

В.М. Сокіл, О.О. Карпін, В.Т. Кремінь, Т.В. Кулик  
Національний університет “Львівська політехніка”,  
кафедра електронних обчислювальних машин

## РОЗПІЗНАВАННЯ РУХІВ НА СЕНСОРНІЙ ПАНЕЛІ

© Сокіл В. М., Карпін О.О., Кремінь В.Т., Кулик Т.В., 2008

**Наведено новий метод розпізнавання рухів на сенсорних панелях. Метод надає можливість розпізнавати найпоширеніші рухи, що виконуються одним або декількома ємнісними об'єктами (наприклад, пальцями). Реалізація методу на сучасних сенсорних дисплеях доводить його працездатність та ефективність.**

**The new method of gesture recognition on sensing panels was proposed. This method gives the possibility to recognize the most popular gestures, which are performed by one or several capacitive objects (for example by finger). Implementation of the proposed method confirms its availability and efficiency.**

**1. Вступ.** Сьогодні електронні обчислювальні машини (ЕОМ) використовують у багатьох галузях людської діяльності, будучи зручним та багатофункціональним інструментом для вирішення різноманітних задач. Однак, користувачі ЕОМ змушені використовувати засоби взаємодії, які є мало адаптованими до можливостей людського сприйняття. Це зменшує ефективність обміну інформацією та призводить до зниження продуктивності праці. Тому виникла потреба створення нового типу інтерфейсу “людина-машина”, який би забезпечував можливість виконання традиційних для користувача дій, але був більш зручним і інтуїтивно зрозумілим недосвідченому користувачеві ЕОМ.

Втіленням цього задуму є розробка інтерфейсу, в якому подача користувачем керуючих команд комп'ютеру з використанням маніпулятора “миша” було замінено на відповідні рухи ємнісного об'єкта на сенсорній панелі, що призводить до генерації відповідних керуючих системних сигналів у відповідь на виконаний рух. Розроблено чимало методів для сприйняття фізичної присутності об'єкта на сенсорній панелі. Сучасні технічні засоби надають можливість визначення наявності та місця контакту з високою точністю.

Проте під час розробки такого інтерфейсу залишається проблемою розпізнавання рухів як певної послідовності контакту пальця з панеллю. Ця проблема виникає через індивідуальні особливості користувачів – різні люди по-різному виконують здавалось би однакові дії. Відрізняються сила натиску, швидкість переміщення, можливі зайві і нерівномірні переміщення пальців на панелі під час виконання певного руху.

**2. Аналіз останніх публікацій.** Розпізнавання руху складається з його відстеження та інтерпретації як семантично вираженої команди [1]. Отже, метою системи розпізнавання рухів є автоматизація групи процесів сприйняття і розпізнавання, пов'язаних з пошуком, виділенням, ідентифікацією, класифікацією і описом образів на основі аналізу реальних даних, отриманих тим або іншим способом. Пошук і виділення образів здійснюються на початковому етапі аналізу під час обробки початкових даних і виконуються для того, щоб отримати деякі проміжні результати, які краще представляють образи з погляду розв'язання відповідної задачі. Наступний етап, розроблення “класифікатора”, передбачає: аналіз вибіркового (або перетворених) даних; синтез моделі, що враховує мінливість образів, які належать деякому класу; вибору із заданого набору характеристик

деякої їх підмножини, яка характеризує окремі класи об'єктів; визначення методів виділення вказаної підмножини; і розроблення власне алгоритму розпізнавання (класифікації) [2].

Сьогодні відомо декілька різних методів розпізнавання рухів на сенсорній панелі: прихована модель Маркова (НММ) [3], нейронна мережа затримки часу (TDNN) [4], мережа Елмана, метод динамічної трансформації шкали часу (DTW) [5], динамічне програмування [6], класифікатор Байеса [7], багатошаровий перцептрон (розпізнаючий елемент) [8], генетичні алгоритми [9], механізм нечіткого виводу [10], метод відповідності шаблонів [11], метод конденсації, функції радіального базису, самоорганізуюча карта, нейронна мережа Хопфілда [12], бінарні асоціативні машини, синтаксичне розпізнавання образів, дерево рішень [13–14], статичні нейронні мережі [15].

Приховані моделі Маркова (НММ), дуже популярні в розпізнаванні мови, були використані для розпізнавання рухів [16, 17]. НММ моделює систему як марківський процес, що рухається між різними (прихованими) станами, видаючи певні символи під час переходу від одного стану до іншого.

Значна частина алгоритмів розпізнавання рухів ґрунтується на використанні нейронних мереж. Представниками цієї групи алгоритмів є: нейронна мережа затримки часу, багатошаровий перцептрон, нейронна мережа Хопфілда. Застосовуючи статичні нейронні мережі до розпізнавання жестів, необхідно мати базу даних зразків рухів для навчання мережі [15]. Нейронні мережі є стійкими щодо шумів та поділяють простір розпізнавання на оптимальні за об'ємом розділи. Головний недолік цього методу – потреба в навчанні, яке повинне бути повторним, якщо було введено новий рух. Окрім того, імовірна необхідність навчання для кожного користувача. Іншими проблемами є ризик надмірного навчання; високі вимоги до обчислювальних ресурсів; та проблема пошуку відповідних навчальних даних.

Деякі алгоритми ґрунтуються на обчисленні ймовірності появи певного руху (класифікатор Байеса). Імовірність розпізнавання руху можна визначити як умовну ймовірність того, що жест належить до певного класу рухів, якщо присутня певна група ознак. Імовірнісний підхід не може дати однозначної відповіді про рух, оскільки імовірність приналежності до групи рухів не гарантує точності результату розпізнавання.

