

Магістерська кваліфікаційна робота на тему:

Інформаційна технологія нейромережевого розпізнавання треків маніпулятора «миша»

Виконав студент гр. 1КН-17м Литвиненко Н. В.

Керівник: к.т.н., доц. Колесницький О.К.

Спеціальність 122 “Комп’ютерні науки”

МЕТА ТА ЗАВДАННЯ ДОСЛІДЖЕННЯ

Актуальність: актуальною задачею на даний момент є розробка програмних засобів для розпізнавання знаків, що намальовані маніпулятором «миша», для подальшого використання цієї інформації з метою прискорення роботи користувача з комп'ютером

Метою дослідження магістерської кваліфікаційної роботи є підвищення достовірності розпізнавання треків, що намальовані маніпулятором «миша», за рахунок застосування нейронної мережі.

Для досягнення поставленої мети необхідно розв'язати такі завдання:

- провести аналіз проблеми розв'язання задачі розпізнавання треків маніпулятора «миша»;
- розглянути існуючі методи вирішення задачі розпізнавання треків маніпулятора «миша» та обрати й обґрунтувати вибір методу, який задовольняє мету даної магістерської кваліфікаційної роботи;
- розробити метод розпізнавання треків маніпулятора «миша»;
- сформулювати стадії інформаційної технології, розробити структуру та алгоритм роботи програмного засобу;
- виконати програмну реалізацію запропонованої інформаційної технології;
- провести тестування програмного продукту та виконати аналіз отриманих результатів.

ОБ'ЄКТ, ПРЕДМЕТ ТА МЕТОДИ ДОСЛІДЖЕННЯ

Об'єкт дослідження – процес розпізнавання треків, що намальовані маніпулятором «миша».

Предмет дослідження – інформаційна технологія та програмні засоби розпізнавання треків маніпулятора «миша» та достовірність їх роботи.

Методи дослідження

У роботі використані наступні методи наукових досліджень:

- системного аналізу,
- розпізнавання образів,
- теорії штучних нейронних мереж для реалізації інформаційної технології розпізнавання треків маніпулятора «миша»,
- методи математичної статистики для розробки процесу розпізнавання треків маніпулятора «миша» та обрахунків результатів експериментів із програмним засобом,
- об'єктно-орієнтованого програмування.

НАУКОВА НОВИЗНА ОДЕРЖАНИХ РЕЗУЛЬТАТІВ

1. Набула подальшого розвитку інформаційна технологія розпізнавання треків маніпулятора «миша», яка відрізняється використанням штучної нейронної мережі прямого поширення, що дозволило підвищити достовірність розпізнавання треків маніпулятора «миша».

2. Удосконалено метод формування ознак треків маніпулятора миша, який відрізняється використанням синусів та косинусів кутів між послідовними точками у треку, що підвищило інформативність ознак треку і тим самим збільшило достовірність розпізнавання треків маніпулятора «миша».

ПРАКТИЧНЕ ЗНАЧЕННЯ ОДЕРЖАНИХ РЕЗУЛЬТАТІВ

1. розроблено алгоритм роботи програмного забезпечення розпізнавання треків маніпулятора «миша» на основі штучної нейронної мережі;
2. розроблено програмні засоби для розпізнавання треків маніпулятора «миша» на основі штучної нейронної мережі;

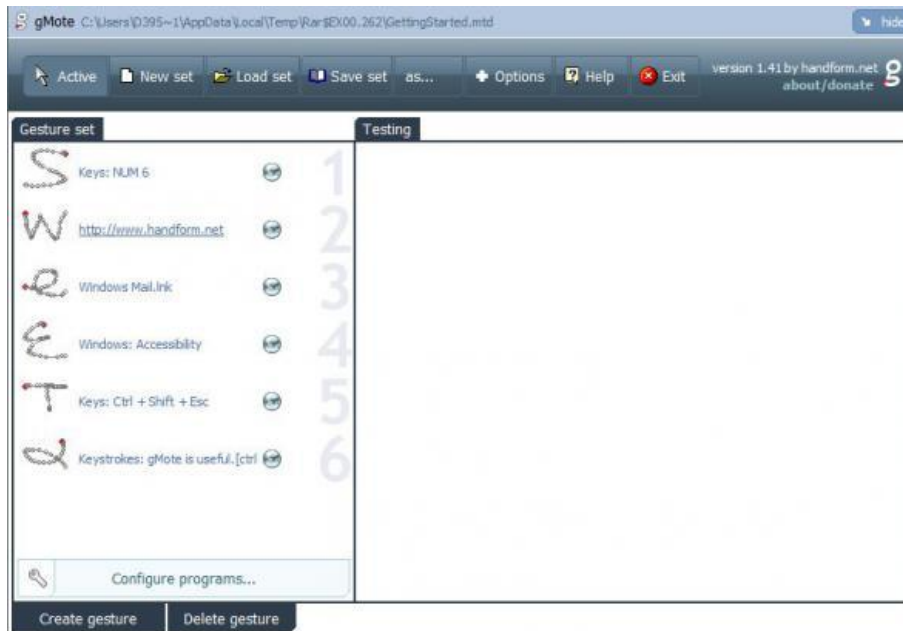
Аналіз предметної області розпізнавання треків

