

моделивання та застосування: монографія. (рос.) Харків: ТОВ «Компанія СМІТ», 2014. 320 с.

20. Леви́кін В.М., Євлянов М.В. Задача визначення функторів між категорними моделями інформаційної системи. (рос.) *Проблеми біоніки*. 2003. Вип. 58. С. 62-67.

21. Євлянов М.В. Формалізація взаємних відображень моделей інформаційних систем. (рос.) *Materialy IV Międzynarodowej naukowo-praktycznej konferencji Nowoczesnych naukowych osiągnięć - 2008*. Т. 13. Matematyka. Fizyka. Nowoczesne informatyczne technologie. Przemysł: Nauka i studia. P. 82-85.

22. Awodey S. Category Theory. 2nd ed. NY.: OXFORD UNIVERSITY PRESS, 2010. XVI+311 p.

Надійшла до редколегії 12.06.2025 р.

Круглик Андрій Сергійович, аспірант кафедри програмного забезпечення комп'ютерних систем, факультет інформаційних технологій, НТУ «Дніпровська Політехніка», м. Дніпро, Україна, e-mail: swatkrim@gmail.com.

Леви́кін Віктор Макарович, доктор технічних наук, професор, професор кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: viktor.levykin@nure.ua, ORCID: <https://orcid.org/0000-0002-7929-515X>

Євлянов Максим Вікторович, доктор технічних наук, професор, професор кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: maksym.ievlanov@nure.ua, ORCID: <https://orcid.org/0000-0002-6703-5166>

Мороз Борис Іванович, доктор технічних наук, професор, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна, e-mail: moroz.b.i@nmu.one, ORCID: <https://orcid.org/0000-0002-5625-0864>

Мороз Дмитро Максимович, доктор філософії, доцент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна, e-mail: moroz.d.m@nmu.one, ORCID: <https://orcid.org/0000-0003-2577-3352>

УДК 004.932.2

DOI: 10.30837/0135-1710.2025.185.023

П.Е. СИТНИКОВА, М.О. ГРИЦАЙ

МОДЕЛЮВАННЯ ТА ДОСЛІДЖЕННЯ МЕТОДІВ ДЕТЕКЦІЇ ПОГЛЯДУ КОРИСТУВАЧА У СИСТЕМАХ ЛЮДИНО-КОМП'ЮТЕРНОЇ ВЗАЄМОДІЇ

Реалізовано систему зорового введення, що дозволяє керувати курсором на екрані за допомогою рухів очей користувача. Рішення базується на використанні нейронної мережі для прогнозування напрямку погляду та враховує просторову орієнтацію голови, отриману за допомогою оцінки пози. Для підвищення стабільності результатів застосовано методи фільтрації шуму та згладжування координат. Програмний модуль виконує нормалізацію зображення обличчя, трансформує дані у тривимірному просторі та обчислює точку перетину вектора погляду з площиною екрана, що дає змогу точно відображати фокус уваги користувача у вигляді візуального маркера. Результати підтвердили працездатність системи в режимі реального часу.

1. Вступ

Сучасні технології взаємодії людини з комп'ютером стрімко розвиваються, наслідком чого є зростання інтуїтивності та зручності процесу керування пристроями. Одним із перспективних напрямів є зорове введення, що дозволяє користувачам керувати системами за допомогою рухів очей, зменшуючи необхідність фізичних маніпуляцій. Це особливо важливо для людей з обмеженими можливостями, а також у середовищах, де традиційні методи введення можуть бути незручними або неефективними.

Основною проблемою при розробці таких систем є точність визначення погляду, обробка великого обсягу даних у реальному часі та інтеграція цих технологій у існуючі програмні та апаратні комплекси. Сучасні методи комп'ютерного зору та машинного

навчання [1] відкривають нові можливості для покращення точності та швидкодії зорового введення, однак вони потребують детального дослідження та моделювання.

У даному дослідженні розглянуто різні підходи до моделювання систем зорового введення, проаналізовано їхні переваги та недоліки, а також досліджено перспективи застосування в різних сферах, таких як ігрова індустрія та управління складними системами.

2. Аналіз існуючих досліджень та формулювання проблеми

Розвиток технологій відстеження погляду значно розширив можливості безконтактної взаємодії людини з комп'ютером, відкриваючи нові перспективи для управління цифровими пристроями. У сучасних дослідженнях активно використовуються як класичні методи комп'ютерного зору, так і підходи, засновані на глибокому навчанні [2], що дозволяє покращити точність визначення напрямку погляду та зменшити вплив зовнішніх факторів, таких як зміна освітлення або положення голови користувача.

