. Кожна вага відповідає «силі» одного біологічного синаптичної зв'язку. (Множина ваг в сукупності позначається вектором W) (рис. 1.2).

Підсумовуючий блок, що відповідає тілу біологічного елемента, складає зважені входи алгебраїчно, створюючи вихід – NET.

Сигнал NET перетворюється активаційний функцією F і дає вихідний нейронный сигнал OUT. Актіваціонная функція може бути звичайною лінійною функцією де К – постійна, порогової функції де Т – деяка постійна порогова величина, або ж функцією, більш точно моделює нелінійну передатну характеристику біологічного нейрона і представляє нейронної мережі великі можливості.[5]

**1.3 Штучні нейронні мережі**

**1.3.1 Одношарові штучні нейронні мережі**

Хоча один нейрон і здатний виконувати найпростіші процедури розпізнавання, сила нейронних обчислень виникає від з'єднань нейронів в мережах. Найпростіша мережа складається з групи нейронів, що утворюють шар, що показано на рис. 1.3.

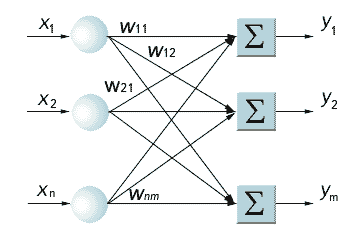


Рисунок 1.3 – Одношарова нейронна мережа

Вершини-кола зліва служать лише для розподілу вхідних сигналів. Вони не виконують будь-яких обчислень, і тому не будуть вважатися шаром. З цієї причини вони позначені колами, щоб відрізняти їх від вычисляющих нейронів, позначених квадратами. Кожен елемент з множини входів Х окремим вагою з'єднаний з кожним штучним нейроном, а кожен нейрон видає зважену суму входів в мережу. У штучних та біологічних мережах багато сполуки можуть бути відсутніми, всі з'єднання показані в цілях спільності. Можуть мати місце також з'єднання між виходами і входами елементів у шарі.

Ваги всіх елементів матриці можна позначити через W. Матриця має m рядків і n стовпців, де m – число входів, а n – число нейронів. Наприклад, w2,3 – це вага, що зв'язує третій вхід з другим нейроном. Таким чином, обчислення вихідного вектора N, компонентами якого є виходи OUT нейронів, зводиться до матричному множенню N = XW, де N і Х – вектори-рядки.[8]

**1.3.2 Багатошарові штучні нейронні мережі**

Більш великі і складні нейронні мережі мають, як правило, і великими обчислювальними можливостями. Хоча створені мережі всіх конфігурацій, які тільки можна собі уявити, пошарова організація нейронів копіює шаруваті структури певних відділів мозку. Такі багатошарові мережі володіють великими можливостями, ніж одношарові.

Багатошарові мережі можуть утворюватися каскадами шарів. Вихід одного шару є входом для наступного шару. Подібна мережа показана на рис. 1.4 і знову зображена з усіма сполуками.

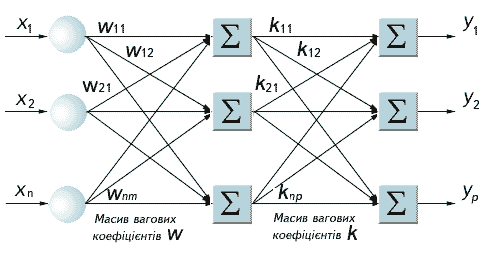


Рисунок 1.4 – Багатошарова нейронна мережа

Багатошарові мережі не можуть привести до збільшення обчислювальної потужності в порівнянні з одношарової мережею лише в тому випадку, якщо актіваціонная функція між шарами буде нелінійної. Обчислення виходу шару полягає в множенні вхідного вектора на першу вагову матрицю з подальшим множенням (якщо відсутня нелінійна актіваціонная функція) результуючого вектора на другу вагову матрицю.

Оскільки множення матриць асоціативно, то X(W1W2). Це показує, що двошарова лінійна мережа еквівалентна одному шару з ваговою матрицею, що дорівнює добутку двох вагових матриць. Отже, будь-яка багатошарова лінійна мережа може бути замінена еквівалентною одношарової мережею, однак, одношарові мережі досить обмежені за своїми обчислювальними можливостями. Таким чином, для розширення можливостей мереж порівняно з одношарової мережею необхідна нелінійна актіваціонная функція.[4]

**1.3.3 Нейронні мережі Хеба**

Мінський і Пейперт відзначали, що недоліки простих перцептронів можна подолати з допомогою багатошарових мереж, так і введенням в мережу зворотних зв'язків, що допускають циркуляцію сигналів по замкнутих контурах. Використовувати властивості такого роду мереж для моделювання функцій мозку ще в 1949 р. запропонував Хебб.

Згідно з поглядами Хеба нервові клітини мозку з'єднані один з одним великою кількістю прямих та зворотних зв'язків збуджуючих і утворюють нейронну мережу. Кожен нейрон здійснює просторово-тимчасову суммацию приходять до нього сигналів від збуджених нейронів, визначаючи потенціал на своїй мембрані. Коли потенціал на мембрані перевищує порогове значення, нейрон збуджується. Нейрон має рефрактерність і втомою. Ефективність зв'язків може змінюватися в процесі функціонування мережі, підвищуючись одночасно між збудженими нейронами. Це призводить до об'єднання нейронів у клітинні ансамблі – групи клітин, які найчастіше порушувалися разом, і до відокремлення ансамблів один від одного. При порушенні достатньої частини ансамблю він збуджується. Різні ансамблі можуть перетинатися: один і той самий нейрон може входити в різні ансамблі. Електрична активність мозку обумовлена послідовним збудженням окремих ансамблів.[16]

Ідеї Хеба справили великий вплив на уявлення про роботу мозку і послужили основою для створення нейронних моделей довготривалої пам'яті. Ансамблевую нейронну мережу можна розглядати як структуру, що реалізує функції розподіленої асоціативної пам'яті. Формування ансамблів в такій мережі відповідає запам'ятовування образів (ознак, об'єктів, подій, понять), закодованих паттерном поданням. Процес збудження всього ансамблю при активації частини його нейронів можна інтерпретувати як витяг запомненной інформації за її частині – ключа пам'яті. Модель пам'яті на основі ансамблевої нейронної мережі володіє деякими розгалуженість, паралельність, стійкість до шумів або збоїв і надійність.

Як ми бачили, вихід *NET* простого штучного нейрона є зваженою сумою його входів. Це може бути виражене таким чином:

де  вихід NET нейрона j; OUTi- вихід нейрона i; wij - вага зв'язку нейрона i з нейроном j.

Можна показати, що в цьому випадку лінійна багатошарова мережа не є потужнішою, ніж одношарова мережа; можливості розглянутої мережі можуть бути поліпшені тільки введенням нелінійності в передатну функцію нейрона. Кажуть, що мережа, що використовує сигмоїдальну функцію активації і метод навчання Хеба, навчається по сигнальному методу Хеба. У цьому випадку рівняння Хеба модифікується таким чином:

де *wij*(*t*) - сила синапсу від нейрона *i* до нейрона *j* в момент часу *t*;*OUTi* - вихідний рівень пресинаптичного нейрона рівний *F*(*NETi*); *OUTj* - вихідний рівень постсинаптичного нейрона рівний *F*(*NET*).

Метод сигнального навчання Хеба передбачає обчислення згортки попередніх змін виходів для визначення зміни ваги. Справжній метод, званий методом диференційного навчання Хеба, використовує наступну рівність:

wij(t+1) = wij(t) + [OUTi(t) - OUTi(t-1)][ OUTj(t) - OUTj(t-1)],

де *wij*(t)- сила синапсу від нейрона *i* донейрона *j* вмомент часу *t*;*OUTi*(*t*) -вихідний рівень пресинаптичного нейрона в момент часу *t*; *OUTj*(*t*) - вихідний рівень постсинаптичного нейрона в момент часу *t*.

Повний алгоритм навчання з застосуванням вищенаведених формул буде виглядати так:

1. На стадії ініціалізації всім ваговим коефіцієнтам присвоюються малі випадкові значення.

2. На входи мережі подається вхідний образ, і сигнали збудження поширюються по всім верствам згідно з принципами класичних прямопоточных (feedforward) мереж, тобто для кожного нейрона розраховується зважена сума його входів, до якого потім застосовується актіваціонная (передавальний) функція нейрона, в результаті чого виходить його вихідне значення yi(n), i=0...Mi-1, де Mi – число нейронів у шарі i; n=0...N-1, а N – число шарів у мережі.

3. На підставі отриманих вихідних значень нейронів за формулами

проводиться зміна вагових коефіцієнтів.

4. Цикл – перехід на крок 2, до тих пір, поки вихідні значення мережі не застабилизируются з заданою точністю.

Застосування цього нового способу визначення завершення навчання, відмінного від подстраиваемые значення синапсів фактично не обмежені.

На другому кроці циклу поперемінно пред'являються всі образи із вхідного набору.

Слід зазначити, що вид відгуків на кожен клас вхідних образів не відомий заздалегідь і буде представляти собою довільне поєднання станів нейронів вихідного шару, обумовлене випадковим розподілом ваг на стадії ініціалізації. Разом з тим, мережа здатна узагальнювати схожі образи, відносячи їх до одного класу. Тестування навченої мережі дозволяє визначити топологію класів у вихідному шарі. Для приведення відгуків навченої мережі до зручному поданням можна доповнити мережу одним шаром, який, наприклад, за алгоритмом навчання одношарового персептрона необхідно змусити відображати вихідні реакції мережі в необхідні образи.[7]

**1.3.4 Нейронні мережі Хопфілда**

Джон Хопфілд вперше представив свою асоціативну мережу у 1982 р. у Національній Академії Наук. На честь Хопфілда та нового підходу до моделювання, ця мережна парадигма згадується як мережа Хопфілда. Мережа базується на аналогії фізики динамічних систем. Початкові застосування для цього виду мережі включали асоціативну, або адресовану за змістом пам'ять та вирішували задачі оптимізації.

Мережа Хопфілда використовує три прошарки: вхідний, прошарок Хопфілда та вихідний прошарок. Кожен прошарок має однакову кількість нейронів. Входи прошарку Хопфілда під'єднані до виходів відповідних нейронів вхідного прошарку через змінні ваги з'єднань. Виходи прошарку Хопфілда під'єднуються до входів всіх нейронів прошарку Хопфілда, за винятком самого себе, а також до відповідних елементів у вихідному прошарку. В режимі функціонування, мережа скеровує дані з вхідного прошарку через фіксовані ваги з'єднань до прошарку Хопфілда. Прошарок Хопфілда коливається, поки не буде завершена певна кількість циклів, і біжучий стан прошарку передається на вихідний прошарок. Цей стан відповідає образу, вже запрограмованому у мережу.[12]

Навчання мережі Хопфілда вимагає, щоб навчальний образ був представлений на вхідному та вихідному прошарках одночасно. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань. Недвійкова реалізація мережі повинна мати пороговий механізм у передатній функції. Для правильного навчання мережі відповідні пари "вхід-вихід" мають відрізнятися між собою.