Численна група алгоритмів ґрунтується на порівнянні руху із заздалегідь збереженими зразками. Порівняння зі зразком можна виконувати на двох рівнях – на рівні необроблених даних або певних обрахованих характеристичних ознак як в просторовій, так і в частотній областях. Цей метод обчислює кореляцію зі всіма точками шаблону для всіх пікселів зображення. Такий підхід, як і нейронні мережі, має високі вимоги до обчислювальних ресурсів. Іншим недоліком є неможливість розпізнавання руху, якщо він трохи відрізняється від збереженого зразка.

Деякі кращими характеристиками володіють алгоритми, що ґрунтуються на механізмі нечіткого виводу. Вони частково спроможні вирішити проблему розпізнавання рухів при неточних даних, оскільки траєкторія під час руху не завжди є однаковою.

**3. Постановка завдання.** Хоча відомі методи і дають можливість розпізнавати рухи, проте їх використання супроводжується певними обмеженнями і недоліками. Ці методи вимагають або наявності значних апріорних відомостей для утворення бібліотеки зразкових об'єктів та необхідності збереження їх у великій кількості (алгоритми порівняння зі зразками), або виконання складних математичних обчислень (ймовірнісні алгоритми). Нейронні мережі здатні розпізнавати рухи, що дещо відрізняються від зразкових. Але, вони потребують попереднього тривалого навчання. Все це призводить до обмеження впровадження цих методів для розпізнавання рухів.

Тому необхідно розробити метод, який буде спроможний однозначно розпізнавати рухи та не вимагатиме значних апаратних і обчислювальних ресурсів.

**4. Метод розпізнавання рухів на сенсорній панелі.** Розроблений метод розпізнавання рухів (жестів) полягає в послідовній обробці сигналів від сенсорної панелі. Цю обробку можна поділити на три фази:

- 1) введення точок дотику в системі координат дисплея;

- 2) обчислення зміни координат і характеристичних величин у власній системі координат;
- 3) визначення переважаючого руху, генерування відповідного повідомлення для подальшої обробки.

Така схема дозволить забезпечити обробку і передавання всієї необхідної інформації про виявлений і розпізнаний рух.

**Введення точок дотику в системі координат дисплея.** Як вхідну інформацію для розпізнавання рухів використовують координати дотику емнісного об'єкта із сенсорною панеллю. У світі розроблено чимало сенсорних панелей, які покривають дисплеї і виконані за однією з таких технологій [2]: чутливі до тиску (резистивні), чутливі до електричних зарядів (емнісні), чутливі до акустичних хвиль, fotocутливі (інфрачервоні хвилі). Дія таких чутливих покриваючих поверхонь (т. зв. оверлеїв) дозволяє використати дисплей як засіб введення інформації в обчислювальну техніку, замінивши традиційні пристрої, такі, як миша та клавіатура. Найбільш зручними для детектування рухів є чутливі до електричних зарядів емнісні сенсорні панелі. Особливості їхньої реалізації дозволяють виявляти присутність двох і більше пальців, що є необхідною умовою розпізнавання певних рухів, серед яких зміна розміру, поворот тощо.

Розпізнавання рухів розпочинається з виявлення дотику емнісного об'єкта (палець) до панелі. Контакт з панеллю фіксується, якщо сигнал від певного сенсора є вищим, ніж визначене порогове значення. Введенням порогового значення відсіюються хибні сигнали, які з'являються внаслідок шумів панелі.

Можливим є виявлення одинарного або багатоточкового контакту з сенсорною панеллю. Щоб прийняти рішення про одинарний дотик, окремо сканується кожна з координат. Точка дотику є в місці, де визначено перетин двох сенсорів з максимальними для кожної з координат значеннями. Цей дотик дає змогу визначити такі рухи, як дотик (click), подвійний дотик (double click), прокручування (scroll), деякі рухи перетягування об'єкта (drag), які вимагають один дотик. Деякі рухи вимагають опрацювання багатоточкового дотику. Для цього потрібно брати до уваги декілька перетинів сигналів, величина яких перевищує певне порогове значення (рис. 1).

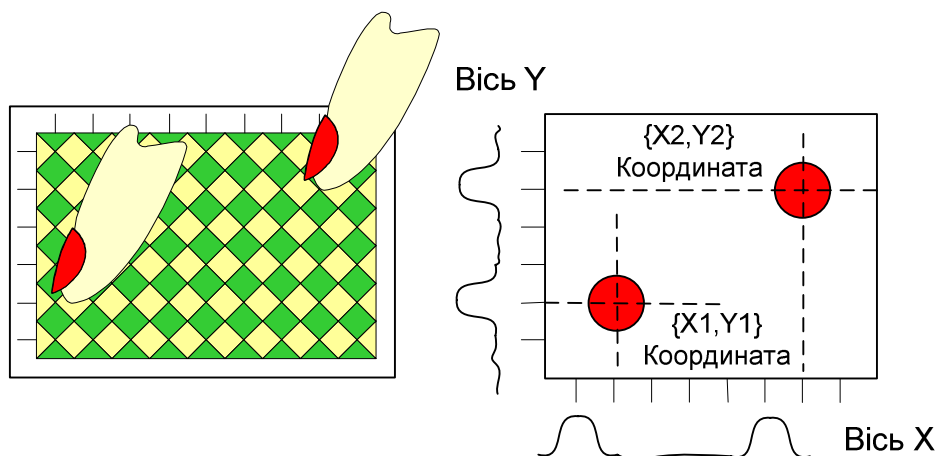


Рис. 1. Виявлення кількох дотиків

**Обчислення зміни координат і характеристичних величин у власній системі координат.** Всі рухи, що можуть бути розпізнані за допомогою сенсорної панелі, поділяють на рухи, виконані одним або декількома пальцями. Під час руху одного пальця по сенсорній панелі отримується послідовність координат  $x[0]$ ,  $y[0]$ , а під час руху двох пальців надходить послідовність значень координат двох точок дотику у системі координат дисплея  $(x[0], y[0])$ ,  $(x[1], y[1])$ . Індeksi 0 і 1 відповідають позиціям пальців (рис. 2). Для розпізнавання одноточкових рухів потрібна наявність і зміна однієї координати, а для розпізнавання рухів виконаних двома пальцями потрібна зміна двох пар координат.