Розпізнавання треків (образів) — це віднесення вхідних даних до певного класу за допомогою виділення істотних ознак, що характеризують ці дані.

Розрізняють такі методи розпізнавання образів :

- Лінгвістичний (синтаксичний) метод.
- Статистичний метод.
- Евристичний метод (порівняння з еталонами).
- Нейромережевий метод.

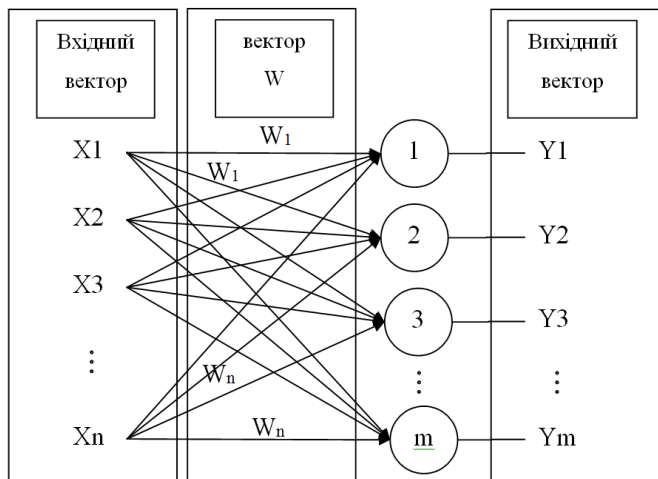
Було обрано нейромережевий метод



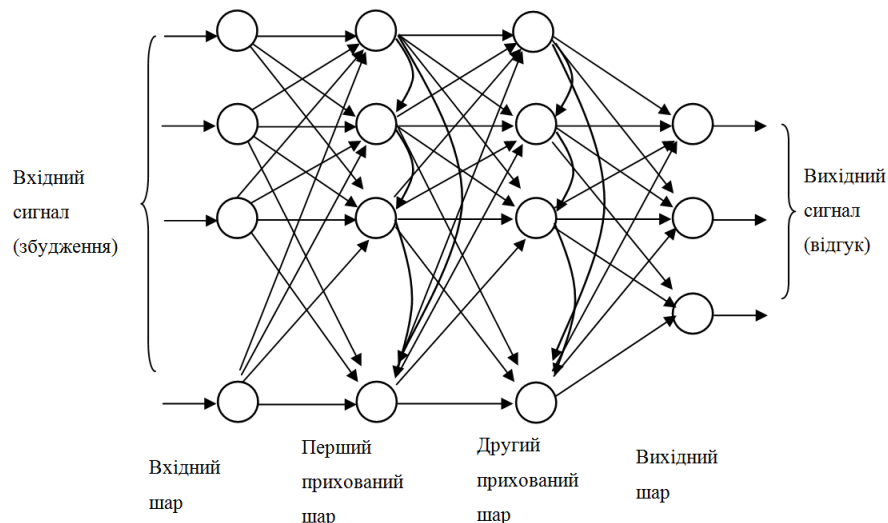
Вибір і обґрунтування аналогу

Головне вікно програми gMote 1.41 для розпізнавання жестів мишки

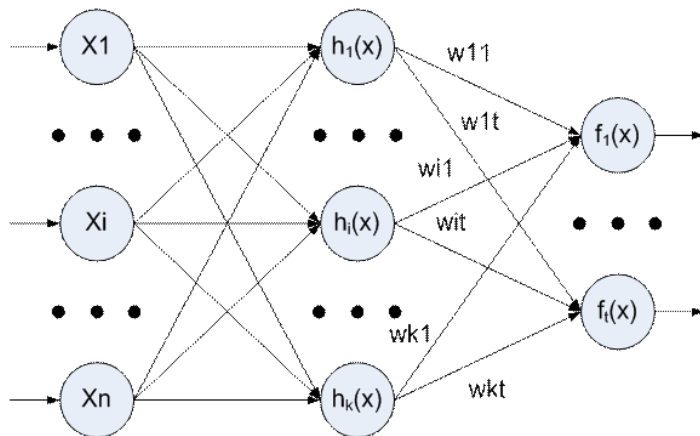
ОБГРУНТУВАННЯ ВИБОРУ НЕЙРОННОЇ МЕРЕЖІ



Одношаровий перцептрон



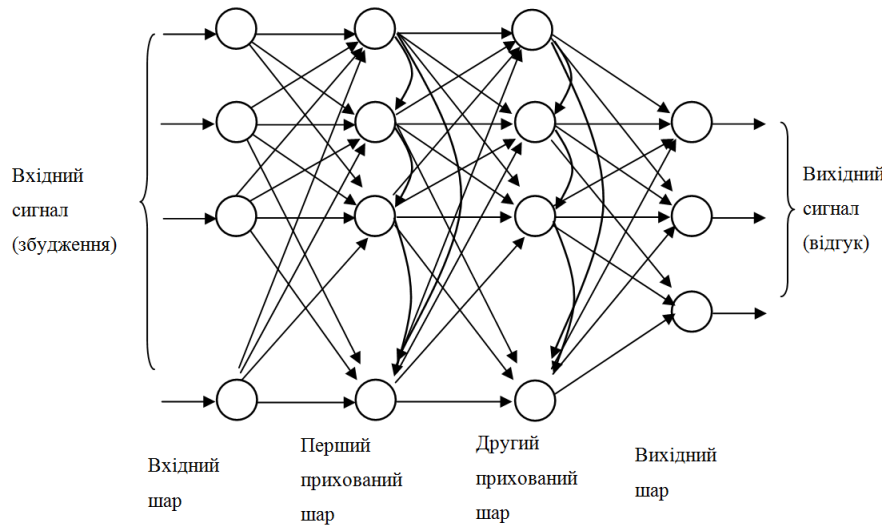
Багатошаровий перцептрон



Мережа на основі радіальних базисних функцій

Була обрана нейронна мережа багатошаровий перцептрон (нейронна мережа прямого поширення)

Архітектура та математична модель нейронної мережі прямого поширення



Функція нейрона:

$$f(x) = \text{sign}\left(\sum_{i=1}^n w_i x_i - \theta\right) \quad (3.1)$$

вагові коефіцієнти w_i зв'язків

Алгоритм зворотного поширення помилки:

1. Початкові значення ваг всіх нейронів всіх шарів V ($t = 0$) і W ($t = 0$) визначаються випадковими числами.
2. Мережі пред'являється вхідний образ X_a , в результаті формується вихідний образ $Y_1 Y_a$. При цьому нейрони послідовно від шару до шару функціонують за наступними формулами:

$$\text{Прихований шар: } x_j = \sum_i W_{ij} X_i^a; Y_i = f(x_j) \quad (3.2)$$

$$\text{Вихідний шар: } x_k = \sum_j V_{jk} Y_j; Y_k = f(x_k) \quad (3.3)$$

де $f(x)$ - сигмоїдальна функція.

Функціонал квадратичної помилки мережі для даного вхідного образу має вигляд:

$$E = \frac{1}{2} \sum_k (y_k - Y_k^a)^2 \quad (3.4)$$

Даний функціонал потрібно мінімізувати. Класичний градієнтний метод оптимізації полягає в ітераційному уточненні аргумента:

$$V_{jk}(t+1) = V_{jk}(t) - h * \frac{\partial E}{\partial N_{jk}} \quad (3.5)$$

Функція помилки не має залежності від ваги V_{jk} , тому використовуються формули неявного диференціювання складної функції:

$$\frac{\partial E}{\partial y_k} = \delta_k = (y_k - Y_k^a); \quad (3.6)$$

$$\frac{\partial E}{\partial x_k} = \frac{\partial E}{\partial y_k} * \frac{\partial y_k}{\partial x_k} = \delta_k * y_k(1 - y_k); \quad (3.7)$$

$$\frac{\partial E}{\partial N_{jk}} = \frac{\partial E}{\partial y_k} * \frac{\partial y_k}{\partial x_k} * \frac{\partial x_k}{\partial N_{jk}} = \delta_k * y_k(1 - y_k) * y_i \quad (3.8)$$

Тут врахована корисна властивість сигмоїдальної функції $f(x)$: її похідна виражається тільки через саме значення функції: $f'(x) = f(1-f)$. Таким чином, всі необхідні величини для підстроювання ваг вихідного шару V отримані.