Якщо мережа Хопфілда використовується як пам'ять, що адресується за змістом вона має два головних обмеження. По-перше, число образів, що можуть бути збережені та точно відтворені є строго обмеженим. Якщо зберігається занадто багато параметрів, мережа може збігатись до нового неіснуючого образу, відмінному від всіх запрограмованих образів, або не збігатись взагалі. Межа ємності пам'яті для мережі приблизно 15% від числа нейронів у прошарку Хопфілда. Другим обмеженням парадигми є те, що прошарок Хопфілда може стати нестабільним, якщо навчальні приклади є занадто подібними. Зразок образу вважається нестабільним, якщо він застосовується за нульовий час і мережа збігається до деякого іншого образу з навчальної множини. Ця проблема може бути вирішена вибором навчальних прикладів більш ортогональних між собою.[14]

Структурна схема мережі Хопфилда приведена на рис. 1.5

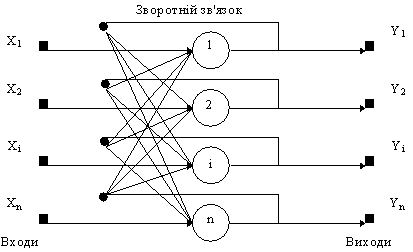


Рисунок 1.5 – Структурна схема мережі Хопфілда

Алгоритм функціонування мережі наступний:

1. На стадії ініціалізації мережі вагові коефіцієнти синапсів встановлюються таким чином:

Тут *i* і *j* - індекси, відповідно, предсинаптичного і постсинаптичного нейронів; *xik*, *xjk* - *i*-ий і *j*-ий елементи вектора *k*-ого зразка.

1. На входи мережі подається невідомий сигнал (t - номер ітерації). Його поширення безпосередньо встановлює значення виходів:

yi(0) = xi, i = 0...n-1,

тому позначення на схемі мережі вхідних сигналів у явному виді носить чисто умовний характер. Нуль у скобці справа від *yi* означає нульову ітерацію в циклі роботи мережі.

1. Розраховується новий стан нейронів

і нові значення виходів

де *f* - передатна функція у виді порогової.

4.Перевіряємо чи змінилися вихідні значення виходів за останню ітерацію. Якщо так - перехід до пункту 2, інакше (якщо виходи стабілізувались) - кінець. При цьому вихідний вектор являє собою зразок, що найкраще відповідає вхідним даним.

Іноді мережа не може провести розпізнавання і видає на виході неіснуючий образ. Це пов'язано з проблемою обмеженості можливостей мережі. Для мережі Хопфилда число запам'ятованих образів *m* не повинно перевищувати величини, приблизно рівної 0.15•*n*. Крім того, якщо два образи А и Б сильно схожі, вони, можливо, будуть викликати в мережі перехресні асоціації, тобто пред'явлення на входи мережі вектора А призведе до появи на її виходах вектори Б и навпаки.

**1.3.5 Нейронні мережі Хемінга**

Мережа Хемінга є розширенням мережі Хопфілда. Ця мережа була розроблена Річардом Ліппманом у середині 80-х рр. Мережа Хемінга реалізує класифікатор, що базується на найменшій похибці для векторів двійкових входів, де похибка визначається відстанню Хемінга. Відстань Хемінга визначається як число бітів, які відрізняються між двома відповідними вхідними векторами фіксованої довжини. Один вхідний вектор є незашумленим прикладом образу, інший є спотвореним образом. Вектор виходів навчальної множини є вектором класів, до яких належать образи. У режимі навчання вхідні вектори розподіляються до категорій для яких відстань між зразковими вхідними векторами та біжучим вхідним вектором є мінімальною.

Мережа Хемінга має три прошарки: вхідний прошарок з кількістю вузлів, скільки є окремих двійкових ознак; прошарок категорій (прошарок Хопфілда), з кількістю вузлів, скільки є категорій або класів; вихідний прошарок, який відповідає числу вузлів у прошарку категорій.[8]

Мережа є простою архітектурою прямого поширення з вхідним рівнем, повністю під'єднаним до прошарку категорій. Кожен елемент обробки у прошарку категорій є зворотно під'єднаним до кожного нейрона у тому ж самому прошарку і прямо під'єднаним до вихідного нейрону. Вихід з прошарку категорій до вихідного прошарку формується через конкуренцію.

Навчання мережі Хемінга є подібним до методології Хопфілда. На вхідний прошарок надходить бажаний навчальний образ, а на виході вихідного прошарку надходить значення бажаного класу, до якого належить вектор. Вихід містить лише значення класу до якої належить вхідний вектор. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань.

Структурна схема мережі Хопфилда приведена на рис. 1.6

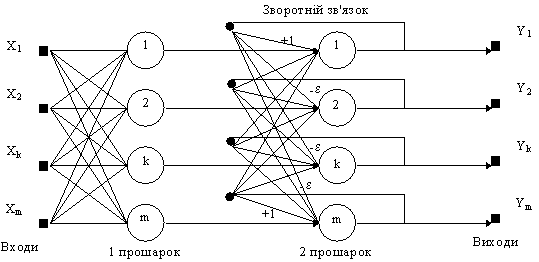


Рисунок 1.6 – Структурна схема мережі Хемінга

Алгоритм функціонування мережі Хемінга:

1. На стадії ініціалізації ваговим коефіцієнтам першого прошарку і порогу передатної функції присвоюються такі значення:

Wik=xIk/2, i=0...n-1, k=0...m-1

bk = n / 2, k = 0...m-1

Тут *xik* - *i*-ий елемент *k*-ого зразка.

Вагові коефіцієнти гальмуючих синапсів у другому прошарку беруть рівними деякій величині 0 < *v* < 1/m. Синапс нейрона, пов'язаний із його ж виходом має вагу +1.

1. На входи мережі подається невідомий вектор *x*1*, xi*, *xn* *...*Розраховуються стани нейронів першого прошарку (верхній індекс у скобках указує номер прошарку):

Після цього отримання значення ініціалізують значення виходів другого прошарку:

yj(2) = yj(1), j = 0...m-1

1. Обчислюються нові стани нейронів другого прошарку:

і значення їх виходів:

Передатна функція *f* має вид порога, причому величина *b* повинна бути достатньо великою, щоб будь-які можливі значення аргументу не призводили до насичення.

1. Перевіряється, чи змінилися виходи нейронів другого прошарку за останню ітерацію. Якщо так - перейти до кроку 3. Інакше - кінець.

Роль першого прошарку є умовною: скориставшись один раз на першому кроці 1 значеннями його вагових коефіцієнтів, мережа більше не вертається до нього, тому перший прошарок може бути взагалі виключений із мережі.

Мережа Хемінга має ряд переваг над мережею Хопфілда. Вона реалізує оптимальний класифікатор мінімуму похибки, якщо похибки вхідних бітів є випадковими та незалежними. Для функціонування мережі Хемінга потрібна менша кількість нейронів, оскільки середній прошарок вимагає лише один нейрон на клас, замість нейрону на кожен вхідний вузол. І, нарешті, мережа Хемінга не страждає від неправильних класифікацій, які можуть трапитись у мережі Хопфілда. В цілому, мережа Хемінга є як швидшою, так і точнішою за мережу Хопфілда.[12]

**1.3.6 Мережа Кохонена**

Мережа розроблена Тойво Кохоненом на початку 1980-х рр. і принципово відрізняється від розглянутих вище мереж, оскільки використовує неконтрольоване навчання і навчальна множина складається лише із значень вхідних змінних.

Мережа розпізнає кластери в навчальних даних і розподіляє дані до відповідних кластерів. Якщо в наступному мережа зустрічається з набором даних, несхожим ні з одним із відомих зразків, вона відносить його до нового кластеру. Якщо в даних містяться мітки класів, то мережа спроможна вирішувати задачі класифікації. Мережі Кохонена можна використовувати і в задачах, де класи відомі - перевага буде у спроможності мережі виявляти подібність між різноманітними класами.[3]

Мережа Кохонена має всього два прошарки: вхідний і вихідний, що називають самоорганізованою картою. Елементи карти розташовуються в деякому просторі - як правило двовимірному, що наведено на рис. 1.7.

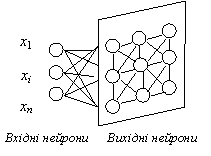


Рисунок 1.7 – Мережа Кохонена

Мережа Кохонена навчається методом послідовних наближень. Починаючи з випадковим чином обраного вихідного розташування центрів, алгоритм поступово покращується для кластеризації навчальних даних.

Проте, алгоритм може працювати і на іншому рівні. В результаті ітеративної процедури навчання мережа організовується таким чином, що елементи, які відповідають центрам, розташованим близько один від одного в просторі входів, будуть розташовані близько один від одного і на топологічній карті. Топологічний прошарок мережі можна уявити як двовимірну штахету, яку потрібно так відобразити в *N*-вимірний простір входів, щоб по можливості зберегти вихідну структуру даних. Звісно ж, при будь-якій спробі відтворити *N*-вимірний простір на площині буде загублено багато деталей, але такий прийом дозволяє користувачу візуалізувати дані, що неможливо зрозуміти іншим засобом.

Основний ітераційний алгоритм Кохонена послідовно проходить ряд епох, на кожній епосі опрацьовується один навчальний приклад. Вхідні сигнали - вектори дійсних чисел - послідовно пред'являються мережі. Бажані вихідні сигнали не визначаються. Після пред'явлення достатнього числа вхідних векторів, синаптичні ваги мережі визначають кластери. Крім того, ваги організуються так, що топологічне близькі вузли чуттєві до схожих вхідних сигналів.[3]

Алгоритм функціонування мережі Кохонена:

1. Ініціалізація мережі. Ваговим коефіцієнтам мережі надаються малі випадкові значення.
2. Пред'явлення мережі нового вхідного сигналу.
3. Обчислення відстані до всіх нейронів мережі:

Відстані *dj* від вхідного сигналу до кожного нейрона *j* визначаються за формулою:

де *xi* - *i-ий* елемент вхідного сигналу в момент часу *t*, *wij*(*t*) - вага зв'язку від *i-го* елемента вхідного сигналу до нейрона *j* у момент часу *t*.

1. Вибір нейрона з найменшою відстанню:

Вибирається нейрон-переможець *j\**, для якого відстань *dj* найменше.

1. Налаштування ваг нейрона *j\** і його сусідів:

Робиться налаштування ваг для нейрона *j\** і всіх нейронів з його околу NE. Нові значення ваг:

wij(t+1)=wij(t)+r(t)(xi(t)-wij(t))

де *r*(*t*) - швидкість навчання, що зменшується з часом (додатне число, менше одиниці).

1. Повернення до кроку 2.