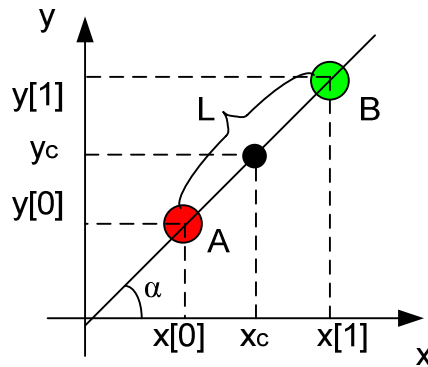


Рис. 2. Система координат дисплея і власна система координат

Зміну сигналу натискання двома пальцями по сенсорній панелі можна розглядати як рух системи з чотирма ступенями вільності. “Природними” координатами є власне координати самих точок  $(x[0], y[0])$ ,  $(x[1], y[1])$ . Але такі системи координат не описують взаємні позиції пальців, що є важливим для розпізнавання рухів. Перехід до полярних координат дає можливість обробляти взаємне положення пальців і їхні зміни. Полярними координатами запропоновані такі значення: положення центральної точки  $dx_c$ ; нахил лінії, яка з’єднує дві точки  $\alpha$ ; відстань між точками  $dl$ .

Для розпізнавання рухів зміна полярних координат є більш важливою, ніж їхні абсолютні значення. Це дає можливість уникнути небажаних ефектів у разі використання сенсорної панелі користувачами з різними переважаючими руками. Відповідно до визначення полярних координат зміна позицій двох пальців в момент  $n$  характеризується такими параметрами:

- зміна положення центральної точки  $(dx_c^n, dy_c^n)$ ;
- зміна відстані між точками  $dl_n$ ;
- кут повороту лінії (уявної), що з’єднує дві точки  $\alpha_n$ .

Обчислення характеристичних величин зміни координат вибраного об’єкта.

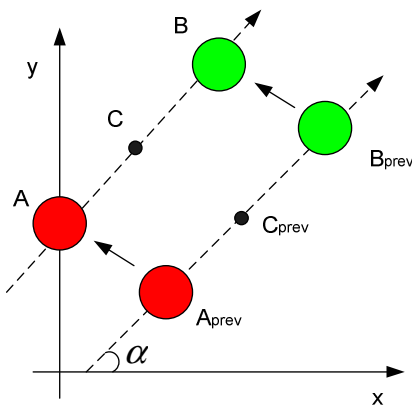


Рис. 3. Зміна координат вибраного об’єкта

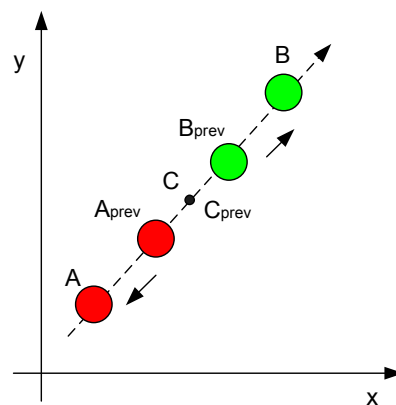


Рис. 4 : Зміна розміру вибраного об’єкта

Відбувається рух без зміни відстані між пальцями і кута нахилу віртуальної лінії, яка з’єднує пальці (рис. 3). У цьому випадку центральна точка описує траєкторію у вигляді кривої. Можливою реакцією на такий рух може бути зміна координат виділеного на дисплеї об’єкта.

Координати центральної точки обчислюються так:

$$dx_c = \frac{x[0] + x[1]}{2} - \frac{x_{prev}[0] + x_{prev}[1]}{2}; dy_c = \frac{y[0] + y[1]}{2} - \frac{y_{prev}[0] + y_{prev}[1]}{2} \quad (1)$$

Ці значення вимірюються в пікселях дисплея.

Характеристичне значення має вигляд:

$$dL_c = \sqrt{dx_c^2 + dy_c^2} \quad (2)$$

Обчислення характеристичних величин зміни розміру об'єкта або масштабування

Рух пальців, симетрично відносно центральної точки вздовж лінії, яка з'єднує їхні позиції, може бути визначене двома еквівалентними значеннями (рис. 4).

$$dL = \sqrt{(x[0] - x[1])^2 + (y[0] - y[1])^2} - \sqrt{(x_{prev}[0] - x_{prev}[1])^2 + (y_{prev}[0] - y_{prev}[1])^2} \quad (3)$$

Коефіцієнт масштабування

$$K_l = \sqrt{\frac{(x[0] - x[1])^2 + (y[0] - y[1])^2}{(x_{prev}[0] - x_{prev}[1])^2 + (y_{prev}[0] - y_{prev}[1])^2}} \quad (4)$$

Обчислення характеристичних величин повороту вибраного об'єкта

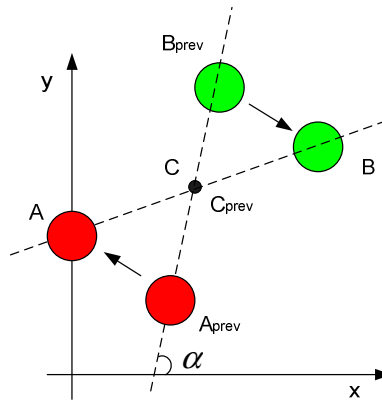


Рис. 5. Поворот вибраного об'єкта.

