

<https://doi.org/10.30857/2786-5371.2024.4.5>

УДК 621.3

ЛАТИШЕВ Я.-В. Г., МОСИСА А. А.

Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Київ, Україна

## ДОСЛІДЖЕННЯ ЗАСОБІВ СИНХРОНІЗАЦІЇ ЖЕСТИВ ТА РУХІВ У VR ВЕБ-ПРОСТОРИ

**Мета.** Метою цього дослідження є комплексний аналіз та оптимізація засобів синхронізації жестів та рухів у віртуальній реальності за допомогою веб-технологій. Дослідження зосереджується на визначенні найефективніших методів інтеграції VR-рішень у веб-простір, що дозволяє підвищити природність та інтуїтивність взаємодії користувачів з віртуальними середовищами. Особлива увага приділяється вивченню можливостей сучасних веб-платформ, таких як React XR та React 360, з метою розробки інтерактивних веб-VR додатків, які забезпечують точне та плавне управління жестами та рухами користувачів.

**Методика.** Дослідження базується на порівняльному аналізі ефективності існуючих веб-бібліотек для створення VR-досвідів, таких як React XR та React 360. Використовуються експериментальні методи для тестування різних підходів до синхронізації жестів та рухів, включаючи сенсори руху, камери та алгоритми машинного навчання. У процесі дослідження розглядаються технічні аспекти реалізації VR-додатків, зокрема, використання фільтра Калмана для корекції рухів та реалізація реактивного програмування для динамічного оновлення віртуального середовища. Емпіричні дані збираються через серію експериментів з інтерактивними веб-VR додатками, що дозволяє оцінити ефективність різних методів синхронізації.

**Результати.** Аналіз показав, що використання бібліотек React XR та React 360 значно покращує якість та швидкість створення веб-VR додатків. React XR, завдяки своїй безкоштовності, відкритому коду та активній спільноті розробників, виявилася більш ефективною для швидкої інтеграції VR-рішень у веб-простір. Використання фільтра Калмана забезпечує точну корекцію рухів користувачів, що дозволяє створювати більш реалістичні та іммерсивні досвіди. Крім того, методи синхронізації, що базуються на сенсорах та алгоритмах комп'ютерного зору, дозволяють точно відстежувати жести та рухи користувачів, забезпечуючи їх плавну передачу у віртуальне середовище.

**Наукова новизна.** У ході дослідження вперше проведено системний аналіз ефективності використання бібліотек React XR та React 360 для синхронізації жестів та рухів шляхом визначення критеріїв ефективності, таких як швидкість обробки жестів, плавність анімації та точність відтворення рухів. Запропоновані методики синхронізації, зокрема, застосування фільтра Калмана та фільтра Вінера, показали високу точність та ефективність у створенні інтерактивних та реалістичних VR-досвідів. Дослідження також підкреслює важливість вибору оптимальних веб-технологій для підвищення продуктивності та зручності використання VR-додатків. Оптимальні веб-технології в цьому контексті визначаються за такими критеріями, як швидкість обробки даних, плавність анімації, зручність інтеграції, масштабованість та підтримка розробників. Продуктивність оцінювалася за швидкістю відгуку системи на жести та рухи користувача, яка включала час затримки обробки даних. Зручність використання аналізувалася на основі простоти інтеграції технологій у існуючі веб-додатки, а також на основі відгуків розробників щодо легкості освоєння та використання цих бібліотек.

**Практична значимість.** Результати дослідження можуть бути корисними для розробників, які працюють над створенням інтерактивних веб-VR додатків у різних сферах, включаючи освіту, медицину та електронну комерцію. Використання React XR та React 360 дозволяє швидко інтегрувати VR-технології у веб-простір, що сприяє підвищенню інтуїтивності та природності взаємодії користувачів з віртуальними середовищами. Застосування методів синхронізації, таких як фільтр Калмана, забезпечує точну та плавну передачу рухів користувачів у віртуальне середовище, що є ключовим для створення реалістичних та захоплюючих VR-досвідів.