**РОЗДІЛ 2**

**РОЗПІЗНАВАННЯ ОБРАЗІВ**

**2.1 Поняття образу**

Образ, клас – класифікаційне угруповання в системі класифікації, що об'єднує певну групу об'єктів за певною ознакою. Образне сприйняття світу – одне з властивостей живого мозку, що дозволяє розібратися в нескінченному потоці сприйманої інформації і зберігати орієнтацію в розрізнених даних про зовнішній світ. Сприймаючи зовнішній світ, ми завжди проводимо класифікацію інформації, тобто розбиваємо їх на групи схожих, але не тотожних явищ.

Наприклад, незважаючи на істотну відмінність, до однієї групи відносяться всі букви "А", написані різними почерками, або всі звуки, що відповідають одній і тій же ноті, взятої в будь октаві і на будь-якому інструменті. Для складання поняття про групи сприйняттів досить ознайомитися з незначною кількістю її представників. Це властивість мозку дозволяє сформулювати таке поняття, як образ.[6]

Образи володіють характерним властивістю, що виявляється в тому, що ознайомлення з кінцевим числом явищ з одного і того ж безлічі дає можливість дізнаватися як завгодно велике число його представників. Образи володіють характерними об'єктивними властивостями в тому сенсі, що різні люди, які навчаються на різному матеріалі спостережень, здебільшого однаково і незалежно один від одного класифікують одні і ті ж об'єкти. Саме ця об'єктивність образів дозволяє людям усього світу розуміти один одного.

Здатність сприйняття зовнішнього світу у формі образів дозволяє з певною вірогідністю дізнаватися нескінченне число об'єктів на підставі ознайомлення з їх кінцевим числом, а об'єктивний характер основного властивості образів дозволяє моделювати процес їх розпізнавання.[13]

**2.2 Проблема розпізнавання образів**

Розпізнавання образів – це завдання ідентифікації об'єкта або визначення будь-яких його властивостей за його зображенням (оптичне розпізнавання) або аудіозаписи (акустичне розпізнавання). В процесі біологічної еволюції багато тварин за допомогою зорового та слухового апарату вирішили цю задачу досить добре.

Створення штучних систем з функціями розпізнавання образів залишається складною технічною проблемою.

В цілому проблема розпізнавання образів (ПРО) складається з двох частин: навчання і розпізнавання. Навчання здійснюється шляхом відображення окремих об'єктів із зазначенням їх приналежності того чи іншого образу. В результаті навчання розпізнає система повинна набути здатність реагувати однаковими реакціями на всі об'єкти одного образу та іншими реакціями - на всі об'єкти відмітних образів. Дуже важливо, що процес навчання повинен завершитися тільки шляхом показів кінцевого числа об'єктів. В якості об'єктів навчання можуть бути або картинки, або інші візуальні зображення (літери, цифри). Важливо, що в процесі навчання вказуються тільки самі об'єкти та їх приналежність образу. За навчанням слід процес розпізнавання нових об'єктів, який характеризує дії вже навченої системи. Автоматизація цих процедур і становить проблему навчання розпізнаванню образів. У тому випадку, коли людина сама розгадує або вигадує, а потім нав'язує машині правило класифікації, проблема розпізнавання вирішується частково, так як основну і головну частину проблеми (навчання) людина бере на себе.[15]

Коло завдань, які можуть вирішуватися за допомогою розпізнавальних систем, надзвичайно широкий. Сюди відносяться не тільки задачі розпізнавання зорових і слухових образів, але і завдання класифікації складних процесів і явищ, що виникають, наприклад, при виборі доцільних дій керівником підприємства або виборі оптимального керування технологічними, економічними, транспортними або військовими завданнями.

Перш ніж розпочати аналіз будь-якого об'єкта, потрібно отримати про нього визначену упорядковану інформацію.

Вибір вихідного опису об'єктів є однією з центральних задач проблеми розпізнавання образів. При вдалому виборі вихідного опису (простору ознак) задача розпізнавання може виявитися тривіальною і, навпаки, невдало обраний початкове опис може призвести до дуже складної подальшу переробку інформації, або взагалі до відсутності рішення.[15]

**2.3 Геометричний та структурний підходи**

Будь-яке зображення, яке виникає в результаті спостереження якого-небудь об'єкта в процесі навчання або іспиту, можна представити у вигляді вектора, а значить і у вигляді точки деякого простору ознак. Якщо стверджується, що при показі зображень можливо однозначно віднести їх до одного з двох (або декількох) образів, то тим самим стверджується, що в деякому просторі існує дві (або декілька) області, що не мають загальних точок, і що зображення – точки з цих областей. Кожній такій області можна приписати найменування, тобто дати назву, що відповідає образу.

Проинтерпретируем тепер в термінах геометричної картини процес навчання розпізнаванню образів, обмежившись поки випадком розпізнавання тільки двох образів.[11]

Заздалегідь вважається відомим лише тільки те, що потрібно розділити дві області в деякому просторі, і що показуються точки тільки з цих областей. Самі ці галузі заздалегідь не визначені, тобто немає яких-небудь відомостей про розташування їх меж або правил визначення належності точки до тієї чи іншої області.

В ході навчання пред'являються точки, випадково вибрані з цих областей, і повідомляється інформація про те, до якої галузі належать пропоновані точки. Ніякої додаткової інформації про цих областях, тобто про розташування їх меж, в ході навчання не повідомляється. Мета навчання полягає або в побудові поверхні, яка розділяла б не тільки показані в процесі навчання точки, але і всі інші точки, що належать цим областям, або в побудові поверхонь, що обмежують ці області так, щоб у кожної з них знаходилися тільки точки одного образу. Мета навчання полягає в побудові таких функцій від векторів зображень, які були б, наприклад, позитивні на всіх точках одного та негативні на всіх точках іншого образу. У зв'язку з тим, що області не мають спільних точок, завжди існує ціла безліч таких поділяють функцій, а в результаті навчання повинна бути побудована одна з них.[10]

Якщо пред'являються зображення належать не двома, а великим числом образів, то завдання полягає в побудові за показаним в ході навчання точках поверхні, що розділяє всі області, відповідні цим образам, один від одного. Це завдання може бути вирішена, наприклад, шляхом побудови функції, приймаючої над точками кожної з областей однакове значення, а над точками з різних областей значення цієї функції повинно бути по-різному.

На перший погляд здається, що знання лише деякої кількості точок області недостатньо, щоб відокремити всю область. Дійсно, можна вказати незліченну кількість різних областей, які містять ці точки, і як би ні була побудована за ним поверхню, виділяє область, завжди можна вказати іншу область, яка перетинає поверхню і разом з тим містить показані точки. Однак відомо, що задача про наближення функції з інформації про неї в обмеженій множині точок, істотно більш вузькою, ніж всі безліч, на якому функція задана, є звичайною математичною задачею про апроксимації функцій. Рішення таких завдань вимагає введення певних обмежень на класі розглянутих функцій, а вибір цих обмежень залежить від характеру інформації, яку може додати вчитель у процесі навчання. Однією з таких підказок є гіпотеза про компактності образів.[11]

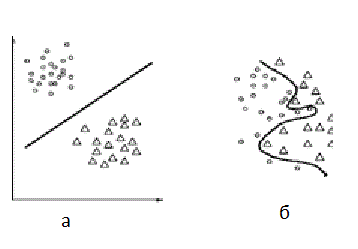


Рисунок 2.1 – Поділ двох класів у просторі

Апроксимація розділяє функції буде завданням тим легшою, ніж більш компактні і чим більше рознесені в просторі області, що підлягають розділенню. Так, наприклад, у випадку, показаному на рис. 2.1 а, поділ свідомо більш просто, ніж у випадку, показаному на рис. 2.1 б. Дійсно, у випадку, зображеному на рис. 2.1 а, області можуть бути розділені площиною, і навіть при великих погрішностей у визначенні розділяє функції вона все ж буде продовжувати розділяти області. У разі ж на рис. 2.1 б, поділ здійснюється хитромудрої поверхнею, і навіть незначні відхилення в її формі призводять до помилок поділу. Саме це інтуїтивне уявлення про порівняно легко розділимих областей призвело до гіпотези компактності.

Поряд з геометричною інтерпретацією проблеми навчання розпізнаванню образів існує й інший підхід, який названий структурним, або лінгвістичним.

Пояснимо лінгвістичний підхід на прикладі розпізнавання зорових зображень.

Спочатку виділяється набір вихідних понять – типових фрагментів, що зустрічаються на зображеннях, і характеристик взаємного розташування фрагментів – "ліворуч", "знизу", "всередині" і т. д. Ці вихідні поняття утворюють словник, що дозволяє будувати різні логічні висловлювання. Завдання полягає в тому, щоб з великої кількості висловлювань, які могли б бути побудовані з використанням цих понять, відібрати найбільш істотні для даного конкретного випадку.

Далі, переглядаючи кінцеве і по можливості невелике число об'єктів з кожного образу, потрібно побудувати опис цих образів. Зведені описи повинні бути настільки повними, щоб вирішити питання про те, до якого образа належить даний об'єкт. При реалізації лінгвістичного підходу виникають дві задачі: задача побудови вихідного словника, тобто набір типових фрагментів, і завдання побудови правил опису з елементів заданого словника.

В рамках лінгвістичної інтерпретації проводиться аналогія між структурою зображень і синтаксисом мови. Прагнення до цієї аналогії було викликано можливістю, використовувати апарат математичної лінгвістики, тобто методи по своїй природі є синтаксичними. Використання апарату математичної лінгвістики для опису структури зображень можна застосовувати тільки після того, як проведена сегментація зображень на складові частини, тобто вироблені слова для опису типових фрагментів і методи їх пошуку. Після попередньої роботи, що забезпечує виділення слів, виникають власне лінгвістичні завдання, що складаються із завдань автоматичного граматичного розбору описів для розпізнавання зображень. При цьому проявляється самостійна область досліджень, яка вимагає не тільки знання основ математичної лінгвістики, але й оволодіння прийомами, які розроблені спеціально для лінгвістичної обробки зображень.[17]

**2.4 Навчання, самонавчання і адаптація**

Навчання – це процес, в результаті якого система поступово набуває здатність відповідати потрібними реакціями на певні сукупності зовнішніх впливів, а адаптація – це підстроювання параметрів і структури системи з метою досягнення необхідної якості керування в умовах безперервних змін зовнішніх умов. Якщо б вдалося помітити якусь загальну властивість, не залежну ні від природи образів, ні від їх зображень, а визначає лише їх здатність до розділимості, то поряд із звичайним завданням навчання розпізнаванню з використанням інформації про належність кожного об'єкта з навчальної послідовності того чи іншого образу, можна було б поставити іншу класифікаційну задачу – так звану задачу навчання без вчителя.