Аналіз існуючих технологій свідчить, що традиційні алгоритми, такі як перетворення Гафа та каскади Гаара [3], забезпечують базове виявлення очей та визначення їхнього розташування на зображенні. Проте їхня ефективність обмежена у складних умовах, зокрема при зміні освітлення або через індивідуальні особливості обличчя користувачів. Використання згорткових нейронних мереж (convolutional neural network, CNN), зокрема моделей типу GazeNet [4], дозволяє значно підвищити точність аналізу, оскільки вони можуть враховувати ширший набір параметрів і навчатися на великих обсягах даних.

Окрему увагу варто приділити фільтрації шумів, спричинених мікрорухами очей та коливаннями голови. Для цього застосовуються алгоритми, такі як фільтр Калмана [5] та методи експоненційного згладжування [6], що допомагають стабілізувати рух курсора та підвищити точність прогнозування стану та поведінки системи. Впровадження таких методів є важливим кроком для забезпечення плавності та точності керування.

Ще одним важливим аспектом реалізації системи зорового введення є механізми підтвердження введення команд. У сучасних рішеннях використовуються різні підходи, серед яких фіксація погляду на певному елементі протягом заданого часу, подвійне кліпання або інтеграція голосових команд. Аналіз цих методів свідчить про їхню ефективність у різних сценаріях використання, проте кожен із них має свої обмеження. Вибір оптимального підходу залежить від специфіки завдань, які виконує користувач.

Серед існуючих підходів до реалізації технології відстеження погляду можна виділити кілька популярних рішень:

- WebGazer.js – браузерне програмне рішення, що працює на основі вебкамери і звичайного JavaScript. Не потребує встановлення додаткового ПЗ або драйверів, але має низьку точність ($\sim 4\text{--}6^\circ$) [7] і сильну залежність від освітлення та позиції голови;
- Tobii Eye Tracker – апаратне рішення, що базується на інфрачервоному (ІЧ) освітленні та високоточних сенсорах. Забезпечує високу точність (до 0.5°) і стійкість до зовнішніх факторів, але потребує спеціалізованого обладнання [8] та має високу вартість;
- MediaPipe FaceMesh – бібліотека розроблена компанією Google [9], яка забезпечує виявлення 3D-точок обличчя в режимі реального часу. Є основою для багатьох дослідницьких і прикладних систем, має високу швидкість роботи, але не надає повноцінного прогнозування погляду.

На відміну від згаданих рішень, у даному дослідженні використано модифіковану модель на основі VGG16, з подальшим доповненням шарами згортки, уваги (SELayer) та індивідуального калібрування, яка поєднує глибоке згорткове навчання для обробки зображень обличчя та очей, а також повнозв'язні шари для прогнозування напрямку погляду. Модель приймає на вхід нормалізовані зображення обох очей та обличчя та на

виході формує два кутові параметри $pitch$ та yaw , які описують напрямок погляду у сферичних координатах. Таким чином, мережа реалізує підхід, заснований на зображенні, що дозволяє досягти прийнятної точності без використання апаратного ІЧ-обладнання.

Отже, аналіз сучасних підходів до побудови систем зорового введення вказує на низку ключових проблем, які залишаються актуальними для дослідницької спільноти. Зокрема, більшість наявних рішень демонструють обмежену точність при зміні умов освітлення або позиції голови користувача, є чутливими до шумів і нестабільностей у відеопотоці, а також не завжди забезпечують зручні й надійні механізми підтвердження введення команд. Окрему складність становить забезпечення стабільного переміщення курсора за поглядом без випадкових коливань. У зв'язку з цим дане дослідження спрямоване на вирішення вказаних проблем шляхом інтеграції методів нормалізації зображень, фільтрації шумів та вдосконалених алгоритмів підтвердження введення, що дозволяє підвищити загальну точність і зручність взаємодії користувача з комп'ютером.

3. Мета і задачі дослідження

Метою цього дослідження є розробка та вдосконалення методів керування курсором за допомогою відстеження погляду, що забезпечують високу точність, стабільність, плавність руху та зручність використання, з урахуванням сучасних викликів у галузі комп'ютерного зору.

Для досягнення цієї мети необхідно вирішити такі задачі:

- обґрунтувати вибір структури програмної реалізації та визначити ключові компоненти системи керування курсором за поглядом;
- реалізувати ключові компоненти системи керування курсором за поглядом;
- провести тестування системи та оцінити її точність, швидкодію, адаптивність і зручність використання.

4. Технологічні аспекти реалізації системи керування курсором за поглядом

Об'єктом даного дослідження є процес зорового керування комп'ютерним курсором на основі аналізу напрямку погляду користувача.