Рух пальців виконується навколо центральної точки без зміни відстані між ними. Великою величиною повороту є (рис. 5):

$$\tan d\alpha = \frac{(x[0] - x[1])(y_{prev}[0] - y_{prev}[1]) - (x_{prev}[0] - x_{prev}[1])(y[0] - y[1])}{(x[0] - x[1])(x_{prev}[0] - x_{prev}[1]) + (y[0] - y[1])(y_{prev}[0] - y_{prev}[1])} \quad (5)$$

Як характеристичні величини для такого руху можна використати значення  $dz = |R \sin d\alpha|$ . Великою, що характеризує розмір об'єкта, є величина  $R$ . Отже, домінуючими рухами для малих об'єктів на дисплеї будуть зміна положення і розміру, а для великих об'єктів – поворот.

**Визначення переважаючого руху, генерування відповідного повідомлення для подальшої обробки.** Одним з найпростіших рухів є короткий дотик – одинарний чи подвійний. Вхідними параметрами для обробки коротких дотиків є індикатор кількості пальців на панелі  $N$  та їхня кількість на попередньому кроці.

Короткий дотик характеризується максимальною тривалістю періоду *ClickDuration* для генерування повідомлення “Дотик” і максимальним періодом *DoubleClickPeriod* між двома дотиками для генерування повідомлення “Подвійний дотик”. Якщо зустрічаються такі рухи, то зміна координат не береться до уваги. Для фіксації події “Дотик” та “Подвійний дотик” необхідно ввести параметр *CurrentTime*, що відповідає локальному часу. Це значення може змінюватись апаратним таймером, або бути синхронним до даних, які отримуються від сенсорної панелі.

**Одинарний дотик.** Рух “один дотик” використовується або для вибору об'єкта або як початок складнішого руху. Алгоритм розпізнавання руху “один дотик” забезпечує передавання інформації про координати дотику  $(x, y)$ . Він складається з таких кроків:

1. Вимірюється сила тиску, прикладена до панелі протягом контакту пальця, та координати дотику пальця з панеллю. На основі цих даних генерується сигнал присутності пальця на панелі та координати дотику.

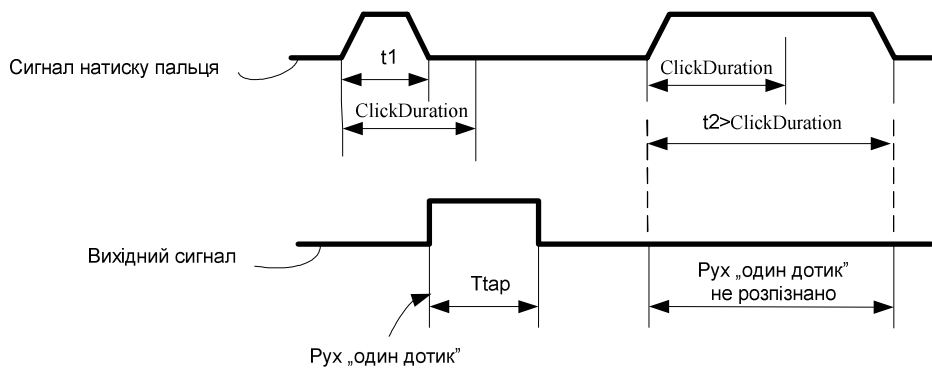


Рис. 6. Часова діаграма руху “один дотик”

2. Протягом присутності пальця на сенсорній панелі виконується одноразове вимірювання і порівняння. Під час цього вимірювання, якщо удар пальця виконаний так, що тиск є вищим за вказане порогове значення протягом часу  $t1$ , меншого, ніж вказаний період часу (наприклад,  $ClickDuration$ ) і не рухається за межами вказаного радіуса (наприклад,  $TapRadius$ ), то рух “один дотик” вважається розпізнаним.

3. Хост-системі відсилаються координати точки дотику пальця до панелі, попередньо провівши фільтрацію для того, щоб відкинути випадкові коливання координати дотику.

На рис. 6 ємнісний об’єкт (палець) присутній на панелі менше визначеного періоду часу  $ClickDuration$ . Відповідно генерується сигнал розпізнавання руху “один дотик” протягом визначеного проміжку часу  $Ttap$ . Цей сигнал представляє натиснення на віртуальну кнопку. Також на цьому рисунку представлено ігнорування присутності пальця і відсутність появи сигналу розпізнавання руху, якщо час  $t2$  є більшим за період часу  $ClickDuration$ .