3. На цьому кроці виконується підстроювання ваг прихованого шару. Градієнтний метод, як і раніше, дає:

$$W_{ij}(t+1) = W_{ij}(t) - h * \frac{\partial E}{\partial W_{ij}} \quad (3.9)$$

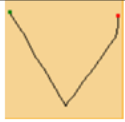
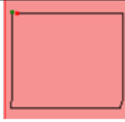

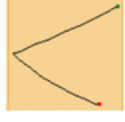
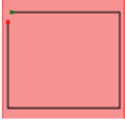

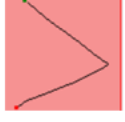
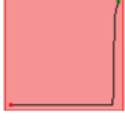
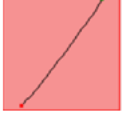
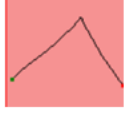
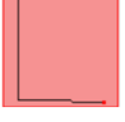
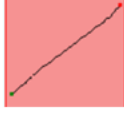
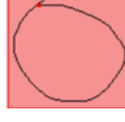




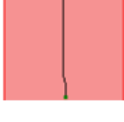



Обчислення похідних виконуються за тими ж формулами, за винятком деякого ускладнення формули для помилки d_j :

$$\begin{aligned} \frac{\partial E}{\partial x_k} &= \frac{\partial E}{\partial y_k} * \frac{\partial y_k}{\partial x_k} = \delta_k * y_k(1 - y_k) \\ \frac{\partial E}{\partial y_i} &= \delta_j = \sum_k \frac{\partial E}{\partial x_k} * \frac{\partial x_k}{\partial y_i} = \sum_k \delta_k * y_k(1 - y_k) * V_{jk}; \end{aligned} \quad (3.10)$$

$$\frac{\partial E}{\partial W_{ij}} = \frac{\partial E}{\partial y_i} * \frac{\partial y_i}{\partial x_i} * \frac{\partial x_i}{\partial W_{ij}} = \delta_j * y_j(1 - y_j) * X_i^a = [\sum_k \delta_k * y_k(1 - y_k) * V_{jk}] * [y_j(1 - y_j) * X_i^a] \quad (3.11)$$

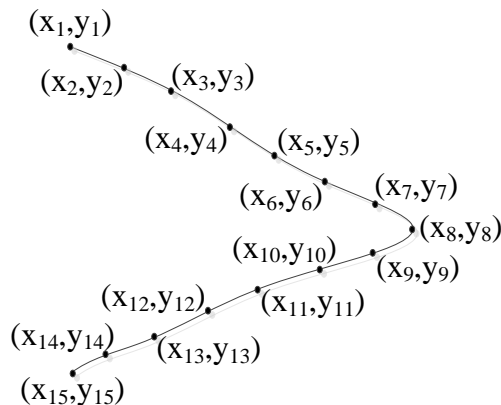
При обчисленні d_j тут і було застосовано принцип зворотного поширення помилки: окремі похідні беруться тільки по змінним наступного шару. За отриманими формулами модифікуються ваги нейронів прихованого шару.

ПРИКЛАДИ ТРЕКІВ, НАМАЛЬОВАНИХ МАНІПУЛЯТОРОМ «МИША»

ArrowDown <i>Стрілка вниз</i>		CSquare <i>Лівий квадрат</i>		LineNW <i>Лінія ВЛ (вгору-ліворуч)</i>	
ArrowLeft <i>Стрілка ліворуч</i>		CCSquare <i>Правий квадрат</i>		LineSE <i>Лінія НП (низ-праворуч)</i>	
ArrowRight <i>Стрілка праворуч</i>		LeftL <i>Ліве L</i>		LineSW <i>Лінія НЛ (низ-ліворуч)</i>	
ArrowUp <i>Стрілка вгору</i>		RightL <i>Праве L</i>		LineNE <i>Лінія ВП (вгору-праворуч)</i>	
CCCircle <i>Праве коло</i>		LineLeft <i>Лінія ліворуч</i>		LineDown <i>Лінія вниз</i>	
CCircle <i>Ліве коло</i>		LineRight <i>Лінія праворуч</i>		LineUp <i>Лінія вгору</i>	
HourGlass <i>Пісочний годинник</i>		M		S	

СТРУКТУРА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ РОЗПІЗНАВАННЯ ТРЕКІВ МАНІПУЛЯТОРА «МИША» НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ

Метод формування ознак треків маніпулятора миша, який використовує синуси та косинуси кутів між послідовними точками у треку



path {170:82 172:83 175:85 177:86 ...}
transformed into
vector {0.45 0.55 0.45 0.71 ... 0.89 0.83 0.89 0.71 ...}



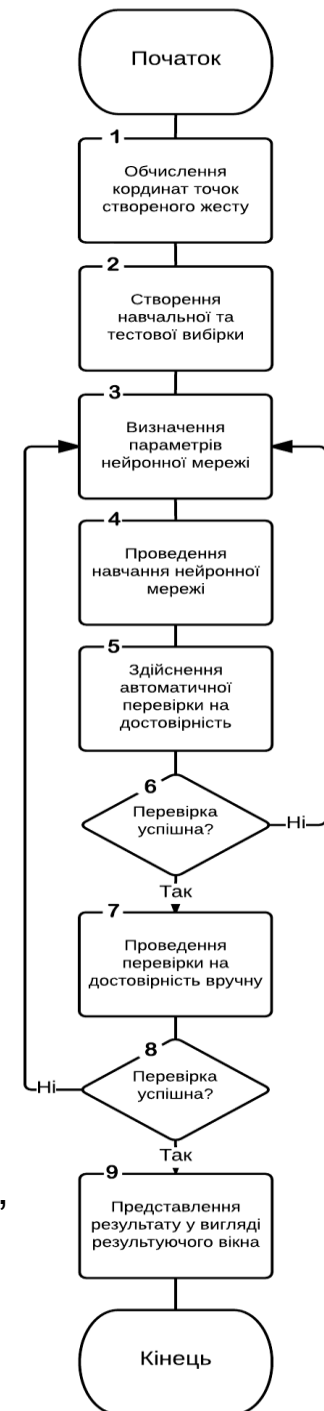
Алгоритм роботи програмного забезпечення розпізнавання треків маніпулятора «миша»

Скорочений алгоритм роботи програми:

1. Записати шлях миші
2. Представити шлях базовими точками
3. Перетворити координати точок у вектор кутів "
4. Обчислити синуси і косинуси цих кутів
5. Передати значення (косинусів і синусів) на входи мережі
6. Застосовувати SOFTmax функцію для вихідних сигналів нейронів мережі
7. Знайти і перевірити нейрон переможець, який і визначає введений знак

У програмі передбачено, що користувач може задавати:

- Кількість шарів нейронної мережі прямого поширення,
- Кількість нейронів у кожному шарі,
- Вид функції активації для нейронів окремо кожного шару (лінійна, сигмоїдальна, біполярний сигмоїд або гіперболічний тангенс),
- Параметр «альфа» обраної функції активації,
- Величину зсуву для кожного нейрона