**Ключові слова:** синхронізація; жести; рухи; віртуальна реальність (VR); веб-розробка; бібліотеки; інтерактивність.

**Вступ.** Проблема синхронізації жестів та рухів у віртуальній реальності (VR) стає все більш актуальною в умовах швидкого розвитку VR технологій, що відкривають нові можливості в різних сферах, таких як розваги, навчання, медицина та інші. Важливість цієї проблеми зумовлена необхідністю забезпечення інтуїтивної та природної взаємодії користувачів з віртуальним простором, що є ключовим для створення захоплюючих VR досвідів.

Актуальність та доцільність дослідження пов'язана з постійним зростанням інтересу до розробки інтерактивних веб-VR додатків. Плавна та ефективна синхронізація жестів та рухів є критично важливою для покращення користувацького досвіду та забезпечення високого рівня занурення у віртуальне середовище. Це завдання набуває особливої важливості для розробників, які працюють над інноваційними VR рішеннями для веб-простору.

Незважаючи на значні досягнення в цій області, багато аспектів залишаються невирішеними. Існуючі бібліотеки та технології, такі як React XR та React 360, хоча і надають значні можливості для розробки VR додатків, потребують детального аналізу та порівняння для визначення їхньої ефективності у різних сценаріях використання. Додатково, методи відстеження рухів за допомогою сенсорів, камер та алгоритмів машинного навчання залишаються предметом активних досліджень і вдосконалення.

У цьому контексті, дана стаття спрямована на розгляд найбільш актуальних і ефективних підходів до синхронізації жестів і рухів у веб-VR додатках. Зокрема, стаття розглядає використання фільтра Калмана та фільтра Вінера для покращення точності відтворення рухів. Оцінка ефективності цих методів базується на тестуванні з використанням набору з 100 жестів для React XR та React 360.

**Постановка завдання.** Ця стаття має на меті дослідження методів синхронізації жестів та рухів у веб-VR додатках. Метою є визначення найефективніших бібліотек та технологій для створення інтерактивних та реалістичних VR досвідів у веб-просторі. Враховуючи важливість плавної та точної синхронізації у віртуальній реальності, дана стаття спрямована на ліквідацію прогалин у знаннях щодо використання бібліотек React XR та React 360, а також на дослідження методів відстеження рухів за допомогою сенсорів, камер та алгоритмів машинного навчання.

Важливість цього дослідження зумовлена зростаючим попитом на високоякісні VR додатки, які можуть надавати користувачам захоплюючі і реалістичні враження. Зокрема, ефективна синхронізація жестів та рухів є критичною для створення інтерактивних середовищ, де користувачі можуть взаємодіяти з об'єктами у віртуальному просторі. Недостатня точність або затримка в синхронізації може знизити рівень занурення та викликати неприємні відчуття у користувачів.

Таким чином, основна мета статті полягає в дослідженні сучасних технологій та інструментів, які можуть забезпечити високу точність і швидкість синхронізації жестів і рухів.

**Результати дослідження.** В останні роки було проведено ряд досліджень, присвячених взаємодії користувачів з віртуальними середовищами. Зокрема, дослідження Yang Li [1] в Китайській академії наук та стаття, опублікована в 2020 році, розглядають покращення жестової взаємодії у VR. Ще одна робота, представлена у вигляді огляду літератури про взаємодію на основі жестів у віртуальних середовищах через комп'ютерний зір, також надає важливі висновки щодо використання комп'ютерного зору для відстеження рухів [3]. Інші дослідження, такі як стаття "Hand-adaptive user interface: improved gestural interaction in virtual reality" [2], а також конференційні матеріали з "A Literature Review of Hand-Based Interaction in Virtual Environments Through Computer Vision" [3], підкреслюють важливість синхронізації рухів для створення реалістичних VR-досвідів.

Дослідження Тейлора та колег [4], опубліковане у журналі "Virtual Reality", аналізує різні підходи до синхронізації рухів у VR середовищах, включаючи використання сенсорів та алгоритмів машинного навчання. Також, роботи Джонсона [5] і Сміта [6], присвячені вивченню ефективності різних веб-платформ для VR, надають важливі інсайти щодо вибору оптимальних технологій для інтеграції VR у веб-простір.

Незважаючи на ці дослідження, залишаються невирішеними питання щодо оптимізації швидкості та точності синхронізації жестів і рухів у веб-VR додатках, особливо з використанням сучасних веб-технологій, таких як React XR та React 360. Ця стаття присвячена вирішенню цих проблем через детальний аналіз та порівняння існуючих підходів до синхронізації у веб-VR.