**Подвійний дотик.** Подвійний дотик емулює подвійний “клік” лівою клавішею миші комп’ютера.

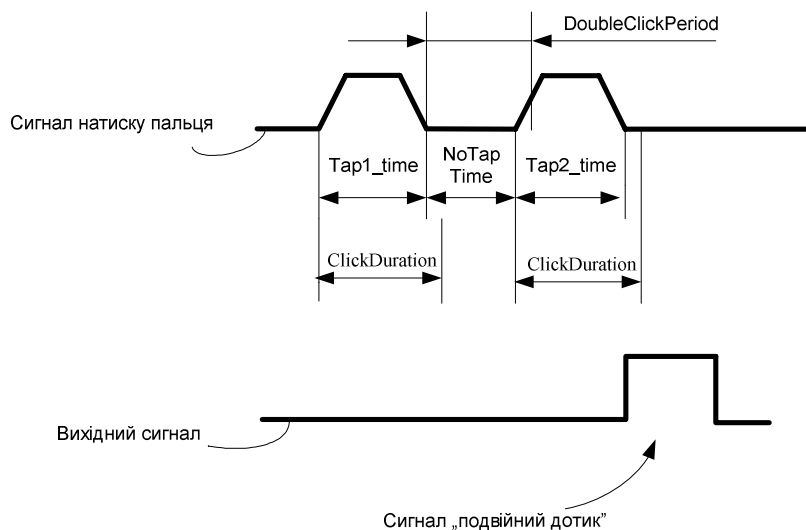


Рис. 7. Часова діаграма руху “подвійний дотик”

На рис. 7 зображена часова діаграма руху “подвійний дотик”, який розпізнано на сенсорній панелі. Рух “подвійний дотик” починається з руху “один дотик”, коли фіксується присутність пальця протягом часу  $Tap1\_time$ , меншого за проміжок часу  $ClickDuration$ . Палець відводиться від панелі на час  $NoTapTime$ . Якщо палець повертається на сенсорну панель поки час  $NoTapTime$  є меншим за час  $DoubleClickPeriod$  і присутній на панелі протягом часу  $Tap2\_Time$ , меншого за час  $ClickDuration$ , то генерується сигнал розпізнавання руху “подвійний дотик”.

*Прокручування одним пальцем.* Прокручування використовується в багатьох програмах і має багато реалізацій, що є контекстно залежними. Наприклад, дотик і перетягування об'єкта може сприйматись як прокручування. Трактування цього руху залежить від пристрою і системи, в якій буде застосовуватись прокручування. Один з варіантів реалізації визначає, що, якщо палець рухається в певному напрямку по панелі швидше ніж визначений поріг швидкості, то розпізнається рух “прокручування” і генерується відповідний сигнал.

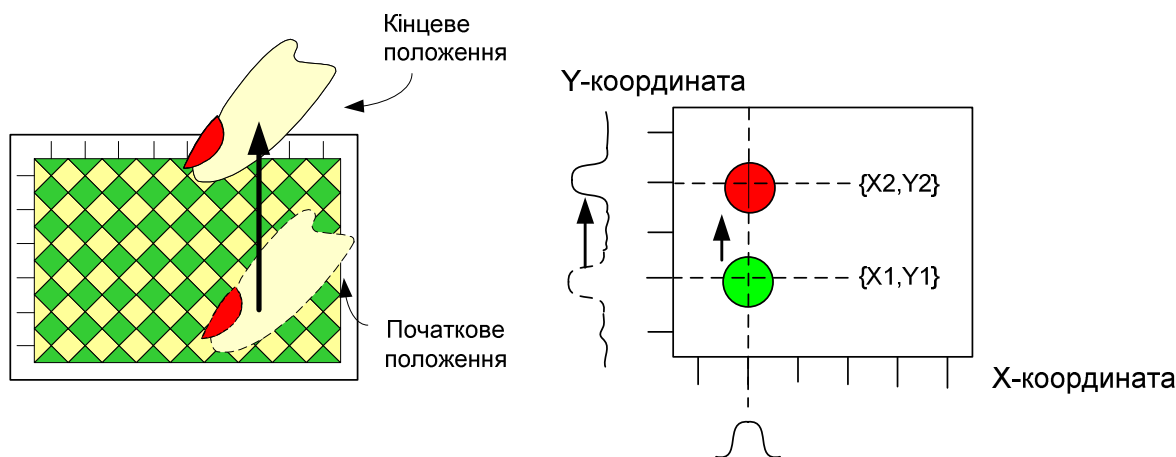


Рис. 8. Вертикальне прокручування

Напрямок прокручування залежить від напрямку руху пальця. Вертикальне прокручування зображено на рис. 8. Загальний алгоритм розпізнавання рухів, виконаних коротким дотиком одного пальця, наведені на рис. 9.

*Рухи двома пальцями.* Виконання рухів двома пальцями призначене для керування об'єктом. Для того, щоб виконати такий рух, потрібна присутність двох пальців на сенсорній панелі. Проте, це контекстно залежні рухи, і деякі з них можна замінити послідовністю ододотикових рухів. Наприклад, перетягування об'єкта двома пальцями може бути замінено на виділення об'єкта з подальшим перетягуванням одним пальцем. Загальна блок-схема алгоритму розпізнавання дводотикових рухів зображена на рис. 10.

Розпізнавання дводотикового руху здійснюється на основі обчислених на попередньому етапі характеристичних величин: зміна положення центральної точки  $(dx_c^n, dy_c^n)$ ; зміна відстані між точками  $dl_n$ ; кут повороту уявної лінії, що з'єднує дві точки  $\alpha_n$ . Зміна положення центральної точки визначає напрям руху об'єкта по віртуальній кривій. Значення  $dL$  і  $K_L$  є повністю еквівалентними. Якщо  $dL > 0$  або  $K_L > 1.0$ , то це означає збільшення розміру. Якщо  $dL < 0$  або  $K_L < 1.0$ , то відбувається зменшення розміру. Характеристична величина в цьому випадку збігається з абсолютною величиною  $dL$ . Зміна кута  $\alpha$  визначає напрям повороту. Якщо кут збільшується, то відбувається поворот вправо, інакше (якщо кут зменшується) виконується поворот вліво.