Завдання такого роду на описовому рівні можна сформулювати наступним чином: системі одночасно або послідовно пред'являються об'єкти без будь-яких вказівок про їх приналежність до образів. Вхідний пристрій системи відображає безліч об'єктів на безліч зображень і, використовуючи деякий закладене в неї заздалегідь властивість разделимости образів, проводить самостійну класифікацію цих об'єктів. Після такого процесу самонавчання система повинна набути здатність до розпізнавання не тільки вже знайомих об'єктів (об'єктів з навчальної послідовності), але і тих, які раніше не демонструвалися. Процесом самонавчання деякої системи називається такий процес, в результаті якого ця система без підказки вчителя набуває здатність до вироблення однакових реакцій на зображення об'єктів одного і того ж образу і різних реакцій на зображення різних образів. Роль учителя при цьому складається лише в підказці системі деякого об'єктивного властивості, однакового для всіх образів і визначає здатність до розділення множини об'єктів на образи. Таким об'єктивним властивістю є властивість компактності образів. Взаємне розташування точок у вибраному просторі вже містить інформацію про те, як слід розділити множину точок. Ця інформація і визначає властивість разделимости образів, яке виявляється достатнім для самонавчання системи розпізнавання образів.[6]

Навчанням зазвичай називають процес вироблення в деякій системі тієї чи іншої реакції на групи зовнішніх ідентичних сигналів шляхом багаторазового впливу на систему зовнішньої коригування. Таку зовнішню коригування в навчанні прийнято називати "заохоченнями" і "покараннями". Механізм генерації цієї коригування практично повністю визначає алгоритм навчання. Самонавчання відрізняється від навчання тим, що тут додаткова інформація про вірність реакції системі не повідомляється.

Більшість відомих алгоритмів самонавчання здатні виділяти абстрактні образи, тобто компактні множини в заданих просторах. Відмінність між ними полягає у формалізації поняття компактності. Результат самонавчання характеризує придатність вибраного простору для конкретної задачі навчання розпізнаванню. Якщо абстрактні образи, які виділяються в процесі самонавчання, збігаються з реальними, то простір вибрано вдало. Чим сильніше абстрактні образи відрізняються від реальних, тим незручніше" вибране простір для конкретної задачі.[12]

Адаптація – це процес зміни параметрів і структури системи, а можливо, і керуючих впливів на основі поточної інформації з метою досягнення певного стану системи при початковій невизначеності і мінливих умовах роботи.

Можливий спосіб побудови розпізнають машин, заснований на розрізненні будь-яких ознак, що підлягають розпізнаванню фігур. В якості ознак можуть бути обрані різні особливості фігур, наприклад, їх геометричні властивості (характеристики складових фігури кривих), топологічні властивості (взаємне розташування елементів фігури) і т. п. Відомі розпізнавальні машини, в яких розрізнення букв або цифр проводиться, за так званим "методом зондів" (рис. 2.2), тобто за кількістю перетинів контуру фігури з кількома особливим чином розташованими прямими.



Рисунок 2.2 – Схема розташування зондів для розпізнавання цифр

Якщо проектувати цифри на поле з зондами, то виявиться, що кожна з цифр перетинає цілком певні зонди, причому комбінації пересікаються зондів різні для всіх десяти цифр. Ці комбінації і використовуються в якості ознак, за якими проводиться розрізнення цифр. Такі машини успішно справляються, наприклад, з читанням машинописного тексту, але їх можливості обмежені тим шрифтом (або групою подібних шрифтів), для якого була розроблена система ознак.[8]

Робота по створенню набору еталонних фігур або системи ознак повинна проводитися людиною. Якість роботи машини, тобто надійність "впізнавання" пред'являються фігур визначається якістю цієї попередньої підготовки і без участі людини не може бути підвищений.

**2.5 Алгоритмічні побудови**

Алгоритмічна універсальність ЕОМ означає, що на них можна програмно реалізовувати (тобто представити у вигляді машинної програми) будь-які алгоритми перетворення інформації, будь то обчислювальні алгоритми, алгоритми управління, пошуку докази теорем, тривимірні графічні або аудіо композиції.

Однак не слід думати, що обчислювальні машини і роботи можуть в принципі вирішувати будь-які завдання. Було строго доведено існування таких типів завдань, для яких неможливий єдиний і ефективний алгоритм, що вирішує всі задачі даного типу; у цьому сенсі неможливе вирішення таких завдань і з допомогою обчислювальних машин. Цей факт сприяє кращому розумінню того, що можуть робити машини і чого вони не можуть зробити.[4]

**2.5.1 Навчання штучних нейронних мереж**

Серед властивостей штучних нейронних мереж основним є їх здатність до навчання, вони навчаються найрізноманітнішими методами. Більшість методів навчання виходять із загальних передумов, і має багато ідентичних характеристик. Їх навчання нагадує процес інтелектуального розвитку людської особистості.

Можливості навчання штучних нейронних мереж обмежені. Тим не менш, вже отримані переконливі досягнення, такі як "мовець мережа" Сейновского, і виникає багато інших практичних застосувань [58]. Мережа навчається, щоб для деякої множини входів давати необхідну безліч виходів. Кожне таке вхідна (або вихідна) множина розглядається як вектор. Навчання здійснюється шляхом послідовного пред'явлення вхідних векторів з одночасною підстроюванням ваг згідно з певною процедурою. У процесі навчання ваги мережі поступово стають такими, щоб кожен вхідний вектор виробляв вихідний вектор.[10]

Навчальні алгоритми можуть бути класифіковані як алгоритми навчання з вчителем і без вчителя.

**2.5.2 Навчання з вчителем**

При навчанні з учителем існує вчитель, який пред'являє вхідні образи мережі, порівнює результуючі виходи з необхідними значеннями, а потім налаштовує ваги мережі таким чином, щоб зменшити відмінності. Навчання з вчителем припускає, що для кожного вхідного вектора існує цільовий вектор, що представляє собою необхідний вихід. Разом вони називаються навчальної парою. Зазвичай мережа навчається на деякому числі таких навчальних пар.

Пред'являється вихідний вектор, обчислюється вихід мережі і порівнюється з відповідним цільовим вектором, різниця (помилка) з допомогою зворотного зв'язку подається в мережу, і ваги змінюються у відповідності з алгоритмом, прагнуть мінімізувати помилку. Вектори навчальної множини пред'являються послідовно, обчислюються помилки і ваги підлаштовуються для кожного вектора доти, поки помилка по всьому навчального масиву не досягне прийнятно низького рівня.

**2.5.3 Навчання без вчителя**

Навчання з учителем критикувалося за свою біологічну неправдоподібність.

Важко уявити навчальний механізм в мозку, який би порівнював бажані і дійсні значення виходів, виконуючи корекцію за допомогою зворотного зв'язку.

Навчання без вчителя є набагато більш правдоподібною моделлю навчання в біологічній системі. Розвинена Кохоненом та іншими, вона не потребує цільовому векторі для виходів і, отже, не вимагає порівняння з зумовленими ідеальними відповідями. Навчальна множина складається лише з вхідних векторів. Навчальний алгоритм підлаштовує ваги мережі так, щоб виходили узгоджені вихідні вектори, тобто щоб пред'явлення досить близьких вхідних векторів давало однакові виходи. Процес навчання, отже, виділяє статистичні властивості навчальної множини і групує подібні вектори в класи. Пред'явлення на вхід вектора з даного класу дасть певний вихідний вектор, але до навчання неможливо передбачити, який вихід буде проводитися даним класом вхідних векторів. Отже, виходи подібної мережі повинні трансформуватися в якусь зрозумілу форму, обумовлену процесом навчання.[12]

**2.5.4 Процес навчання нейронних мереж**

Процес навчання Штучної Нейронної Мережі (ШНМ) нового класу задач включає наступні стадії:

1. Формулюється постановка задачі і виділяється набір ключових параметрів, які характеризують предметну область.

2. Вибирається парадигма нейронної мережі (модель, що включає в себе вигляд вхідних даних, порогової функції, структури мережі і алгоритмів навчання), найбільш підходяща для розв'язання даного класу задач. Як правило, сучасні нейропакеты, нейроплаты і нейрокомп'ютери дозволяють реалізувати не одну, а кілька базових парадигм.

3. Готується, можливо, більш широкий набір повчальних прикладів, організованих у вигляді наборів вхідних даних, асоційованих з відомими вихідними значеннями. Вхідні значення для навчання можуть бути неповні і частково суперечливі.

4. Вхідні дані по черзі пред'являються ШНМ, а отримане вихідне значення порівнюється з еталоном. Потім проводиться підстроювання вагових коефіцієнтів міжнейронних сполук для мінімізації помилки між реальним та бажаним виходом мережі.

5. Навчання повторюється до тих пір, поки сумарна помилка в усій множині вхідних значень не досягне прийнятного рівня, або ІНС не прийде в стаціонарний стан. Розглянутий метод навчання нейроподібної мережі носить назву «зворотне поширення помилки (error backpropagation) і відноситься до числа класичних алгоритмів нейроматематики.

Налаштована і навчена ШНМ може використовуватися на реальних вхідних даних, не тільки підказуючи користувачеві коректне рішення, але і оцінюючи ступінь його достовірності.[4]

**2.6 Алгоритм січних площин**

Дозволяють розпізнавати образи. Алгоритм навчання машини "впізнавання" образів, заснований на методі січних гіперплощин (рис. 2.3), гіперповерхні "шматками" гиперплощин і складається з наступних основних етапів:

А. Навчання (формування розділяє поверхні):

• Проведення січних площин;

• Виключення зайвих площин;

• Виключення зайвих шматків площин.

Б. Розпізнавання нових об'єктів.

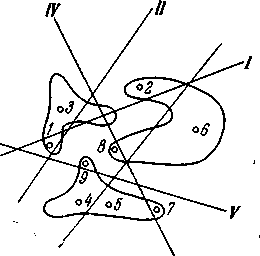


Рисунок 2.3 – Метод січних площин

При використанні методу паралельних варіантів одночасно і незалежно один від одного на одному і тому ж матеріалі навчаються кілька машин. При опознавании нових об'єктів кожна машина буде відносити ці об'єкти до якогось образу, може бути, не до одного і того ж. Остаточне рішення приймається голосуванням" машин – об'єкт ставитися до того образу, до якого його віднесло більше число машин.

Спосіб підвищення надійності розпізнавання полягає в деякому поліпшенні методу січних площин. Можна припустити, що якщо проводити січні площини близько до площини, що проходить через середину прямої, що з'єднує об'єкт і опонент, перпендикулярний цій прямій, то результуюча поверхня буде ближче до істинної кордоні між образами. Експерименти підтверджують це припущення.

**2.7 Алгоритми, засновані на методі потенціалів**

В алгоритмі, заснованому на методі потенціалів, з кожним порушеною елементом поля рецепторів можна зв'язати деяку функцію, рівну одиниці на цьому елементі і убуваючу за всіма напрямками від нього, тобто функцію, аналогічну електричного потенціалу з тією лише різницею, що в даному випадку R є відстань між двома сусідніми елементами поля рецепторів.

Для підрахунку користуються наступним простим правилом: кожний збуджений елемент поля рецепторів має "власний потенціал, рівний одиниці, і який в свою чергу збільшує на потенціали всіх (в тому числі і збуджених) сусідніх із ним елементів по горизонталі, вертикалі і діагоналях. Однак цей метод кодування може бути поліпшений. Якщо пов'язати з кожним порушеною елементом поля рецепторів деяку функцію, рівну одиниці на цьому елементі і убуваючу за всіма напрямками від нього, тобто функцію, аналогічну потенціалу, з тією лише різницею, що в даному випадку R є відстань між двома сусідніми елементами поля рецепторів. Ця функція, може бути апроксимована ступеневою функцією, постійною в межах одного рецептора і стрибкоподібно змінюється на кордонах рецепторів.[17]

Найпростіший алгоритм впізнавання, побудований на методі потенціалів, можна здійснити в два етапи:

1. Навчання (у процесі навчання запам'ятовуються коди всіх з'явилися точок і вказівки, до якого з образів відноситься кожна точки).