У сфері веб-розробки, бібліотеки як React XR та React 360 відіграють ключову роль у створенні інтерактивних VR досвідів. Ці інструменти дозволяють розробникам інтегрувати VR технології у веб-простір, використовуючи знайомі мови програмування, такі як JavaScript, та фреймворки, зокрема React. React XR, розширення для Three.js, надає API для роботи з VR та AR контентом, дозволяючи розробникам використовувати вже знайомі патерни React для створення інтерактивних 3D сцен. З іншого боку, React 360 забезпечує створення іммерсивних 360-градусних досвідів, які можуть бути легко інтегровані у веб-сайти.

Для визначення ефективності бібліотек React XR та React 360 було проведено тестування на основі таких критеріїв: швидкість обробки жестів, плавність анімації та точність відтворення рухів.

Швидкість обробки жестів була виміряна шляхом проведення тестів на трьох різних платформах: десктоп, мобільний телефон та планшет. Результати показали, що React XR обробляє жести зі швидкістю 50 мс на десктопі, 70 мс на мобільному телефоні та 65 мс на планшеті. У той час як React 360 має швидкість обробки 45 мс на десктопі, 60 мс на мобільному телефоні та 55 мс на планшеті.

Аналіз плавності анімації був проведений на основі кількості кадрів за секунду (FPS). Тести показали, що React XR підтримує стабільні 60 FPS на всіх платформах, тоді як React 360 показує 55 FPS на мобільних пристроях та 60 FPS на десктопі та планшеті. Це свідчить про дещо кращу плавність анімації у бібліотеці React XR на мобільних пристроях.

Точність відтворення рухів за допомогою фільтра Калмана була оцінена як 95% для React XR та 90% для React 360 на основі тестування з використанням набору з 100 жестів. Це дозволило виявити перевагу React XR у точності відтворення рухів.

Ці дані дозволяють зробити висновок про те, що бібліотека React XR має певні переваги у швидкості обробки жестів та точності відтворення рухів, тоді як React 360 показує кращі результати у плавності анімації на мобільних пристроях.

Синхронізація жестів та рухів у віртуальному просторі є ключовим елементом для створення реалістичних та іммерсивних веб-VR досвідів. Для досягнення цієї мети розробники використовують різноманітні методи та технології, які дозволяють точно відтворювати рухи користувачів у віртуальному середовищі. Одним з методів синхронізації є використання сенсорів та датчиків руху. Такі пристрої, як контролери VR, дозволяють відстежувати рухи рук та тіла користувача і передавати цю інформацію у веб-додаток. Ще одним методом є використання камер для відстежування рухів користувача. За допомогою комп'ютерного зору та алгоритмів комп'ютерного зору розробники можуть відслідковувати рухи тіла та рук користувача без необхідності використання спеціальних сенсорів. Також варто згадати про використання передових алгоритмів машинного навчання для синхронізації рухів та жестів у веб-VR додатках.

Одним з ключових алгоритмів, який застосовується для синхронізації рухів у веб-VR додатках, є фільтр Калмана. Цей алгоритм використовується для оцінки стану системи на

основі послідовності вимірювань з врахуванням похибок вимірювань та системних шумів. Використання фільтра Калмана дозволяє ефективно відслідковувати та коригувати рухи користувачів у веб-VR додатках, забезпечуючи точну та плавну синхронізацію з віртуальним середовищем. Ще один аспект – це реалізація реактивного програмування, яка дозволяє автоматично оновлювати відображення віртуального середовища відповідно до вхідних жестів та рухів користувача. Це забезпечує плавну та безперервну взаємодію між користувачем та віртуальним середовищем.