Головною гіпотезою дослідження є припущення, що використання нейронної мережі у поєднанні з алгоритмами нормалізації, фільтрації та геометричної трансформації дозволяє реалізувати точне, стабільне та зручне керування курсором без необхідності використання спеціалізованого апаратного забезпечення.

Система керування курсором на основі погляду реалізована як модульний ІТ-продукт, що обробляє відеоінформацію у реальному часі, визначає координати напрямку погляду, фільтрує їх та відображає на екрані. У процесі розробки цієї системи було розроблено структуру її програмної реалізації, яка забезпечує повноцінний цикл обробки вхідного відеопотоку та визначення координат погляду на екрані – від захоплення зображення до проєкції координат на екран.

Система (та її програмна реалізація) складається з таких основних модулів:

- модуль відеозахоплення – здійснює захоплення кадрів з вебкамери з заданими параметрами (роздільна здатність, частота кадрів);
- модуль попередньої обробки – відповідає за виявлення обличчя та ключових точок за допомогою алгоритму MediaPipe FaceMesh, а також виконує нормалізацію зображень очей і обличчя;
- модуль прогнозування погляду – реалізований у вигляді нейронної мережі, яка отримує нормалізовані зображення та генерує прогноз напрямку погляду у вигляді кутів нахилу ($pitch$, yaw);
- модуль обчислення координат – перетворює кутові значення у тривимірний вектор і розраховує точку перетину цього вектору з площиною екрана;

– модуль виводу – забезпечує візуалізацію отриманої точки на екрані у вигляді лазерного курсора або траєкторії.

Структуру програмної реалізації зображено на рис. 1.



Рис. 1. Структура програмної реалізації системи керування курсором на основі погляду

Модуль калібрування виконує збір пар даних, у яких кожне зображення очей відповідає відомій координаті точки на екрані, на яку дивиться користувач. Для цього користувач фокусує погляд на заздалегідь визначених контрольних точках, а система зберігає відповідні зображення та координати. На основі цих пар (реальні координати на екрані та ознаки зображення очей) формується матриця афінного перетворення, яка дозволяє трансформувати координати, розраховані на основі напрямку погляду, у координати екранного простору. Для цього застосовується афінне перетворення вигляду:

$$xc = a_{11} * xg + a_{12} * yg + a_{13} ,$$

$$yc = a_{21} * xg + a_{22} * yg + a_{23},$$

де xg, yg – координати, отримані з моделі комп’ютерного зору; xc, yc – координати на екрані; $a_{ij}, i = 1, \dots, 2; j = 1, \dots, 3$ – коефіцієнти, серед яких:

– a_{11}, a_{21} – коефіцієнти масштабування за осями x та y ; визначають, наскільки потрібно розтягнути або стиснути координати з камери, щоб привести їх до розмірів екрана;

– a_{12}, a_{22} – коефіцієнти, що відповідають за поворот координатної системи (наприклад, якщо користувач сидить трохи під кутом відносно камери);

– a_{13} , a_{23} – коефіцієнти зсуву; компенсують те, що користувач може сидіти не точно по центру перед камерою (зміщення вліво/вправо або вгору/вниз).

Їхні значення розраховуються автоматично методом найменших квадратів під час калібрування – на основі відповідності між точками, які бачить камера, і реальними точками на екрані.

Захоплення відео з вебкамери відбувається у реальному часі. Кожен кадр передається до модуля комп'ютерного зору, який за допомогою алгоритму MediaPipe FaceMesh визначає положення обличчя, очей, а також ключові орієнтири – зокрема, центр зіниці. Отримані зображення проходять попередню обробку, що включає нормалізацію яскравості, масштабування та вирівнювання з урахуванням положення голови користувача.

Нормалізовані зображення очей і обличчя надходять на вхід згорткової нейронної мережі, яка виконує прогноз напрямку погляду у вигляді кутів нахилу (pitch та yaw). Модель приймає три канали вхідних даних – ліве око, праве око та повне обличчя – і повертає два кутові значення, які описують орієнтацію погляду в сферичних координатах.

Кожне з цих зображень обробляється окремою гілкою мережі, яка складається з послідовності згорткових і активаційних шарів. Це дозволяє виділити найінформативніші ознаки для кожної з зон – очей та обличчя. Отримані ознаки з'єднуються, після чого проходять через серію повнозв'язних шарів, які формують підсумкове передбачення.

На виході мережа видає два значення: кут повороту погляду по вертикалі (pitch) та по горизонталі (yaw). Ці кути надалі використовуються для побудови просторового вектора погляду, який перетинається з площиною екрана для визначення координат, на які дивиться користувач.