2. Пізнавання (у процесі впізнавання здійснюється ідентифікація і видається інформація, до якого образа належить закодована матриця).

**2.8 Приклад алгоритму навчання Персептрона з нерекуррентной мережею**

Незважаючи на деякі обмеження вихідної форми моделі Персептрона Розенблатта, вона стала основою для багатьох сучасних найбільш складних алгоритмів навчання з учителем. Вона використовує алгоритм навчання з вчителем; іншими словами, навчальна вибірка складається з безлічі вхідних векторів, для кожного з яких зазначено свій необхідний вектор мети. Компоненти вхідного вектора представлені безперервним діапазоном значень; компоненти вектора мети є двійковими величинами (0 або 1). Після навчання мережа отримує на вході набір безперервних входів і виробляє необхідний вихід у вигляді вектора з бінарних компонентів.

Навчання здійснюється наступним чином:

1. Рандомизируются всі ваги мережі в малі величини.

2. На вхід мережі подається вхідний навчальний вектор Х і обчислюється сигнал NET від кожного нейрона, використовуючи стандартний вираз

3. Обчислюється значення порогової функції активації для сигналу NET від кожного нейрона наступним чином:

OUTj = 1, якщо NETj більше ніж порог θj,

OUTj = 0 в іншому випадку.

Тут j являє собою поріг, відповідний нейрона j (в найпростішому випадку, всі нейрони мають один і той же поріг).

4. Обчислюється помилка для кожного нейрона за допомогою віднімання отриманого виходу з необхідного виходу:

errorj = targetj - OUTj.

5. Кожен вага модифікується таким чином:

Wij(t+1) = wij(t) +axierrorj.

6. Повторюються кроки з другого по п'ятий до тих пір, поки помилка не стане достатньо малою.

**2.9 Метод навчання Уидроу-Хоффа**

Персептрон Розенблатта обмежується бінарними виходами. Уидроу разом з Хоффом розширили алгоритм навчання персептрона на випадок безперервних виходів, використовуючи сигмоидальную функцію. Вони розробили математичне доказ того, що мережа при певних умовах буде сходитися до будь-якої функції, яку вона може уявити. Їх перша модель – Адалин має один вихідний нейрон, більш пізня модель – Мадалин розширює її на випадок з багатьма вихідними нейронами.[7]

Істотні відмінності є у четвертому кроці, де використовуються безперервні сигнали NET замість бінарних OUT. Модифікований крок 4 в цьому випадку реалізується наступним чином:

4. Обчислюється помилка для кожного нейрона за допомогою віднімання отриманого виходу з необхідного виходу:

**2.10 Нейро-мережева самоорганізація**

Результати досліджень Кохонена на самоорганізуються структурах, для задач розпізнавання образів класифікують образи, представлені векторними величинами, в яких кожен компонент вектора відповідає елементу образу. Алгоритми Кохонена грунтуються на техніці навчання без вчителя. Після навчання подача вхідного вектора з даного класу буде приводити до вироблення збудливого рівня в кожному вихідному нейроні; нейрон з максимальною порушенням представляє класифікацію. Так як навчання проводиться без зазначення цільового вектора, то немає можливості заздалегідь визначати, який нейрон буде відповідати даного класу вхідних векторів. Тим не менш, це легко планування проводиться шляхом тестування мережі після навчання.[11]

Алгоритм трактує набір з n вхідних ваг нейрона як вектор в n-мірному просторі. Перед навчанням кожен компонент цього вектора ваг ініціалізується у випадкову величину. Потім кожен вектор нормалізується вектор з одиничною довжиною в просторі ваг. Це робиться діленням кожного випадкового ваги на квадратний корінь із суми квадратів компонент цього вагового вектора.

Всі вхідні вектора навчального набору також нормалізуються, і мережа навчається згідно з наступним алгоритмом:

1. Вектор Х подається на вхід мережі.

2. Визначаються відстані Dj (в n-мірному просторі) між Х і ваговими векторами Wj кожного нейрона. В евклідовому просторі це відстань обчислюється за такою формулою

де хі - компонента i вхідного вектора X, wij - вага входу нейрона i j.

3. Нейрон, який має ваговий вектор, найближчий до X, оголошується переможцем. Цей ваговий вектор, званий Wc, стає основним у групі вагових векторів, які лежать у межах відстані D від Wc.

4. Група вагових векторів настроюється у відповідності з наступним вираженням:

Wj(t+l) = Wj(t) + a[X - w j(t)]

для всіх вагових векторів у межах відстані D від Wc

5. Повторюються кроки з 1 по 4 для кожного вхідного вектора.

В процесі навчання нейронної мережі значення D і поступово зменшуються.

Коефіцієнт на початку навчання лучі встановлювати приблизно рівним 1 і зменшувався в процесі навчання до 0, в той час як D може на початку навчання дорівнювати найбільшій відстані між ваговими векторами і в кінці навчання стати настільки маленьким, що буде навчатися лише один нейрон.

У відповідності з існуючою точкою зору, точність класифікації буде поліпшуватися при додатковому навчанні. Згідно рекомендації Кохонена, для отримання хорошої статистичної точності кількість навчальних циклів повинно бути, принаймні, в 500 разів більше кількості вихідних нейронів.[3]

Навчальний алгоритм налаштовує вагові вектори в околиці порушеної нейрона таким чином, щоб вони були більш схожими на вхідний вектор. Так як всі вектори нормалізуються в вектори з одиничною завдовжки, вони можуть розглядатись як точки на поверхні одиничної гиперсферы. У процесі навчання група сусідніх вагових точок переміщається ближче до точки вхідного вектора. Передбачається, що вхідні вектори фактично групуються в класи згідно з їх положенням у векторному просторі. Певний клас буде асоціюватися з певним нейроном, переміщаючи його ваговий вектор в напрямку центру класу і сприяючи його порушення при появі на вході будь-якого вектора даного класу. Після навчання класифікація виконується за допомогою подачі на вхід мережі випробуваного вектора, обчислення збудження для кожного нейрона з подальшим вибором нейрона з найвищим збудженням як індикатора правильної класифікації.

**РОЗДІЛ 3**

**ПРОГРАМНА РЕАЛІЗАЦІЯ**

**3.1 Постановка задачі**

1. Обрати програмне забезпечення для реалізації програми розпізнавання;

2. Розробити програму для розпізнавання образів за допомогою нейронних мереж;

3. Використовуючи веб-сервіс GitHub реалізувати програму онлайн у вільному доступі;

4. Протестувати роботу програми.

**3.2 Програмне забезпечення**

Основна ідея полягає в створенні безкоштовного онлайн додатку у вільному доступі для всіх бажаючих.

Було обрано веб-сервіс GitHub, як безкоштовний хостинг на якому реалізовано додаток у вільному доступі.

В якості мови програмування було використано Java Script, CSS, HTML.

**3.3 Робота програми**

Функція кореляції дає змогу програмі розпізнавати нечіткі розмиті зображення:

|  |
| --- |
| function correlation(xArray, yArray) { |
|  | var altx, alty, i, k; |
|  | altx = alt(xArray); |
|  | alty = alt(yArray); |
|  |  |
|  | if ((summ(multArray(altx, altx)) === 0 && summ(multArray(alty, alty)) === 0) || summ(multArray(altx, alty)) === 0) |
|  | return 1; |
|  | return summ(multArray(altx, alty)) / Math.sqrt(summ(multArray(altx, altx)) \* summ(multArray(alty, alty))); |
|  | }; |

В залежності від якості зображення користувач може встановлювати коефіцієнт наближення розпізнавання від 0,5 до 1 (рис. 3.1):



Рисунок 3.1 – Вибір коефіцієнту розпізнавання

Даний додаток можна використовувати для будь-яких типів розпізнавання, застосовувати для розпізнавання обличь, автомобільних номерних знаків, тексту, цифр і т.д.

Приклад розпізнавання обличчя показано на рис. 3.2:

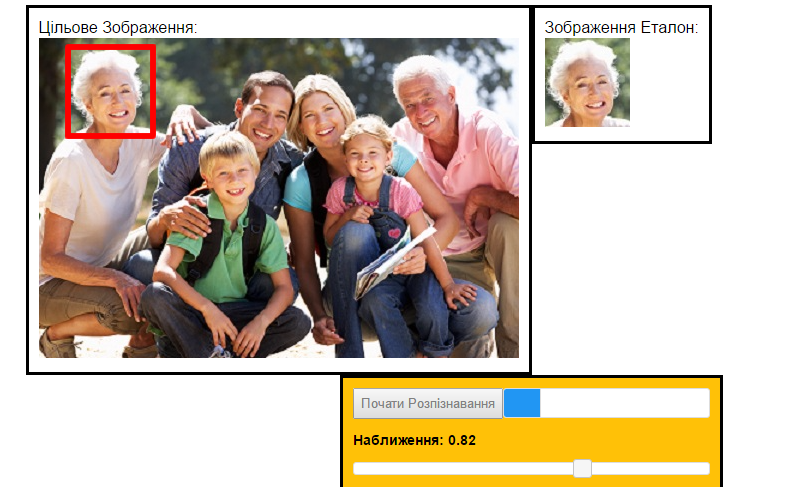


Рисунок 3.2 – Розпізнавання обличчя

Приклад розпізнавання з нечітким зображенням показано на рис. 3.3:

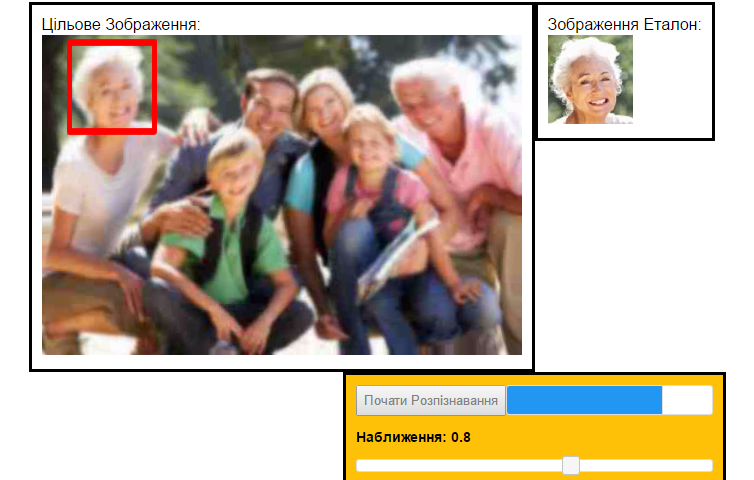


Рисунок 3.3 – Розпізнавання нечіткого забраження

Приклад невдалого розпізнавання показано на рис. 3.4:

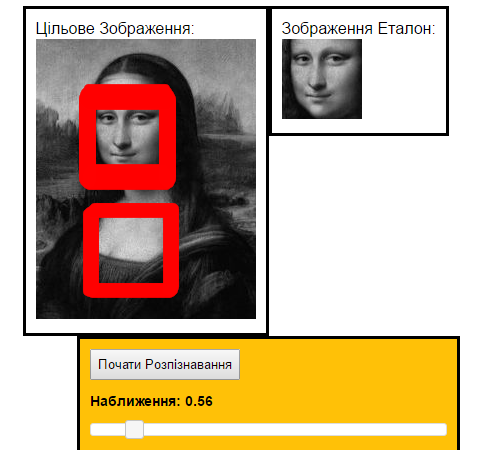


Рисунок 3.4 - Невдале розпізнавання

**РОЗДІЛ 4**

**ОХОРОНА ПРАЦІ**

**4.1 Шкідливі речовини в повітрі робочої зони**

Оточуюче нас повітря (атмосфера) є найважливішим фактором забезпечення нашого життя. Без повітря, що потрапляє через дихальні шляхи в легені, вже через декілька хвилин настає смерть. В природних умовах повітря, як правило, не забруднене отруйними речовинами і життю людини не загрожує. Тільки з того часу, коли людина почала використовувати в своїй діяльності шкідливі для її організму речовини, з'явилася загроза її життю. При цьому з'ясувалось, що наші органи чутливості не дозволяють з достатньою точністю визначати якість повітря і запобігати загрозі отруєння.