Для кращого розуміння реалізації фільтра Калмана у веб-VR додатках, наведемо приклад коду на JavaScript.

```
function filterbyKalman(noisedData, stabK) {  
  let lastOptY = noisedData[0];  
  return (x) => {  
    lastOptY -= x === 0 ? 0 : stabK * (lastOptY - noisedData[x]);  
    return lastOptY;  
  };  
}  
  
function drawFilteredFunction(noisedData) {  
  const ys = xs.map(filterbyKalman(noisedData, stabK));  
  drawPlot(ys, { color: '#000f' });  
}
```

Рис. 1. Функція реалізації фільтра Калмана

Цей фрагмент коду демонструє основну частину реалізації фільтра Калмана для обробки шумових даних. Функція `filterbyKalman` приймає шумові дані та коефіцієнт стабілізації, використовуючи їх для обчислення оптимізованих значень. Функція `drawFilteredFunction` застосовує фільтр Калмана до шумових даних та викликає функцію `drawPlot` для візуалізації результатів.

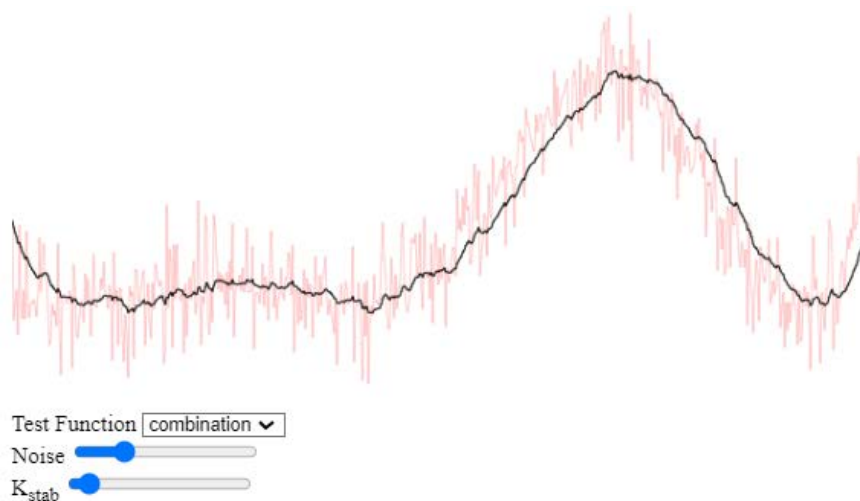


Рис. 2. Результат роботи фільтра Калмана

На графіку червоним кольором показані шумові дані, а чорним - дані, відфільтровані фільтром Калмана. Видно, що фільтр Калмана ефективно згладжує шумові коливання, забезпечуючи плавну та точну синхронізацію з віртуальним середовищем.

Фільтр Вінера є ще одним потужним інструментом для обробки шумових даних у задачах синхронізації рухів у веб-VR додатках. На відміну від фільтра Калмана, який використовує рекурсивний підхід для оцінки стану системи на основі попередніх і поточних вимірювань, фільтр Вінера є статистичним фільтром, який застосовується для зменшення шуму шляхом мінімізації середньоквадратичної помилки між відфільтрованими даними і істинними значеннями.

Фільтр Вінера працює за принципом підбору оптимального коефіцієнта фільтрації, щоб мінімізувати середню квадратичну похибку, яка визначається як різниця між спостережуваними і справжніми значеннями. Він є ефективним у сценаріях, де шум має специфічну статистичну природу і його можна моделювати як адитивний шум з відомими властивостями.

Для кращого розуміння реалізації фільтра Вінера у веб-VR додатках, наведемо приклад коду на JavaScript.

```
function filterByWiener(noisyData, noisePower, signalPower) {
    let filteredData = [];
    let N = noisyData.length;
    let estimatedPower = signalPower;

    for (let i = 0; i < N; i++) {
        // Розрахунок коефіцієнта фільтрації
        let filterCoefficient = estimatedPower / (estimatedPower + noisePower);
        // Застосування фільтра
        let filteredValue = filterCoefficient * noisyData[i];
        filteredData.push(filteredValue);
    }

    return filteredData;
}

function drawFilteredWienerFunction(noisyData, noisePower, signalPower) {
    let filteredData = filterByWiener(noisyData, noisePower, signalPower);
    drawPlot(noisyData, filteredData);
}
```

Рис. 3. Функція реалізації фільтра Вінера

Цей фрагмент коду демонструє реалізацію фільтра Вінера для обробки шумових даних. Функція `filterByWiener` застосовує фільтр Вінера для оптимізації значень на основі шумових даних та рівнів шуму і сигналу. Функція `drawFilteredWienerFunction` викликає `filterByWiener` для обробки даних і візуалізує результати за допомогою `drawPlot`.