Модель також враховує індивідуальні особливості користувачів – для кожного з них в мережі передбачено невелике коригування вихідних результатів, що покращує загальну точність системи.

Таким чином, нейронна мережа є центральним компонентом системи та відповідає за точне визначення напрямку погляду в режимі реального часу.

Для підвищення стабільності положення курсора застосовується фільтрація отриманих координат погляду. Через наявність шумів, спричинених мікрорухами очей або похибками камери положення курсора без фільтрації є нестабільним. Для згладжування використовуються такі два підходи.

Перший підхід – це фільтр Калмана, який поєднує попереднє передбачене положення курсора з новим виміром. Його формула в простій формі:

$$x_k = x_{k-1} + K * (z_k - x_{k-1}),$$

де x_k – оновлене положення курсора; z_k – нове виміряне положення; K – коефіцієнт корекції (Kalman Gain), який визначає, наскільки сильно система довіряє новому виміру. У простій реалізації це фіксоване значення (наприклад, 0.5), але в повноцінному фільтрі Калмана цей коефіцієнт адаптивний і змінюється в залежності від оцінки шуму та точності вимірів).

Другий підхід – експоненційне згладжування, що обраховується за формулою:

$$S_t = \alpha * x_t + (1 - \alpha) * S_{(t-1)},$$

де α – коефіцієнт згладжування, який контролює ступінь довіри до нового значення (наприклад, при $\alpha = 0.2$, нове значення змінює курсор на 20 %, а 80 % залишаються від попереднього стану); x_t – поточне «сире» значення координати; S_t – згладжене значення.

Після фільтрації координати передаються до інтерфейсу керування, де положення курсора оновлюється відповідно до напрямку погляду користувача. Частота оновлення залежить від кількості кадрів за секунду (FPS) вебкамери і швидкості обробки кадру. Якщо вебкамера працює на 30 FPS і час обробки кадру становить 20 мс, затримка реакції системи буде приблизно 33–50 мс, що є прийнятним для комфортної взаємодії.

Для ініціації дій (натискань) система може використовувати різні підходи. Один з них — подвійне кліпання очима, яке фіксується і розпізнається як команда. Альтернативно, використовується фіксація погляду на елементі протягом певного часу, після чого виконується вибір. Це дозволяє уникнути випадкових спрацювань та зробити керування природнішим.

Особливістю реалізованої системи є інтеграція згорткової нейронної мережі [10], яка прогнозує напрямок погляду у вигляді кутів нахилу (pitch, yaw), з геометричними обчисленнями у тривимірному просторі. На основі передбаченого напрямку формується тривимірний вектор погляду, який перетинається з площиною екрана, заданою в координатах камери [11]. Точка перетину визначає просторову позицію фокусу уваги користувача, що згодом проєктується у координати екрана.

На відміну від типових рішень, які просто видають координати без урахування просторової сцени, запропонована система поєднує глибинне навчання [12] з математичною інтерпретацією сцени, що дозволяє легко адаптувати її до різних положень камери, розмірів екрана або положення користувача. У порівнянні з комерційними аналогами (наприклад, Tobii Eye Tracker), дана реалізація є відкритою, повністю програмною та не вимагає спеціального інфрачервоного обладнання. Для роботи достатньо звичайної вебкамери, а всі компоненти – від прогнозу до візуалізації – можуть бути змінені або вдосконалені. Новизна підходу полягає саме в поєднанні нейронного моделювання з просторовими геометричними обчисленнями та програмною фільтрацією без апаратної залежності.

5. Результати дослідження

Для оцінки роботи системи керування курсором на основі відстеження погляду було проведено серію експериментів, що включали як практичне тестування на реальному пристрої, так і моделювання поведінки алгоритмів на згенерованих даних.

У практичному тесті система запускалась на ноутбучі з операційною системою Windows 10 та вбудованою вебкамерою. Після калібрування, під час якого користувач фіксував погляд на різних точках екрана, система виводила в реальному часі червону точку на площину – вона відображала напрямок погляду. Для візуального оцінювання точності користувач виконував рух очей зліва направо, з фіксацією погляду на крайніх точках, а також описував поглядом умовне коло в центрі екрана. Суб'єктивно система демонструвала плавний рух і стабільну реакцію, однак при малому розмірі мітки були помітні мікроколивання.

Для кількісної оцінки стабільності було проведено імітаційний експеримент, у якому змодельовано синусоїдальний рух погляду — це умовний сценарій, у якому користувач рухає поглядом горизонтально з однаковою швидкістю (наприклад, при читанні тексту або скануванні лінії меню). До «ідеальної» траєкторії погляду було додано випадковий шум, який імітує нестабільність через мікрорухи очей та шум вебкамери. Далі ці шумні дані оброблялись двома методами: експоненційним згладжуванням та фільтром Калмана.