Наше відчуття на нюх нездатне сигналізувати про наявність у повітрі деяких шкідливих речовин, наприклад, оксиду та діоксиду вуглецю, оксидів азоту та інших речовин. В той же час, коли ми і відчуваємо присутність у повітрі незначної кількості отруйних речовин (таких, як синільна кислота), наш організм не відповідає на це захисною реакцією. Реакція організму настає з запізненням, коли отрута вже накопичилась в організмі в значній кількості і стала небезпечною для життя. Ступінь отруєння залежить як від кількості отрути, що потрапила в організм, так і від індивідуальної чутливості організму людини до дії конкретної шкідливої речовини.[21]

Зважаючи на викладене, можна констатувати, що для створення здорових і безпечних умов праці потрібно мати гігієнічне нормування шкідливих речовин, надійні способи визначення їх концентрацій у повітрі і сучасне технічне та організаційне забезпечення їх знешкодження.

У чистому повітрі є шкідливі гази, такі, як оксид вуглецю, озон, водень, оксид та діоксид азоту та деякі інші, які не позначаються негативно на здоров'ї людей, тварин та всієї флори і фауни Землі через незначну їх концентрацію.

Концентрації забруднюючих речовин наводяться та розраховуються в одиницях маси, яка міститься в одиниці об'єму повітря (), або у вигляді об'ємного співвідношення газів: 1 частка (об.)/10 часток (об.)=млн\*1.

Ці гази потрапляють у повітря завдяки існуванню вільного озону 03 в поверхневих шарах атмосфери, а також процесам гниття та розкладання () або атмосферними явищами ().

Чистим вважається повітря, не забруднене твердими, рідкими та газоподібними речовинами і газами, які змінюють його природний склад.

Тверді, рідкі або газоподібні речовини будь-якого ряду і походження, що потрапляють у повітря і змінюють його природний склад, називають емісіями. Існує ще поняття іммісія - це забруднюючі атмосферне повітря речовини, що присутні в атмосфері в безпосередній близькості від зони своєї дії, як правило, на висоті 1,5 км від поверхні землі або верхньої межі рослинності, або на відстані 1,5 км від поверхні будівлі.[19]

Емісії – це забруднення техногенного походження. В технічній літературі користуються поняттям "забруднення", "шкідливі речовини" в тих випадках, коли ці речовини присутні у повітрі в концентраціях, шкідливих і небезпечних для флори та фауни Землі.

Всесвітня організація охорони здоров'я (ВООЗ) дає таке визначення: "Забруднення повітря має місце в такому випадку, коли забруднююча повітря речовина або декілька речовин присутні в атмосфері в такій кількості і протягом такого часу, що спричиняють шкоду або можуть сприяти шкоді людям, тваринам, рослинам та майну, або можуть призвести до погіршення здоров'я людини або стану майна, які не піддаються обліку".

**4.2 Розрахунок штучного освітлення робочого приміщення**

Штучне освітлення – здійснюється штучними джерелами світла (лампами розжарювання) і призначене для освітлення приміщень у темні години доби, або таких приміщень, які не мають природного освітлення.

У якості джерел штучного освітлення широко використовують лампи розжарювання, газорозрядні лампи та світодіодні. Штучне освітлення може бути загальним та комбінованим.

Загальним називаються освітлення, при якому світильники розміщуються у верхній зоні приміщення не нижче 2,5 м над підлогою рівномірно (загальне рівномірне освітлення) або з врахуванням розташування робочих місць (загальне локалізоване освітлення).

Комбіноване освітлення складається із загального та місцевого. Його доцільно застосовувати при роботах високої точності, а також, якщо необхідно створити певний або змінний, в процесі роботи, напрямок світла. Місцеве освітлення створюється світильниками, що концентрують світловий потік безпосередньо па робочих місцях. Застосовування лише місцевого освітлення не допускається з огляду на небезпеку виробничого травматизму та професійних захворювань.

За функціональним призначенням штучне освітлення поділяється на робоче, аварійне, евакуаційне, охоронне, чергове.

Робоче освітлення призначене для забезпечення виробничого процесу, переміщення людей, руху транспорту і є обов'язковим для всіх виробничих приміщень.[19]

Аварійне освітлення використовується для продовження роботи у випадках, коли раптове відключення робочого освітлення, та пов'язане з ним порушення нормального обслуговування обладнання може викликати вибух, пожежу, отруєння людей порушення технологічного процесу.

Евакуаційне освітлення призначене для забезпечення евакуації людей з приміщень при аварійному відключенні робочого освітлення. Його необхідно влаштовувати в місцях, небезпечних для проходу людей; в приміщеннях допоміжних будівель, де можуть одночасно знаходитись 100 осіб; в проходах, на сходових клітках, у виробничих приміщеннях, в яких працює більше 50 працівників.

Охоронне освітлення влаштовується вздовж меж території, яка охороняється в нічний час спеціальним персоналом.

Чергове освітлення передбачається у неробочий час при цьому, як правило використовують частину світильників інших видів штучного освітлення.[21]

Розрахунок штучного освітлення робочого приміщення:

**Вихідні дані:**

довжина приміщення а=10м;

ширина приміщення b=4;

нормована освітленість – 200 лк;

коефіцієнт відбиття:

стелі P=70%;

стін Р=50%;

тип світильника з лампами розжарювання – “універсаль”;

висота підвісу світильників над робочою поверхнею h=2,6м.

Розрахувати систему загального рівномірного освітлення для виробничого приміщення з лампами розжарювання, встановленими в світильники (по 1 шт. в світильник);

Визначити загальну потужність електроенергії для живлення систем освітлення з лампами розжарювання.

**Виконання:**

Визначаємо орієнтовну кількість світильників:

Рівномірність освітлення досягається при співвідношенні відстані між світильниками L і висоти їх підвісу h:

L=h=2,6м

Необхідна кількість світильників становить:

Визначаємо показник приміщення:

Визначаємо мінімально необхідно величину світлового потоку лампи розжарювання при її використанні в світильнику типу “Універсаль”:

За табл. знаходимо коефіцієнт використання світлового потоку для світильника типу “Універсаль” при .

За табл. вибираємо лампу розжарювання потужністю – 500Вт з світловим потоком 8200 лм> 4746 лм.

Загальна потужність електроенергії, яка повинна підводитися до системи освітлення становитиме: 500 Вт \* 6=3000 Вт = 3 кВт.

**4.3 Електробезпека робочих місць**

З метою забезпечення електробезпеки на робочому місці всі металеві частини електроустановок, які знаходяться під напругою, повинні бути заземлені шляхом з'єднання з нульовим проводом мережі.

Заземленими мають бути: корпуси електрифікованих машин, електродвигуни, електрокалорифери, переносні електроприймачі, електроінструмент, металеві каркаси та кожухи розподільчих щитів та силових шаф, корпуси пускових і захисних апаратів, металева освітлювальна та опромінювальна апаратура, металева ізоляція кабелів, стальні труби для електропроводки та ін.[1]

Як природні заземлювачі використовуються прокладені в ґрунті водопровідні та інші металеві трубопроводи (окрім трубопроводів горючих рідин, горючих і вибухонебезпечних газів, а також трубопроводів, покритих ізоляцією для захисту від корозії), труби артезіанських свердловин, металеві конструкції та арматура залізобетонних будинків та споруд, які мають з'єднання з землею, свинцеві оболонки кабелів, прокладених у ґрунті.

Штучними заземлювачами можуть бути звичайні вертикальні і горизонтальні електроди зі сталевих труб або стержнів діаметром 30-50 мм, сталі кутового профілю розміром від 40x40 до 60x60 мм, довжиною 2,5 м, штабової сталі площею перерізу не менше 4x12 мм або круглого перерізу діаметром більше 6 мм.[18]

Згідно з ПУЕ в електроустановках напругою до 1000 В з глухо заземленою і ізольованою нейтраллю опір заземлюючого пристрою повинен бути не більш 10 Ом при потужності трансформаторів і генераторів менше 100 КВа і 4 Ом в інших випадках.

Згідно з ПТЕ (Е ІІ-13-22) для визначення технічного стану заземлюючого пристрою періодично проводять:

– зовнішній огляд видимої частини заземлюючого пристрою;

– перевірку цілісності мережі між корпусом електроустановки та заземлювачем;

– перевірку опору заземлюючого пристрою;

– перевірку надійності з'єднання природних заземлювачів;

– перевірку питомого опору ґрунту.

Заміри опору заземлювачів проводять не рідше одного разу на рік у періоди найменшої провідності ґрунту. Зовнішній огляд здійснюють не рідше одного разу на півроку, а вологих приміщень кожні три місяці. Результати вимірів оформляють актом, а результати огляду заносять в оперативний журнал.[1]

Ізоляція електропроводки вважається достатньою, якщо її опір між проводом кожної фази і землею або різними фазами на відрізку між послідовно увімкненими автоматами або плавкими запобіжниками чи за останнім запобіжником становить не менше 0,5 МОм (500000 Ом). У звичайних приміщеннях її заміряють не рідше одного разу в два роки, а у вологих, особливо вологих, пожежонебезпечних, вибухонебезпечних і в приміщеннях з хімічно активними парами, які псують ізоляцію, - щорічно. В проміжках часу між замірами (раз у 6 місяців або в 3 місяці залежно від приміщення) оглядають проводку, вимикачі, арматуру світильників.