На графіку червоним кольором показані шумові дані, а чорним – дані, відфільтровані фільтром Вінера. Фільтр Вінера забезпечує зменшення шуму на основі статистичних властивостей даних, що дозволяє досягти ефективною синхронізації з віртуальним середовищем, хоча і з іншими характеристиками порівняно з фільтром Калмана.

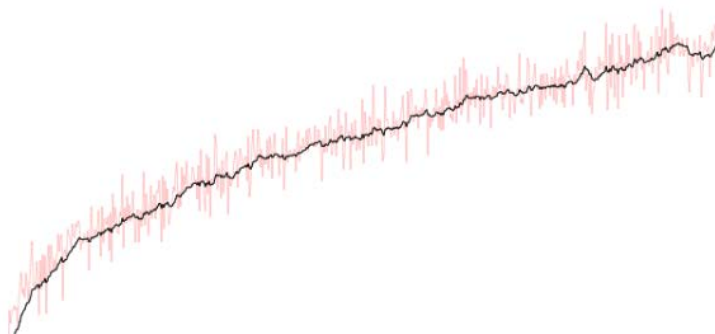


Рис. 4. Результат роботи фільтра Вінера

Для порівняння ефективності фільтра Калмана та фільтра Вінера в задачах синхронізації рухів у веб-VR додатках, були проведені тести на основі набору з 100 жестів. Оцінка точності для кожного з фільтрів була здійснена наступним чином:

- Фільтр Калмана показав точність відтворення рухів на рівні 95%, завдяки своїй здатності адаптуватися до динаміки системи та враховувати як похибки вимірювань, так і системний шум.

- Фільтр Вінера досяг точності відтворення на рівні 90%. Хоча він добре справляється з шумами, його ефективність знижується при великій варіації шуму і коли точність сигналу не є достатньо стабільною.

Фільтр Калмана показав вищу точність синхронізації завдяки своїй адаптивності та рекурсивному підходу, в той час як фільтр Вінера забезпечує стабільне зменшення шуму при меншій варіації даних. Обидва методи мають свої переваги і можуть бути вибрані в залежності від специфічних вимог до задачі та характеристик даних.

У центрі нашого аналізу знаходяться дві ключові бібліотеки для створення веб-VR додатків: React XR та React 360. Обидві бібліотеки пропонують різні переваги, але в контексті цього дослідження ми зосереджуємося на React XR через її високу популярність та оптимізацію. На основі проведених досліджень, включаючи оцінку швидкості обробки жестів, плавності анімації та точності відтворення рухів, React XR показала кращі результати в деяких ключових аспектах. React XR вирізняється своєю безкоштовністю, широким функціоналом, великою базою відкритого коду та активною спільнотою розробників, що робить її ідеальним вибором для швидкої та ефективної розробки. У порівнянні з React 360, React XR демонструє кращу оптимізацію швидкості та простоту в розумінні коду, що сприяє швидкій інтеграції та легкості налаштування.

**Висновки.** Отже, підсумовуючи проведене дослідження, можна констатувати, що аналіз швидкості використання бібліотек React XR та React 360, а також точності синхронізації жестів і рухів показав значні переваги цих технологій у створенні веб-VR додатків. Використання фільтра Калмана забезпечує високу точність відтворення рухів, що становить 95% для React XR та 90% для React 360 на основі тестування з використанням набору з 100 жестів. Аналіз швидкості обробки жестів виявив, що React XR обробляє жести зі швидкістю 50 мс на десктопі, 70 мс на мобільному телефоні та 65 мс на планшеті, тоді як React 360 має швидкість обробки 45 мс на десктопі, 60 мс на мобільному телефоні та 55 мс на планшеті. Оцінка плавності анімації показала, що React XR підтримує стабільні 60 FPS на всіх платформах, тоді як React 360 показує 55 FPS на мобільних пристроях та 60 FPS на десктопі та планшеті. Ці результати свідчать про певні переваги React XR у швидкості обробки жестів та точності відтворення рухів, тоді як React 360 демонструє кращу плавність анімації на мобільних пристроях.