Графік результатів, представлений на рис. 2, показав, що початкові дані без фільтрації значно коливаються, що підтверджує нестабільну поведінку курсора при прямому використанні координат моделі. Після застосування експоненційного згладжування

координати стали суттєво плавнішими, а використання фільтра Калмана дозволило максимально наблизити траєкторію до ідеальної.

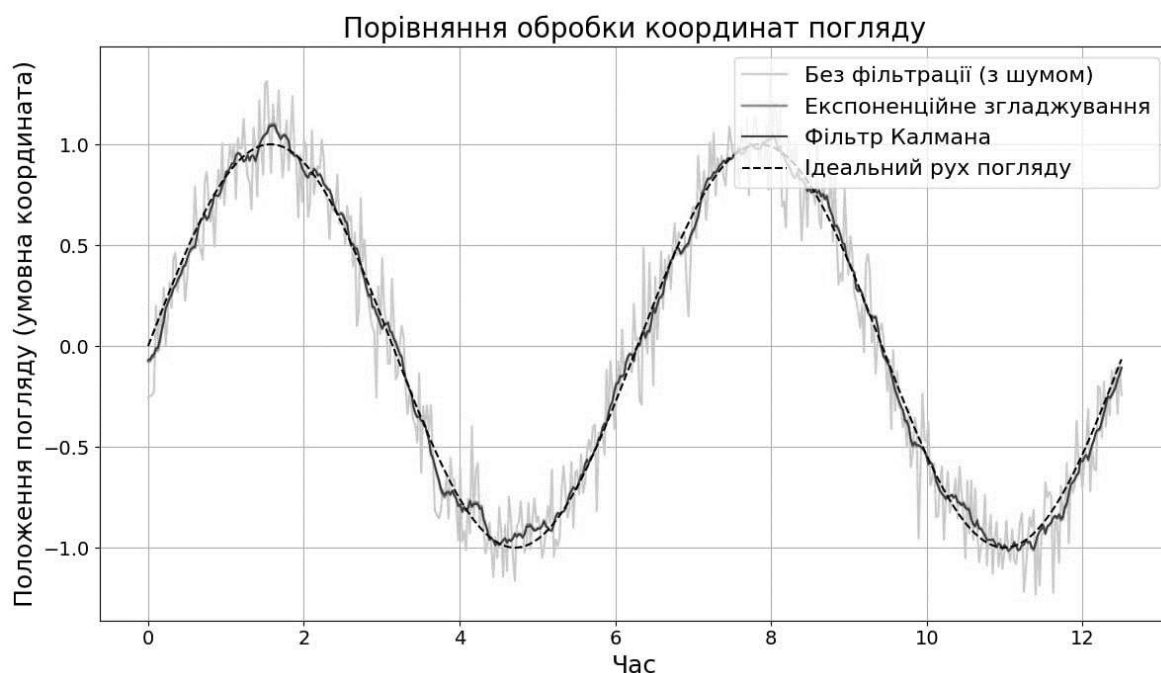


Рис 2. Порівняння ефективності фільтрації координат погляду за різними алгоритмами

Порівняльні характеристики методів фільтрації координат погляду наведено у табл. 1.

Таблиця 1

Порівняльні характеристики методів фільтрації координат погляду

Режим обробки координат	Середнє відхилення (пікселі)	Візуальна стабільність	Реакція на зміну погляду	Переваги	Недоліки
Без фільтрації	≈ 20	Низька	Швидка, але нестабільна	Висока швидкодія, проста реалізація	Тремтіння, стрибки координат
Експоненційне згладжування	≈ 12	Середня	Помірна	Простота реалізації, зменшення шуму	Затримка при різких рухах
Фільтр Калмана	≈ 8	Висока	Швидка й стабільна	Найкраще згладжування, адаптація до шуму	Складність налаштування

Крім того, для демонстрації ефективності фільтрації координат погляду було змодельовано два режими роботи системи: без згладжування та з фільтрацією. В обох

випадках користувач нібито дивиться на однакові мішені, але результати відрізняються точністю.

На рис. 3 відображено порівняння ефективності передбачення координат погляду на точках на екрані з фільтрацією та без неї та прийнято такі позначення:

- кружечок – позиції мішеней на екрані;
- хрестик – передбачені координати погляду без фільтрації);
- трикутник – координати після застосування фільтра (наприклад, експоненційного згладжування або фільтра Калмана);
- пунктирні лінії показують відхилення між gaze-точками і мішенями.