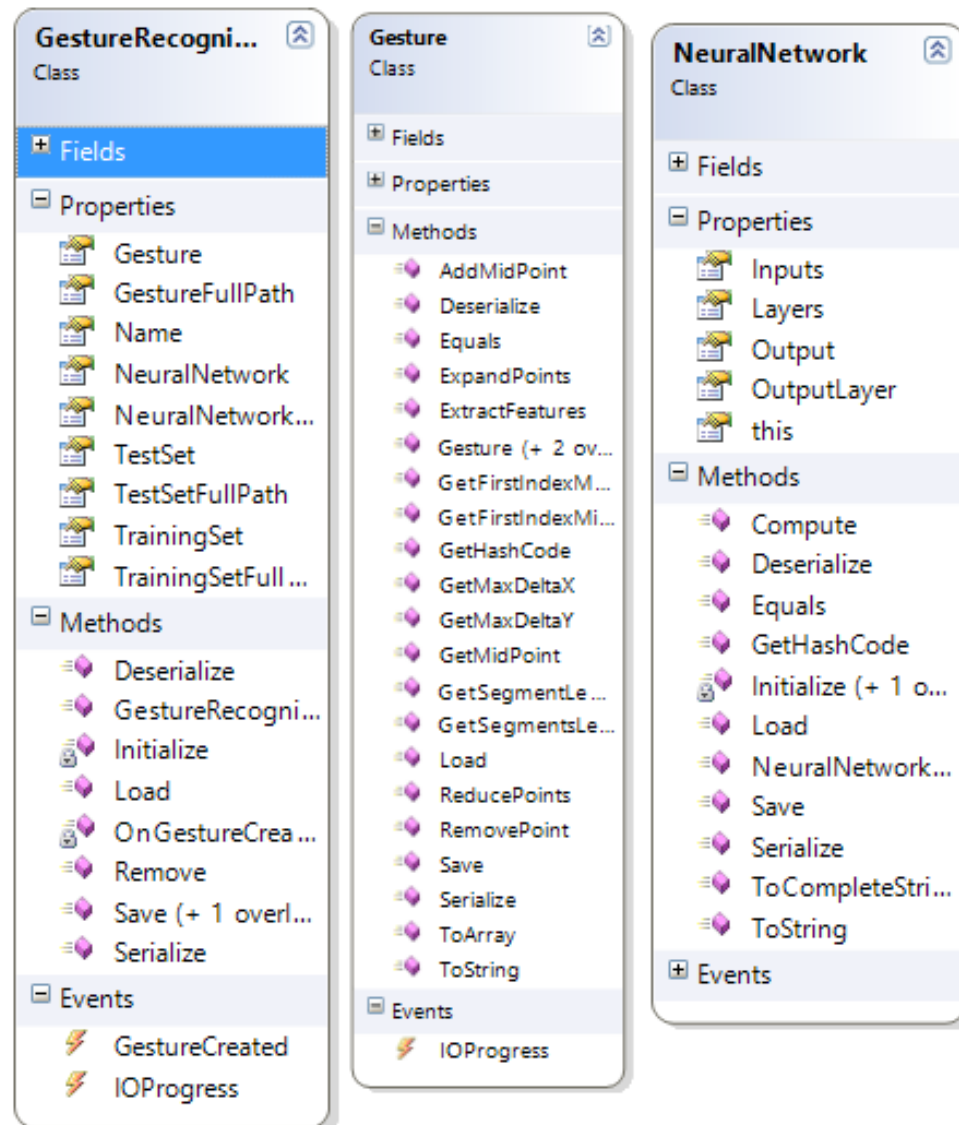
Обґрунтування вибору мови та середовища програмування

Було розглянуто такі мови програмування:

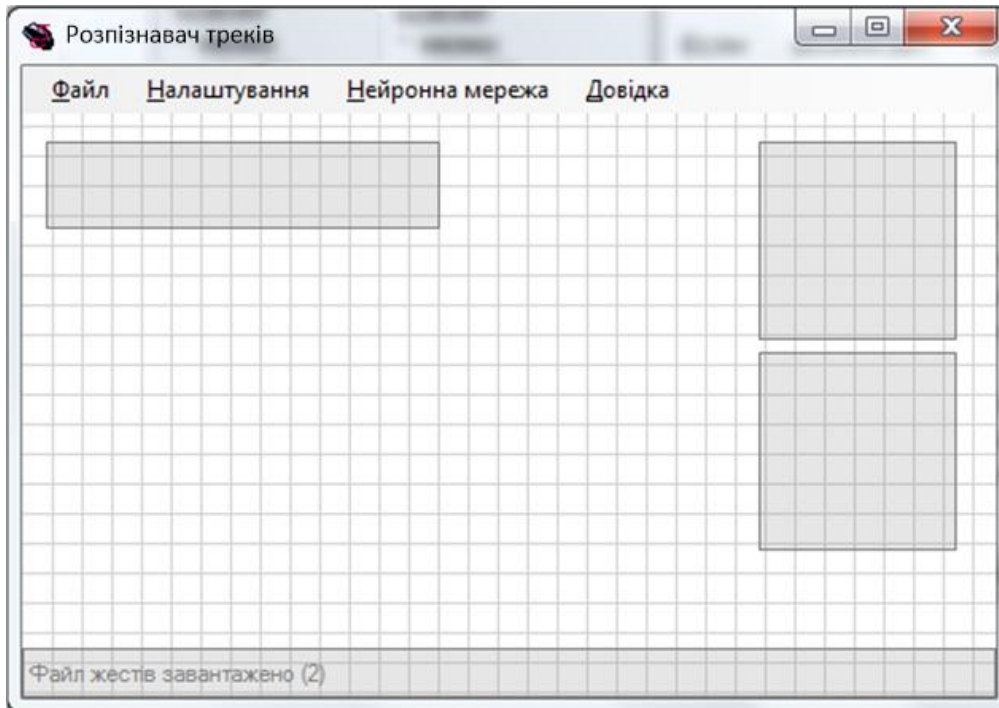
- Visual Basic,
- MatLAB,
- Java,
- C++,
- C#,

Для реалізації програми було обрано мову програмування C# та середовище розробки Microsoft Visual Studio 2015