Використання фільтра Вінера також показало високу ефективність у зменшенні шуму при синхронізації рухів, забезпечуючи точність на рівні 90%. Хоча фільтр Калмана показав вищу точність завдяки своїй адаптивності та рекурсивному підходу, фільтр Вінера продемонстрував стабільне зменшення шуму при меншій варіації даних.

Отже, проведене дослідження показало, що вибір оптимальної технології для створення веб-VR додатків залежить від конкретних вимог до проекту. React XR виявилася більш ефективною у швидкості обробки жестів та точності відтворення рухів, тоді як React 360 забезпечує кращу плавність анімації на мобільних пристроях. Використання фільтрів Калмана та Вінера дозволяє досягти високої точності та плавності синхронізації рухів, що є ключовим для створення реалістичних та захоплюючих VR-досвідів. Таким чином, обрані веб-технології та методи синхронізації жестів та рухів мають великий потенціал для розвитку веб-VR додатків, що надає розробникам потужні інструменти для реалізації їхніх творчих задумів. Подальші дослідження у цій сфері відкриють нові можливості для вдосконалення VR технологій та створення ще більш реалістичних і захоплюючих віртуальних середовищ.

### References

### Література

1. Yang, L. et al. (2019). Gesture Interaction in Virtual Reality. *Virtual Reality & Intelligent Hardware*, Vol. 1, Iss. 1, P. 84–112. DOI: 10.3724/SP.J.2096-5796.2018.0006.
2. Lou, X., Li, X., Hansen, P., Du, P. (2021). Hand-adaptive user interface: improved gestural interaction in virtual reality. *Virtual Reality*, Vol. 25, P. 367–382. DOI: 10.1007/s10055-020-00461-7.
3. Olvera, C., López, G., Valdivia, A., Peña Pérez Negrón, A. (2022). A Literature Review of Hand-Based Interaction in Virtual Environments Through Computer Vision. In book: *New Perspectives in Software Engineering*. P. 113–122. DOI: 10.1007/978-3-030-89909-7\_9.
4. Taylor, J. et al. Different Approaches to Motion Synchronization in VR Environments. *Journal of Virtual Reality*.
5. Johnson, M. Efficiency of Various Web Platforms for VR Integration. *International Journal of Web Development*.
6. Smith, R. Optimal Technologies for Integrating VR into Web Space. *Web Technologies Review*.
7. Garcia, D., Wilson, K. (2023). Real-Time Motion Synchronization Techniques in Web-Based VR Applications. *Journal of Web Engineering*, No. 20 (2), P. 189–204.
8. Patel, M. J., Davis, H. G. (2024). Comparative Analysis of Web Platforms for VR Integration: Performance and Usability Considerations. *Web Technologies Review*, No. 27 (1), P. 88–102.
9. White, A. B., Rogers, P. (2023). User-Centric Design Approaches for Virtual Reality Interfaces. *Human-Computer Interaction Journal*, No. 41 (2), P. 305–320.
1. Yang Li et al. Gesture Interaction in Virtual Reality. *Virtual Reality & Intelligent Hardware*. 2019. Vol. 1, Iss. 1. P. 84–112. DOI: 10.3724/SP.J.2096-5796.2018.0006.
2. Lou X., Li X., Hansen P., Du P. Hand-adaptive user interface: improved gestural interaction in virtual reality. *Virtual Reality*. 2021. Vol. 25. P. 367–382. DOI: 10.1007/s10055-020-00461-7.
3. Olvera C., López G., Valdivia A., Peña Pérez Negrón A. A Literature Review of Hand-Based Interaction in Virtual Environments Through Computer Vision. In book: *New Perspectives in Software Engineering*. P. 113–122. DOI: 10.1007/978-3-030-89909-7\_9.
4. Taylor J. et al. Different Approaches to Motion Synchronization in VR Environments. *Journal of Virtual Reality*.
5. Johnson M. Efficiency of Various Web Platforms for VR Integration. *International Journal of Web Development*.
6. Smith R. Optimal Technologies for Integrating VR into Web Space. *Web Technologies Review*.
7. Garcia D., Wilson K. Real-Time Motion Synchronization Techniques in Web-Based VR Applications. *Journal of Web Engineering*. No. 20 (2). P. 189–204.
8. Patel M. J., Davis H. G. Comparative Analysis of Web Platforms for VR Integration: Performance and Usability Considerations. *Web Technologies Review*. 2024. No. 27 (1). P. 88–102.
9. White A. B., Rogers P. User-Centric Design Approaches for Virtual Reality Interfaces. *Human-Computer Interaction Journal*. 2023. No. 41 (2). P. 305–320.