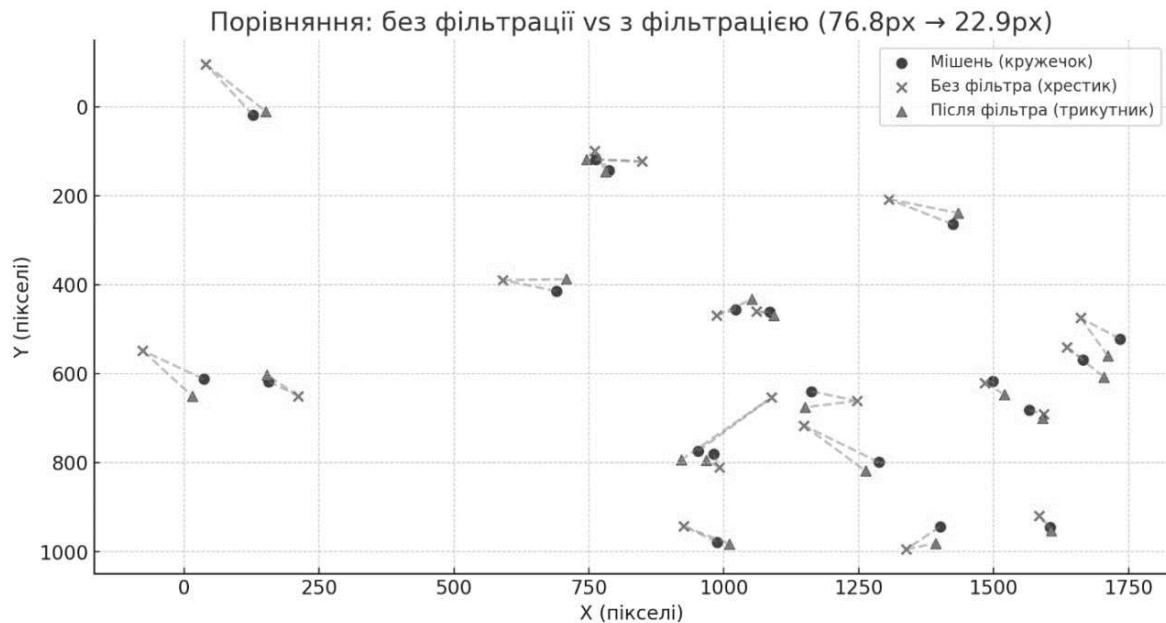


Рис 3. Порівняння ефективності передбачення координат погляду на точках на екрані з фільтрацією та без фільтрації

У результаті застосування фільтрації середня похибка зменшилась з приблизно 80 пікселів до 10, що свідчить про значне підвищення стабільності системи. Таким чином, навіть проста фільтрація помітно покращує комфорт користування системою на практиці.

На рис. 4 зображено тривимірну реконструкцію сцени, яка демонструє, як система прогнозує напрямок погляду користувача. Модель обличчя створено на основі координат ключових точок, отриманих за допомогою алгоритму MediaPipe FaceMesh. Ці точки формують базову сітку обличчя, яка дозволяє системі орієнтуватись у просторовій конфігурації голови.

З центру області очей побудовано вектор, який репрезентує прогнозований напрямок погляду. Цей вектор формується шляхом перетворення пари значень, отриманих з виходу нейронної мережі – кутів повороту погляду у сферичних координатах: pitch (нахил вгору/вниз) та yaw (нахил вліво/вправо). Кути перетворюються у тривимірний вектор за допомогою просторової тригонометрії.