Якщо опір буде нижчим норми, ізоляцію випробовують напругою 1000 В протягом 1 хв (можна подавати напругу від мегомметра 2500 В). При цьому, якщо ізоляція не пробиває, цей проміжок можна залишити в роботі до планової заміни.[18]

Можна вважати ізоляцію задовільною, якщо її опір в омах менше величини застосованої напруги, помноженої на 1000.

R<UB \*1000 (Ом)

**4.4 Пожежна безпека робочих приміщень**

Забезпечення пожежної безпеки є обов'язковою складовою виробничої та іншої діяльності посадових осіб, працівників підприємств, установ, організацій і підприємців. Органи державного пожежного нагляду контролюють стан пожежної безпеки, вдаючись до різних санкцій (відмова у виданні дозволу на початок роботи або оренду приміщень, штрафи, призупинення експлуатації приміщень, споруд, устаткування, об'єктів тощо).[20]

Відповідно до Правил пожежної безпеки в Україні до основних організаційних заходів із її забезпечення належать:

– визначення обов'язків посадових осіб щодо гарантування пожежної безпеки;

– призначення відповідальних за пожежну безпеку окремих будівель, споруд, приміщень, дільниць тощо, технологічного та інженерного устаткування, а також за утримання та експлуатацію наявних технічних засобів протипожежного захисту;

– встановлення на кожному підприємстві (установі, організації) відповідного протипожежного режиму;

– розроблення і затвердження загальнооб'єктової інструкції про заходи пожежної безпеки та відповідних інструкцій для всіх вибухопожежонебезпечних та пожежонебезпечних приміщень, організація вивчення цих інструкцій працівниками;

– розроблення планів (схем) евакуації людей на випадок пожежі;

– встановлення порядку (системи) оповіщення людей про пожежу, ознайомлення з ним усіх працівників;

– визначення категорій будівель та приміщень за вибухопожежною та пожежною небезпекою відповідно до вимог чинних нормативних документів, встановлення класів зон за Правилами улаштування електроустановок;

– забезпечення територій, будівель та приміщень відповідними знаками пожежної безпеки, табличками із зазначенням номера телефону та порядку виклику пожежної охорони;

– створення та організація роботи пожежно-технічних комісій, добровільних пожежних дружин і команд.[9]

Комплексна реалізація цих заходів дає змогу запровадити протипожежний режим.

Протипожежний режим – комплекс загальнообов'язкових норм поведінки, правил виконання робіт та експлуатації об'єкта (виробу), спрямованих на гарантування його пожежної безпеки.

Він є основою системи пожежної безпеки – комплексу організаційних заходів і технічних засобів, спрямованих на запобігання пожежі та збиткам від неї. Пожежобезпека об'єкта – стан об'єкта, за якого з регламентованою ймовірністю виключається можливість виникнення і розвитку пожежі, а також забезпечується захист матеріальних цінностей. Рівень забезпечення пожежної безпеки – кількісна оцінка попереджених збитків у разі можливої пожежі.

Аналіз пожежної небезпеки полягає у визначенні наявності горючих речовин і можливих джерел запалювання, ймовірних шляхів поширення пожежі та необхідних засобів пожежогасіння. [20]

Аналіз пожежної небезпеки будівель, приміщень, технологічного устаткування тощо повинен дати відповіді на питання: де, за яких умов і в який спосіб може виникнути пожежа, як вона розвиватиметься, до чого це призведе. Кінцевою метою такого аналізу має бути максимально можливе виключення потенційних джерел запалювання, зведення до мінімуму горючого середовища, встановлення такого рівня протипожежного режиму, за якого ймовірність виникнення пожежі та масштаби її наслідків будуть найменшими.

Методика аналізу пожежної небезпеки зводиться до виявлення та оцінювання:

– потенційних та наявних джерел запалювання;

– умов формування горючого середовища;

– умов виникнення контакту джерел запалювання та горючого середовища;

– умов і причин поширення вогню в разі виникнення пожежі;

– наявності та масштабів імовірної пожежі, загрози життю і здоров'ю людей, навколишньому середовищу та матеріальним цінностям;

– рівня працездатності систем протипожежного захисту та протипожежної стійкості кожної ділянки і об'єкта загалом;

– порушень протипожежного режиму, норм і правил пожежної безпеки.

Оскільки повністю виключити імовірність виникнення пожежі неможливо, то необхідно використовувати стратегію обмеження її наслідків, вдаючись до таких заходів:

– забезпечення вогнестійкості будівель та споруд;

– своєчасна евакуація людей та відповідність чинним нормам шляхів евакуації;

– створення умов для ефективного гасіння пожежі;

– обмеження поширення пожежі;

– ліквідація горіння.

Вказані заходи реалізують через систему забезпечення протипожежного захисту.[9]

**ВИСНОВКИ**

Випускна робота бакалавра, присвячена актуальній темі – розробці додатку розпізнавання образів.

Було проаналізовано основні методи та алгоритми нейронних мереж.

Розроблено додаток який забезпечує:

- розпізнавання образів;

- в основі покладено нейронну мережу;

- доступність ресурсу онлайн;

- адаптація програми до розпізнавання нечітких зображень.

До недоліків програми можна віднести невелику швидкість розпізнавання.

Далеко не всі задачі розпізнавання образів можуть бути вирішені нейронними мережами. А якщо і можуть, то з якими-небудь обмеженнями, що підтверджують результати роботи написаної мною програми розпізнавання. Незважаючи на те, що штучний нейрон є моделлю біологічного нейрона, він лише апроксимує його властивості, а не в точності повторює їх. Тому машина як і раніше не в змозі навчатися, як людина.

**СПИСОК ЛІТЕРАТУРИ**

1. Анофріков В. Є., Учеб. посібник для вузів / С.А. Бобок, М.М. Дудко, Г.Д. Єлістратов / ГУУ. М., ЗАТ «Финстатинформ», 1999.

2. Антонов А.С. Введение в параллельные вычисления. Методическое пособие / А.С. Антонов. - М.: Изд-во МГУ, 2002. - 70 с.

3. Буцев А.В., Первозванский А.А. Локальная аппроксимация на искусственных нейросетях / А.В. Буцев // Автоматика и Телемеханика 1995.

4. Воеводин В.В. Математические модели и методы в параллельных процессах / В.В. Воеводин. - М.: Наука, 1986.

5. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления / В.В. Воеводин, Вл.В. Воеводин. - СПб.ГУ, 1997. - 308 с.

6. Воронцов, К. В., Лекции по искусственным нейронным сетям // К.В. Воронцов. - Воронеж, 2007. - 29 с.

7. Галушкин А.И. Нейронные сети. Основы теории / А.И. Галушкин. - М.: Горячая линия - Телеком, 2012. - 496 с.

8. Галушкин А.И. Нейронные сети: история развития: учеб. Пособие для вузов / А.И. Галушкин, Я.З. Цыпкин. - М.: ИПРЖ, 2001. - 839 с.

9. Гандзюк, М. П. Основи охорони праці [Текст] : Підручник.3-є вид. / М. П. Гандзюк, Є. П. Желібо, М. О. Халімовський; За ред. М. П. Гандзюка. – К. : Каравела, 2006.

10. Головко В.А. Нейронные сети: обучение, организация и применение / В.А. Головко. - М.: ИПРЖР, 2001, 256 с.

11. Горбань А.Н. Обобщенная аппроксимационная теорема и вычислительные возможности нейронных сетей / А. Н. Горбань // Сибирский журнал вычислительной математики - 1998. - Т. 1, №1. - С. 12-24

12. Горбань А.Н. Обучение нейронных сетей / А.Н. Горбань. - М.: Параграф, 1990. - 160 с.

13. Городняя Л.В. Основы функционального программирования. Курс лекций / Л.В. Городняя. - М.: Интернет-университет информационных технологий, 2004. 280 с.

14. Дорогов А.Ю. Быстрые нейронные сети / А.Ю. Дорогов. - СПб.: СПУ, 2002.

15. Калинин А.В., Подвальный, С.Л. Технология нейросетевых распределенных вычислений / А.В. Калинин, С.Л. Подвальный. - Воронеж: ВГУ, 2004.

16. Каллан Р. Основные концепции нейронных сетей / Р. Каллан. - М.: Вильямс, 2001.

17. Круглов В.В. Нечеткая логика и искусственные нейронные сети // В.В. Круглов, М.И. Дли, Р.Ю. Голунов: Учеб. пособие. - М.: Физматлит, 2001.

18. Нікіфоров Л. Л., [Виробнича безпека](http://ua-referat.com/%D0%92%D0%B8%D1%80%D0%BE%D0%B1%D0%BD%D0%B8%D1%87%D0%B0_%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D0%BA%D0%B0): навч. посібник / В.В. Персияне. - М.: МГУПБ, 2006.

19. Платонов Б. П., Безпека життєдіяльності: навч. посібник.

20. Рожков, А. П. Пожежна безпека [Текст] : справ. изд. / А. П. Рожков – К. : Пожінформтехніка, 1999.

21. Ярошевська В. М. Охорона праці в галузі: Навчальний посібник/ В. М. Ярошевська, В. Й. Чабан; М-во науки і освіти України, Український держ. ун-т водного господарства та природокористування.- Київ: ВД "Професіонал, 2004.

**ДОДАТОК А**

**Розробка додатку**

**Correlation.js:**

|  |
| --- |
| function summ(array) { |
|  | var i, s; |
|  | s = 0; |
|  | for (i = 0; i < array.length; i++) |
|  | s += array[i]; |
|  | return s; |
|  | }; |
|  |  |
|  | function avg(array) { |
|  | return summ(array) / array.length; |
|  | }; |
|  |  |
|  | function alt(array) { |
|  | var b = [], |
|  | i, average; |
|  |  |
|  | average = avg(array); |
|  | for (i = 0; i < array.length; i++) |
|  | b.push(array[i] - average); |
|  | return b; |
|  | }; |
|  |  |
|  | function multArray(aArray, bArray) { |
|  | var cArray = [], |
|  | i; |
|  |  |
|  | for (i = 0; i < aArray.length; i++) |
|  | cArray.push(aArray[i] \* bArray[i]); |
|  | return cArray; |
|  | }; |
|  |  |
|  | function correlation(xArray, yArray) { |
|  | var altx, alty, i, k; |
|  | altx = alt(xArray); |
|  | alty = alt(yArray); |
|  |  |
|  | if ((summ(multArray(altx, altx)) === 0 && summ(multArray(alty, alty)) === 0) || summ(multArray(altx, alty)) === 0) |
|  | return 1; |
|  | return summ(multArray(altx, alty)) / Math.sqrt(summ(multArray(altx, altx)) \* summ(multArray(alty, alty))); |
|  | }; |
|  |  |
|  | function oneToOne(xArray, yArray) { |
|  | for (var i = 0; i < xArray.length; i++) |
|  | if (xArray[i] !== yArray[i]) |
|  | return false; |
|  | return true; |
|  | }; |