Часто виникає ситуація, коли однозначно детектувати один рух не вдається. Тоді, шляхом введення порогових значень та порівняння нормованих безсистемних величин

$$dr = \frac{\sqrt{(d\bar{x}_c^i)^2 + (d\bar{y}_c^i)^2}}{l_i}, d\Theta = \left| \frac{d\bar{\alpha}_i}{2\pi} \right|, dm = \left| \frac{d\bar{l}_i}{l_i} \right|, \quad (6)$$

де  $l_n = \sqrt{(x_1^n - x_2^n)^2 + (y_1^n - y_2^n)^2}$ , можна визначити найбільш вагомих, домінуючий рух.

**5. Реалізація алгоритмів розпізнавання рухів на сенсорній площині.** Наведені вище алгоритми розпізнавання рухів були реалізовані у вигляді окремої бібліотеки. Вихідні коди

бібліотеки написані мовою ANSI C з використанням швидких математичних алгоритмів. Бібліотека має низькі вимоги до обчислювальних ресурсів та може виконуватись на різних хост-процесорних ядрах (таких, як одноцикловий 8051, ARM7, ARM9 та ARM Cortex).

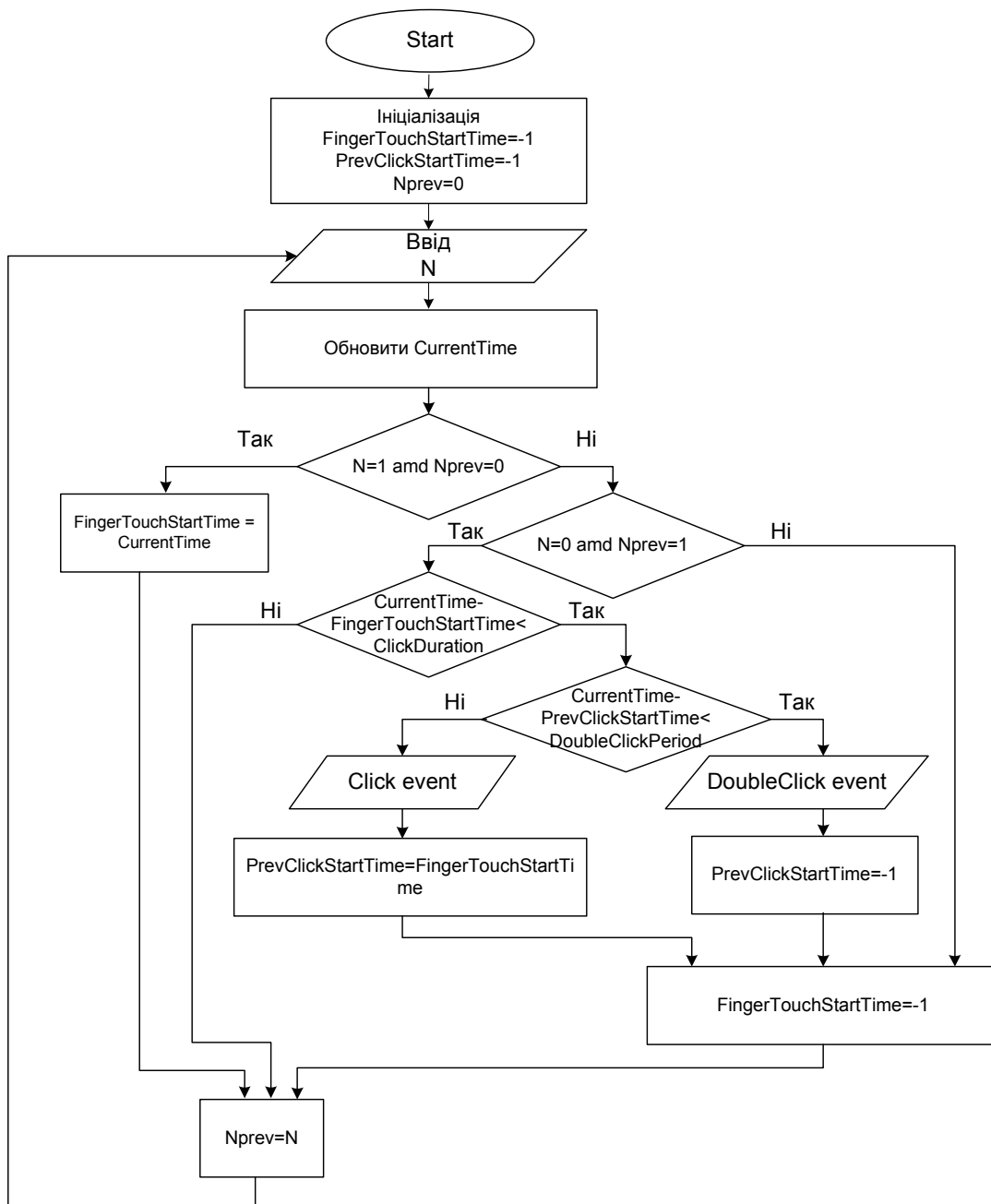


Рис. 9. Алгоритм обробки рухів, виконаних короткими дотиками одного пальця

На рис. 11 зображена друкована плата на базі мікроконтролера фірми Cypress Semiconductor та сенсорної емнісної ІТО панелі. Ця система призначена для сприйняття дотику пальців і реєстрації їх переміщення.