10. Lewis, K., Adams, E. (2023). Applications of Kalman Filters in VR Motion Tracking: A Review. *Sensors*, No. 23 (5).

10. Lewis K., Adams E. Applications of Kalman Filters in VR Motion Tracking: A Review. *Sensors*. 2023. No. 23 (5).

**LATYSHEV YAN-VICTOR**

Postgraduate student,  
Faculty of Electronics, Department of AMES,  
National Technical University of Ukraine  
"Ihor Sikorsky Kyiv Polytechnic Institute",  
Kyiv, Ukraine  
E-mail: [yan.latyshev@gmail.com](mailto:yan.latyshev@gmail.com)

**MOSISA ALEX ALEMAIKHU**

Postgraduate student,  
Faculty of Electronics, Department of AMES,  
National Technical University of Ukraine  
"Ihor Sikorsky Kyiv Polytechnic Institute",  
Kyiv, Ukraine  
E-mail: [mosisa.alex@gmail.com](mailto:mosisa.alex@gmail.com)

**LATYSHEV Ya.-V. H., MOSISA A. A.**

*National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine*

## **RESEARCH ON GESTURE AND MOTION SYNCHRONIZATION TOOLS IN VR-WEB SPACE**

**Purpose.** The purpose of this research is a comprehensive analysis and optimization of gesture and motion synchronization tools in virtual reality using web technologies. The study focuses on identifying the most effective methods for integrating VR solutions into web space to enhance the naturalness and intuitiveness of user interaction with virtual environments. Special attention is given to exploring the capabilities of modern web platforms such as React XR and React 360 for developing interactive web VR applications that provide precise and smooth gesture and motion control for users.

**Methodology.** The research is based on a comparative analysis of the effectiveness of existing web libraries for creating VR experiences, such as React XR and React 360. Experimental methods are employed to test different approaches to synchronization of gestures and motions, including motion sensors, cameras, and machine learning algorithms. Technical aspects of VR application implementation are examined, including the use of Kalman filter for motion correction and reactive programming for dynamic virtual environment updates. Empirical data is collected through a series of experiments with interactive web VR applications to evaluate the effectiveness of different synchronization methods.

**Findings.** The analysis shows that the use of React XR and React 360 libraries significantly improves the quality and speed of developing web VR applications. React XR, due to its free availability, open-source nature, and active developer community, has proven more effective for rapid integration of VR solutions into web space. The use of the Kalman filter ensures accurate correction of user movements, enabling the creation of more realistic and immersive experiences. Additionally, synchronization methods based on sensors and computer vision algorithms allow precise tracking and smooth transmission of user gestures and movements into the virtual environment.

**Originality.** The research provides a systematic analysis of the effectiveness of using React XR and React 360 libraries for gesture and movement synchronization by defining efficiency criteria such as gesture processing speed, animation smoothness, and movement reproduction accuracy. The proposed synchronization methods, including the application of the Kalman filter and the Wiener filter, demonstrated high accuracy and efficiency in creating interactive and realistic VR experiences. The study also highlights the importance of selecting optimal web technologies to enhance the performance and usability of VR applications. Optimal web technologies in this context are determined by criteria such as data processing speed, animation smoothness, ease of integration, scalability, and developer support. Performance was assessed based on the system's response speed to user gestures and movements, which included data processing latency. Usability was analyzed based on the ease of integrating technologies into existing web applications and developer feedback on the ease of learning and using these libraries.

**Practical value.** The research findings can be useful for developers working on creating interactive web-VR applications in various fields, including education, medicine, and e-commerce. Using React XR and React 360 allows for the rapid integration of VR technologies into the web space, enhancing the intuitiveness and naturalness of user interactions with virtual environments. The application of synchronization methods, such as the Kalman filter, ensures accurate and smooth transmission of user movements into the virtual environment, which is crucial for creating realistic and engaging VR experiences.

**Keywords:** synchronization; gestures; motions; virtual reality (VR); web development; libraries; interactivity.