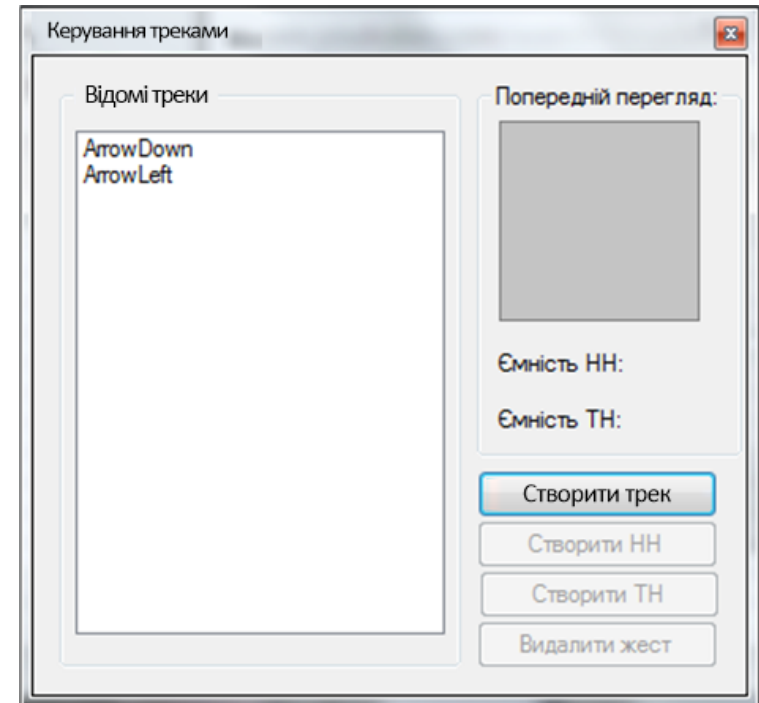
UML-діаграма класів програми розпізнавання треків маніпулятора «миша»



Робочі вікна програми розпізнавання треків маніпулятора «миша»