Результат роботи методу розпізнавання рухів показано на рис. 12. Система розпізнала рух, виконаний двома пальцями на панелі як поворот. У результаті було згенеровано та передано хост-комп'ютеру відповідне повідомлення про виявлений рух і його параметри (вид розпізнаного руху, необхідний кут повороту). Як наслідок, засобами операційної системи було здійснено поворот відповідного зображення на потрібний кут.



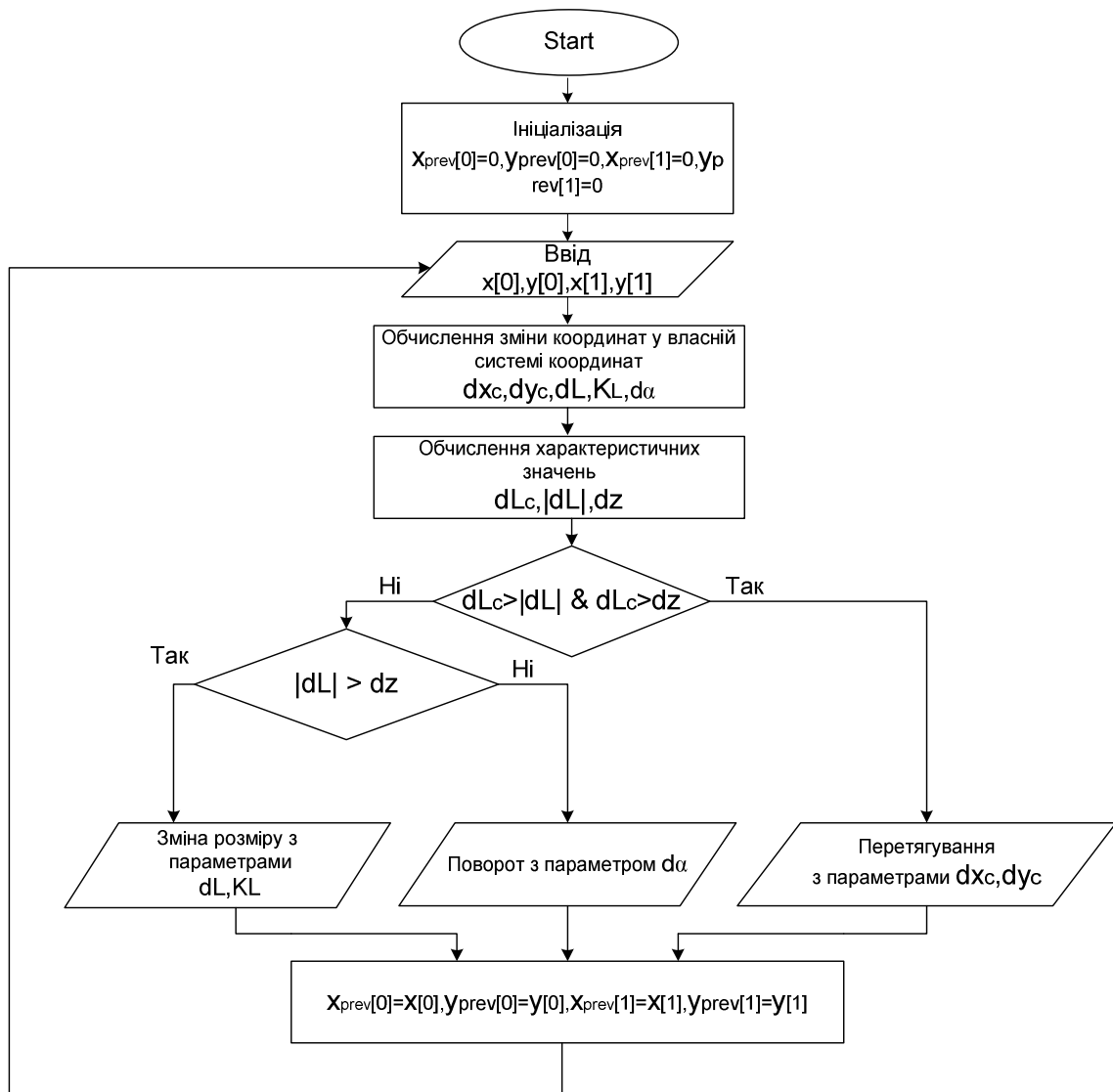


Рис. 10 : Алгоритм обробки рухів, виконаних дотиками двох пальців

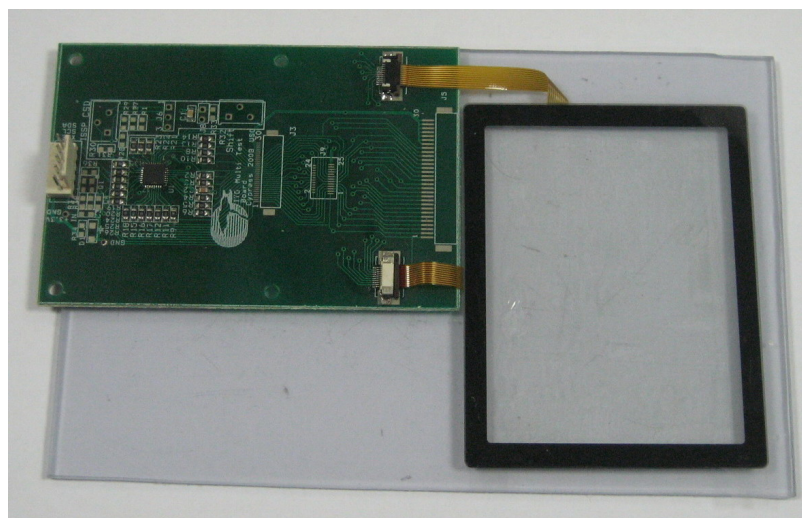


Рис. 11. Друкована плата та сенсорна панель для розпізнавання рухів

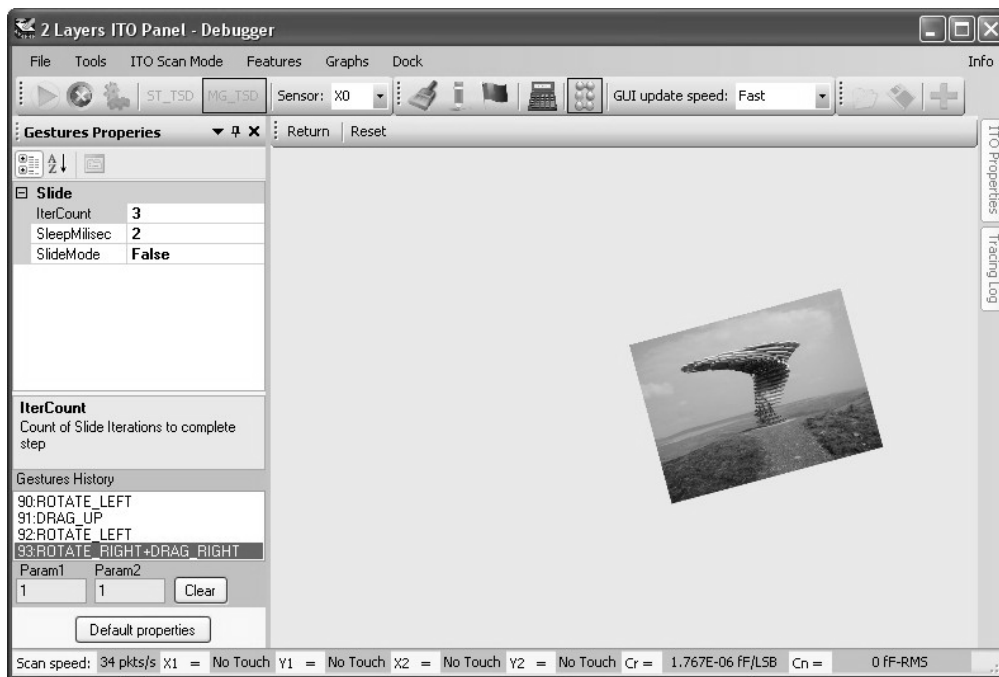


Рис. 12. Результат роботи алгоритму (поворот вибраного об'єкта)

**Висновки.** У роботі описано новий метод автоматичного розпізнавання рухів, який, на відміну від відомих, забезпечує розпізнавання рухів з різними траєкторіями, не використовуючи при тому значних обчислювальних ресурсів. Розроблено алгоритм розпізнавання рухів на сенсорній панелі, який однозначно розпізнає рух ємнісного об'єкта на панелі, забезпечує високу надійність і якість результату, враховуючи до того ж можливу специфіку виконання руху користувачами. Тестування алгоритму на групі рухів, що виконуються одним та двома пальцями, підтверджує працездатність розробленого алгоритму.

1. Hafiz Adan Habib, Muid Mufti. "Real Time Mono Vision Gesture Recognition Based Virtual Keyboard System", *IEEE Transactions on Consumer Electronics*, November 2006. 2. Журавлев Ю.М., Гуревич И.Б. Распознавание образов и распознавание изображений. – М.: Издательский дом "Наука", 1989. 3. Yie Yang Hidden Markov Model for gesture recognition, Carnegie Mellon University, 1994. 