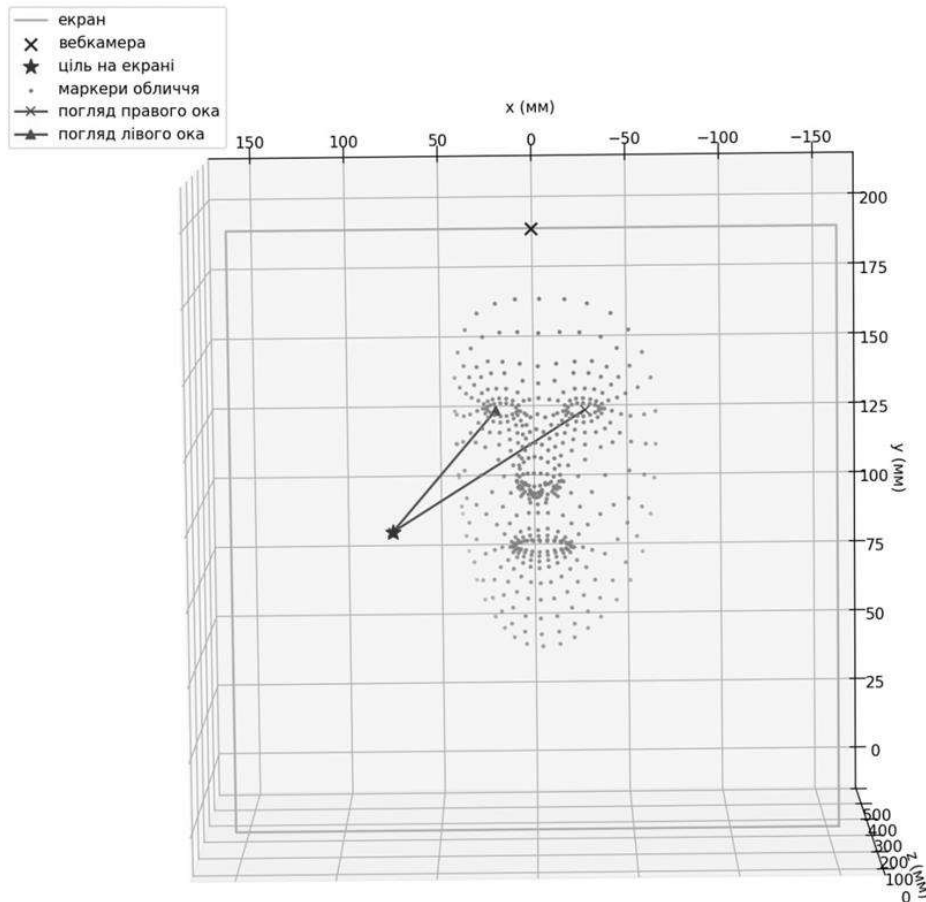


Рис. 4. Тривимірне зображення користувача з вектором погляду та точкою його перетину з площиною екрана

Площину екрана задано в системі координат камери – зазвичай вона розміщується перпендикулярно до осі Z на визначеній відстані. Розраховується точка перетину вектора погляду з цією площиною, яка і є прогнозованою точкою фіксації. У візуалізації ця точка може бути позначена кольоровою міткою (наприклад, червоною), що дозволяє візуально оцінити, куди, на думку системи, спрямовано погляд користувача.

Такий тип тривимірної візуалізації є корисним інструментом не лише для оцінки якості прогнозу, а й для калібрування та виявлення потенційних помилок у розрахунках – наприклад, коли напрямок погляду не збігається з положенням мішені. Це також дозволяє розробнику контролювати узгодженість між геометрією обличчя, положенням голови та вектором погляду.

Отримані результати підтверджують доцільність використання фільтрації при роботі з координатами погляду, особливо у випадках високочастотних шумів. Крім того, результати показують, що навіть без повної реалізації складних моделей у коді, теоретичне моделювання дає можливість оцінити ефективність обраного підходу та обґрунтувати його застосування в реальній системі.

6. Обговорення результатів дослідження

Результати дослідження показали покращену стабільність і точність керування курсором за допомогою відстеження погляду. На відміну від попередніх підходів, які здебільшого зосереджувались на визначенні координат погляду, у цьому дослідженні застосовано методи фільтрації шуму та прогнозування руху очей. Це дозволяє зменшити

випадкові коливання і забезпечити плавніше керування. Такий підхід покращує взаємодію користувача із системою і зменшує втому очей при тривалому використанні.

Серед основних переваг цієї системи — підвищена точність позиціонування курсора та зниження навантаження на процесор завдяки оптимізованим алгоритмам обробки даних. Інтеграція механізму підтвердження команд, наприклад подвійного кліку, також спрощує взаємодію з системою, дозволяючи обійтись без додаткових пристроїв введення. Важливою особливістю є можливість адаптації алгоритмів до індивідуальних потреб користувачів, що розширює потенційні сфери застосування.

Проте, окрім переваг, система має певні обмеження. Її ефективність залежить від якості відеопотоку та умов освітлення. Незважаючи на алгоритми компенсації, різкі зміни освітлення можуть вплинути на точність визначення погляду. Крім того, система може виявляти деяку нестабільність при дуже швидких рухах очей, що може спричинити незначні затримки в переміщенні курсора.

Подальші дослідження можуть зосередитись на покращенні адаптивності алгоритмів, зокрема за допомогою методів глибокого навчання для прогнозування руху очей в реальному часі. Окрім цього, є потенціал для розширення функціональності системи через інтеграцію жестового управління або комбінованих методів введення. Можна також дослідити ефективність цього підходу у різних сценаріях, таких як управління віртуальними середовищами або підтримка людей з обмеженими можливостями. Отже, отримані результати відкривають можливості для подальших удосконалень, які зроблять систему ще точнішою, швидшою та зручнішою в експлуатації.