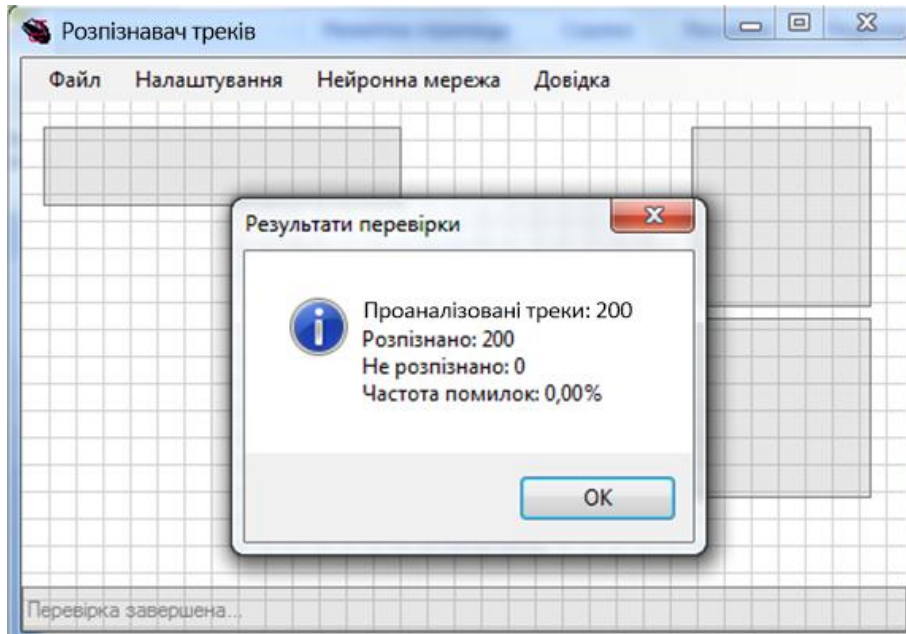


Початкове вікно програми

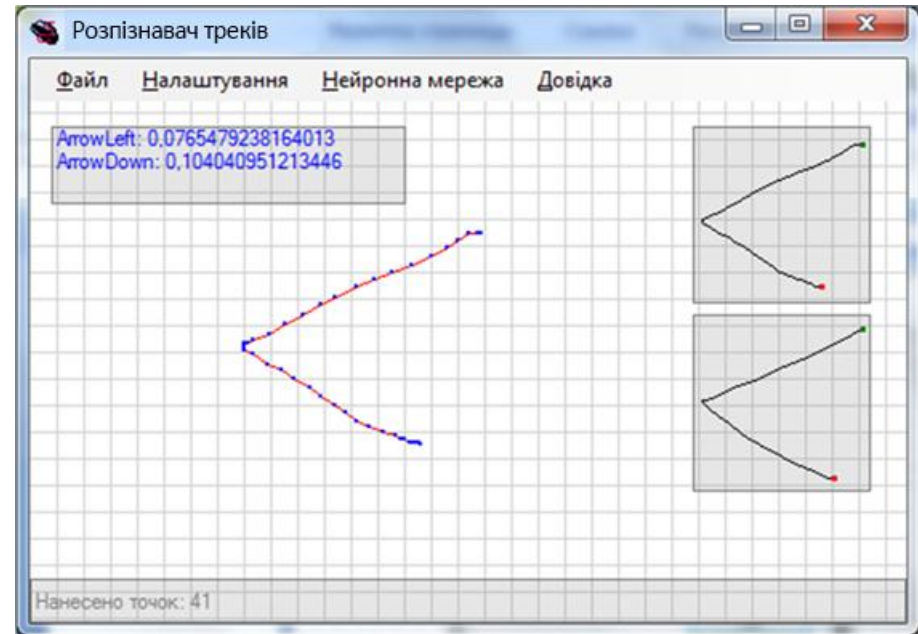


Вікно “Керування треками”

Результати роботи програми розпізнавання треків маніпулятора «миша»



Результуюче вікно перевірки на достовірність.



Результуюче вікно звірки.

Тестування та аналіз результатів роботи інтелектуального модуля

Після проведення експериментів, було сформульовано деякі практичні рекомендації:

По-перше, чим більше знаків потрібно розпізнати, тим більше потрібно додати входів нейронної мережі і зменшити темп навчання, а при появі неточності після цього – збільшити кількість шарів в нейронній мережі.

По-друге, час виконання програми залежить від кількості знаків, які потрібно розпізнати. Для пришвидшення роботи необхідно збільшити темп навчання, але це в деяких випадках призведе до втрат у точності розпізнавання.

В ході проведення експериментів було визначено, що розроблена програма має достовірність розпізнавання знаків мишки 96%. Експерименти по розпізнаванню таких самих знаків, проведені з програмою-аналогом gMote 1.41 показали, що вона має достовірність розпізнавання 91%. Це свідчить про те, що розроблена програма має вищу достовірність розпізнавання, ніж аналогічна програма, а значить мета роботи досягнута.

ЕКОНОМІЧНА ЧАСТИНА

Було проведено економічне обґрунтування доцільності розробки програми для розпізнавання треків маніпулятора «миша» на основі нейронної мережі. Нова розробка має рівень комерційного потенціалу вище середнього. Відносний рівень якості інноваційної розробки на 8% краще базового товару-конкурента. Загальна сума витрат на виконання робіт склала "44540,27" грн. Загальні витрати на виконання та впровадження результатів наукової роботи – 52400,32 грн. Абсолютна ефективність вкладених інвестицій становить 588449,32 грн, і це свідчить про те, що вкладання коштів на виконання та впровадження результатів НДДКР є доцільним. Відносна (щорічна) ефективність вкладених в наукову розробку інвестицій – 124, отже інвестор буде зацікавлений у фінансуванні даної наукової розробки. Термін окупності складає 0,8 року, тобто фінансування розробки програми розпізнавання треків маніпулятора «миша» на основі нейронної мережі є економічно доцільним.

АПРОБАЦІЯ РЕЗУЛЬТАТІВ РОБОТИ ТА ПУБЛІКАЦІЇ

Апробація результатів роботи.

Результати досліджень апробовані на шостій міжнародній науково-технічній конференції «Оптоелектронні інформаційні технології «ФОТОНІКА-ОДС-2018»», м. Вінниця, 2-4 жовтня 2018 року.

Публікації.

За результатами магістерської кваліфікаційної роботи опубліковано 1 тези доповідей на конференції та підготовлена заявка на авторське свідоцтво на твір (програму).

ВИСНОВОК

В результаті виконання МКР розроблено інформаційну технологію та програмне забезпечення для розпізнавання треків маніпулятора «миша» на основі нейронної мережі. Програмне забезпечення створено об'єктно-орієнтованою мовою програмування C# з використанням технології Windows Forms та середовища розробки Microsoft Visual Studio 2015. Програма має підвищену на 5% достовірність розпізнавання (96%) порівняно із відомими програмами аналогічного функціонального призначення (91%). Отже, мета роботи досягнута.

Дякую за увагу!