4. D.S. Clouse, Giles C.L., Home B.G., Cottrell G.W. Time delay neural networks: representation and induction of finite state machines, *IEEE transactions on neural networks*, Vol. 8, Issue 5, 1997. 5. Berndt D., Clifford J. Using dynamic time warping to find patterns in lime series, *AAAI-94 workshop on knowledge discovery in databases*. Seattle, ashington. 6. Denardo E.V. *Dynamic programming; models and applications*, Dover Publications, April 23, 2003. 7. Форсайт, Дэвид А., Понс Жан. Компьютерное зрение. Современный подход.: Пер. с англ. – М.: Издательский дом "Вильямс", 2004. – 928 с.: ил. – Парал. тит. англ. 8. S. Haykins *Neural Networks a comprehensive foundation*, Second edition. Prentice Hall, 1997. 9. M. Mitchell *An introduction to genetic algorithms*. MIT press, February 1998. 10. Li H.X., Yen V.C. *Fuzzy sets and fuzzy decision making*, CRC press 1995. 11. Demant C., Streicher-Abel S., Waszkewitz P. *Industrial image processing visual quality control in manufacturing*. Springer 1999. 12. Hersh M., Billard A. *Extended Hopfield Network for Sequence Learning: Application to Gesture Recognition*, Ecole Polytechnique F'ed'erale de Lausanne (EPFL) Swiss Federal Institute of Technology Lausanne, 2001 13. Duda R.O., Hart P.E. *Pattern classification*, Wiley Inter science. 2000. 14. Shapiro L., Stockman G. *Computer Vision The University of Washington Seattle, Washington*, 2000. 15. A. Sandberg *Gesture recognition using neural networks*, 1997. 16. Lee C. and Xu Y. *Online, interactive learning of gestures for human/robot interfaces*. In *IEEE International Conference on Robotics and Automation*, volume 4, pages 2982-2987, 1996. 17. Sterner T. and Pentland A. *Real-time American sign language recognition from video using hidden Markov models*. Technical Report 375, MIT Media Laboratory Perceptual Computing Section.