7. Висновки

У процесі дослідження було послідовно вирішено поставлені задачі, спрямовані на реалізацію системи керування курсором за допомогою погляду.

Визначено модульну структуру системи керування курсором за поглядом з окремими компонентами для відеозахоплення, обробки зображень, прогнозування погляду, обчислення координат та виводу результатів. Вибрано відповідні інструменти, такі як MediaPipe, PyTorch, OpenCV.

Реалізовано ключові компоненти системи керування курсором за поглядом:

- реалізовано нейронну мережу на основі згорткових шарів і повнозв'язних блоків для прогнозування напрямку погляду у вигляді кутів pitch та yaw, що забезпечило базову оцінку фокусу користувача;
- впроваджено процес нормалізації вхідних зображень очей та обличчя, включаючи вирівнювання по положенню голови, що підвищило точність роботи моделі;
- розроблено алгоритм перетворення кутових координат у просторовий вектор з подальшою проекцією на площину екрана, що дозволяє адаптувати систему до різного положення камери та розміру дисплея;
- досліджено та реалізовано два методи фільтрації координат погляду – ковзне середнє та фільтр Калмана, що дозволило суттєво зменшити випадкові коливання та підвищити стабільність положення курсора;
- запропоновано механізм підтвердження введення команд (подвійне кліпання або фіксація погляду), що забезпечив зручність взаємодії з користувачем без використання фізичних пристроїв введення.

Проведено тестування системи в умовах реального використання. Отримані результати засвідчили прийнятний рівень затримки (до 50 мс), високу адаптивність до різних користувачів і можливість точного керування курсором без апаратних модифікацій.

Таким чином, усі задачі дослідження виконано. Отримана система є програмно гнучкою, не залежить від спеціального обладнання та може бути легко модифікована.

Подальші дослідження можуть включати вдосконалення прогнозу траєкторії погляду, врахування жестів голови або інтеграцію з розширеними інтерфейсами користувача.

Перелік посилань:

1. Zhang Z., Chen Y., Li L. Gaze Tracking Techniques and Applications. Springer, 2019. 320 p.
2. Hennadiy F., Zakharchenko A. Machine Learning Algorithms for Eye Gaze Detection. *Journal of Computational Intelligence*. 2021. Vol. 16, No. 3. P. 189–202.
3. Viola P., Jones M. Rapid Object Detection using a Boosted Cascade of Simple Features. CVPR, 2001. P. 511–518.
4. Cheng Y., Liu W., Lv F. Real-Time Gaze Estimation Based on CNN and Facial Landmark Detection Sensors. 2020. Vol. 20, No. 4. P. 1–15.
5. Welch G., Bishop G. An Introduction to the Kalman Filter. UNC Chapel Hill. 1995. 16 p
6. Васильєва І., Коваленко Ю. Техніки фільтрації шуму в системах відстеження погляду. Харків: Наукова думка, 2020. 192 с.
7. Papoutsaki A., Sadeh N., Huang J., Lasecki W.S., Bigham J.P. WebGazer: Scalable Webcam Eye Tracking Using User Interactions. IJHCS. 2016. Vol. 98. P. 102–113.
8. Tobii Eye Tracking. URL: <https://www.tobii.com/> (дата звернення: 26.04.2025).
9. Медіапайп (MediaPipe). Розробка моделей для відстеження обличчя. URL: https://google.github.io/mediapipe/solutions/face_mesh (дата звернення: 26.04.2025).
10. Мартин Х., Джонс Р. Комп'ютерний зір та глибоке навчання для відстеження погляду. Київ: Наукова думка, 2021. 312 с.
11. Офіційний сайт OpenCV: Бібліотека для комп'ютерного зору та обробки зображень. URL: <https://opencv.org/> (дата звернення: 26.04.2025).
12. Офіційний сайт PyTorch: Бібліотека глибокого навчання. URL: <https://pytorch.org/> (дата звернення: 26.04.2025).

Надійшла до редколегії 10.05.2025 р.

Ситнікова Поліна Едуардівна, кандидат технічних наук, доцент, доцент кафедри системотехніки ХНУРЕ, м. Харків, Україна, e-mail: polina.sytnikova@nure.ua, ORCID: <https://orcid.org/0000-0002-6688-4641>

Грицай Микита Олегович, здобувач вищої освіти, група СПРМ-23-1, факультет комп'ютерних наук, ХНУРЕ, м. Харків, Україна, e-mail: mykyta.hrytsai@nure.ua