**Recognition.js:**

|  |
| --- |
| var Reco = (function() { |
|  |  |
|  | function imageToGrayArray(data) { |
|  | var fragmant = [], |
|  | i = 0; |
|  | while (i < data.length) { |
|  | //var gray = 0.299 \* data[i] + 0.587 \* data[i + 1] + 0.114 \* data[i + 2];//rgb to grayscale (optional) |
|  | fragmant.push(Math.trunc(data[i])); |
|  | i += 4; |
|  | } |
|  | return fragmant; |
|  | }; |
|  |  |
|  | function isSimilar(arrayPicture, arrayPattern, precision) { |
|  | var kof = correlation(arrayPicture, arrayPattern); |
|  | return kof >= precision && kof <= 1; |
|  | } |
|  |  |
|  | function loop(count, a, b, m, n, k, l, drawContextTargetImage, pictureContext, Picture, patternContext, Pattern, cycle, cycleCount, cyclesCount, progress, recognize, precision) { |
|  | setTimeout(function() { |
|  | var grayPicture = imageToGrayArray(pictureContext.getImageData(k, l, a, b).data), |
|  | grayPattern = imageToGrayArray(patternContext.getImageData(0, 0, a, b).data); |
|  |  |
|  | if (isSimilar(grayPicture, grayPattern, precision)) { |
|  | count++; |
|  | drawContextTargetImage.beginPath(); |
|  | drawContextTargetImage.lineWidth = "1"; |
|  | drawContextTargetImage.strokeStyle = "red"; |
|  | drawContextTargetImage.rect(k, l, a, b); |
|  | drawContextTargetImage.stroke(); |
|  | } |
|  | if (k === m - a && l === n - b) { |
|  | console.info(new Date()); |
|  | recognize.prop('disabled', false); |
|  | progress.hide(); |
|  | } |
|  | cycle.text(cycleCount); |
|  | var percents = Math.floor(100 \* cycleCount / cyclesCount); |
|  | progress.progressbar({ |
|  | value: percents |
|  | }); |
|  | }, 1); |
|  | }; |
|  |  |
|  | return { |
|  | countSimilar: function(drawContextTargetImage, pictureContext, Picture, patternContext, Pattern, cycles, cycle, progress, recognize, precision) { |
|  | var k, l, a, b, m, n, count, |
|  | condition, cycleCount; |
|  |  |
|  | a = Pattern.width; |
|  | b = Pattern.height; |
|  | m = Picture.width; |
|  | n = Picture.height; |
|  | cycles.text('/' + (m - a) \* (n - b)); |
|  |  |
|  | console.info(new Date()); |
|  | console.error(precision); |
|  | cycleCount = 0; |
|  | count = 0; |
|  | for (l = 0; l <= n - b; l++) |
|  | for (k = 0; k <= m - a; k++) { |
|  | ++cycleCount; |
|  | loop(count, a, b, m, n, k, l, drawContextTargetImage, pictureContext, Picture, patternContext, Pattern, cycle, cycleCount, (m - a) \* (n - b), progress, recognize, precision); |
|  | } |
|  |  |
|  | return count; |
|  | } |
|  | }; |
|  | }()); |

**Main.js:**

|  |
| --- |
| $(document).ready(function() { |
|  | var contextTargetImage, |
|  | drawContextTargetImage, |
|  | contextPatternImage, |
|  | precision; |
|  |  |
|  | $('#recognize').prop('disabled', true); |
|  |  |
|  | $('#file-target').on('change', function() { |
|  | prepareTargetImage(); |
|  |  |
|  | if ($('#file-target').get(0).files.length !== 0 && $('#file-pattern').get(0).files.length !== 0) |
|  | $('#recognize').prop('disabled', false); |
|  | }); |
|  | $('#file-pattern').on('change', function() { |
|  | preparePatternImage(); |
|  | }); |
|  |  |
|  | function prepareTargetImage() { |
|  | setTimeout(function() { |
|  | $('#break1 canvas').remove(); |
|  | $('#break3 canvas').remove(); |
|  |  |
|  | var targetImage = document.getElementById("target-image"); |
|  | contextTargetImage = loadImage(targetImage, '1'); |
|  | drawContextTargetImage = loadImage(targetImage, '3'); |
|  | }, 100); |
|  | }; |
|  |  |
|  | function preparePatternImage() { |
|  | setTimeout(function() { |
|  | $('#break2 canvas').remove(); |
|  |  |
|  | var patternImage = document.getElementById("pattern-image"); |
|  | contextPatternImage = loadImage(patternImage, '2'); |
|  |  |
|  | if ($('#file-target').get(0).files.length !== 0 && $('#file-pattern').get(0).files.length !== 0) |
|  | $('#recognize').prop('disabled', false); |
|  | }, 100); |
|  | }; |
|  |  |
|  | if ($('#target-image').attr('src') && $('#pattern-image').attr('src')) { |
|  | $('#recognize').prop('disabled', false); |
|  | preparePatternImage(); |
|  | prepareTargetImage(); |
|  | } |
|  |  |
|  | $('#recognize').on('click', function() { |
|  | $('#cleaner').prop('disabled', false); |
|  |  |
|  | var targetImage = document.getElementById("target-image"), |
|  | patternImage = document.getElementById("pattern-image"); |
|  |  |
|  | $('#break3 canvas').remove(); |
|  | $('#progressbar').show(); |
|  |  |
|  | drawContextTargetImage = loadImage(targetImage, '3'); |
|  | precision = $('#slider').slider("value"); |
|  | $('#recognize').prop('disabled', true); |
|  | $('#target-image').hide(); |
|  |  |
|  | var count = Reco.countSimilar(drawContextTargetImage, contextTargetImage, targetImage, contextPatternImage, patternImage, $('#cycles'), $('#cycle'), $('#progressbar'), $('#recognize'), precision); |
|  | }); |
|  |  |
|  | $('#slider').slider({ |
|  | min: 0.5, |
|  | max: 1, |
|  | step: 0.02, |
|  | value: 0.94, |
|  | slide: function(event, ui) { |
|  | $('#slider-input').text('Наближення: ' + ui.value); |
|  | }, |
|  | stop: function(event, ui) { |
|  | console.error(ui.value); |
|  | precision = ui.value; |
|  | } |
|  | }); |
|  |  |
|  | }); |
|  |  |
|  | function loadImage(image, index) { |
|  | var canvas = document.createElement("canvas"); |
|  | document.getElementById('break' + index).appendChild(canvas); |
|  |  |
|  | canvas.width = image.width; |
|  | canvas.height = image.height; |
|  |  |
|  | var context = canvas.getContext("2d"); |
|  | context.drawImage(image, 0, 0); |
|  | return context; |
|  | }; |
|  |  |
|  | function onFileSelected(event, imageName) { |
|  | var selectedFile = event.target.files[0]; |
|  | var reader = new FileReader(); |
|  |  |
|  | var imgtag = document.getElementById(imageName); |
|  | imgtag.title = selectedFile.name; |
|  |  |
|  | reader.onload = function(event) { |
|  | imgtag.src = event.target.result; |
|  | }; |
|  |  |
|  | reader.readAsDataURL(selectedFile); |
|  | } |

**Style.css:**

|  |
| --- |
| .container { |
|  | height: 1000px; |
|  | } |
|  |  |
|  | .side { |
|  | width: 50%; |
|  | } |
|  |  |
|  | .left-side { |
|  | float: left; |
|  | padding-right: 0px; |
|  | } |
|  |  |
|  | .right-side { |
|  | float: right; |
|  | padding-left: 0px; |
|  | } |
|  |  |
|  | .bottom-side { |
|  | clear: both; |
|  | width: 100%; |
|  | } |
|  |  |
|  | #recognize { |
|  | padding: 6px; |
|  | font-family: Helvetica; |
|  | float: left; |
|  | } |
|  |  |
|  | #progressbar { |
|  | width: 205px; |
|  | } |
|  |  |
|  | .clear { |
|  | clear: both; |
|  | } |
|  |  |
|  | .box { |
|  | padding: 10px; |
|  | border: solid 3px; |
|  | display: inline-block; |
|  | } |
|  |  |
|  | .box-left { |
|  | float: right; |
|  | } |
|  |  |
|  | .box-right { |
|  | float: left; |
|  | } |
|  |  |
|  | .box-bottom { |
|  | padding: 10px; |
|  | border: solid black; |
|  | background-color: #FFC107; |
|  | margin: 0px auto; |
|  | display: table; |
|  | } |
|  |  |
|  | .box-bottom-panel { |
|  | height: 30px; |
|  | } |
|  |  |
|  | .title { |
|  | font-size: 16px; |
|  | font-family: Helvetica; |
|  | line-height: 20px; |
|  | display: block; |
|  | } |
|  |  |
|  | .font-pretty { |
|  | font-size: 14px; |
|  | font-family: Helvetica; |
|  | line-height: 16px; |
|  | } |
|  |  |
|  | .bold { |
|  | font-weight: bold; |
|  | } |
|  |  |
|  | .ui-widget-header { |
|  | background: #2196F3; |
|  | } |

**Index.html:**

|  |
| --- |
| <html lang="en"> |
|  |  |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Recognition</title> |
|  | <link rel="stylesheet" href="jquery-ui.min.css"> |
|  | <link rel="stylesheet" href="style.css"> |
|  | </head> |
|  |  |
|  | <body> |
|  | <div class="container"> |
|  | <div class="side left-side"> |
|  | <div class="box box-left"> |
|  | <span class="title">Цільове Зображення:</span> |
|  | <div id='break1' style="display: none"></div> |
|  | <div id='break3'></div> |
|  | <img id="target-image" style="display: none" src="mona\_small.png" alt="target-image"> |
|  | </div> |
|  | </div> |
|  | <div class="side right-side"> |
|  | <div class="box box-right"> |
|  | <span class="title">Зображення Еталон:</span> |
|  | <img id="pattern-image" src="pattern.png" alt="pattern-image"> |
|  | <div id='break2' style="display: none"></div> |
|  | </div> |
|  | </div> |
|  | <div class="side bottom-side"> |
|  | <div class="box-bottom font-pretty"> |
|  | <div class="box-bottom-panel"> |
|  | <button id="recognize">Почати Розпізнавання</button> |
|  | <div id="progressbar"></div> |
|  | </div> |
|  | <p class="clear"> |
|  | <span id="slider-input" class="bold">Наближення: 0.94 |
|  | </span> |
|  | <div id="slider"></div> |
|  | </p> |
|  | <p><span class="bold">Цільове Зобр.:</span> |
|  | <input id="file-target" type="file" onchange="onFileSelected(event, 'target-image')"> |
|  | </p> |
|  | <p> |
|  | <span class="bold">Еталонне Зобр.:</span> |
|  | <input id="file-pattern" type="file" onchange="onFileSelected(event, 'pattern-image')"> |
|  | </p> |
|  | </div> |
|  | </div> |
|  | </div> |
|  | </body> |
|  | <script type="text/javascript" src="jquery.js"></script> |
|  | <script type="text/javascript" src="jquery-ui.min.js"></script> |
|  | <script type="text/javascript" src="correlation.js"></script> |
|  | <script type="text/javascript" src="recognition.js"></script> |
|  | <script type="text/javascript" src="main.js"></script> |
|  |  |
|  | </html> |