

УДК 004.5, 004.92

MSC 65D18

ALGORITHMIC AND GRAPHICAL TOOLS FOR PRE-PROCESSING OF HOLTER DATA

A. V. LIASHKO, M. S. YEFREMOV

Faculty of Computer Science and Cybernetics, Taras Shevchenko National University of Kyiv,
Kyiv, Ukraine, E-mail: andrey_liashko@knu.ua, yefremov@knu.ua

АЛГОРИТМІЧНО-ГРАФІЧНІ ЗАСОБИ ПОПЕРЕДНЬОЇ КОМП'ЮТЕРНОЇ ОБРОБКИ ДАНИХ ХОЛТЕРА

А. В. ЛЯШКО, М. С. ЄФРЕМОВ

Факультет комп'ютерних наук та кібернетики, Київський національний університет
імені Тараса Шевченка, Київ, Україна,
E-mail: andrey_liashko@knu.ua, yefremov@knu.ua

АБСТРАКТ. The work is dedicated to exploring methods of detecting cardiovascular diseases using Holter ECG analysis. An overview of the interface and capabilities of the program is conducted, and an analysis of the threshold algorithm for detecting potentially suspicious intervals is performed.

KEYWORDS: Holter ECG, R-peaks and R-R intervals, marking of electrocardiogram data, pre-processing and post-processing of electrocardiogram data.

АНОТАЦІЯ. В роботі наведено опис програмного комплексу з візуалізації даних електрокардіограми отриманих з датчика Холтера. Наведено огляд графічного інтерфейсу та можливостей програмного забезпечення, розглянуто роботу порогового алгоритму для виявлення R-піків.

КЛЮЧОВІ СЛОВА: ЕКГ за Холтером, R-піки та R-R інтервали, розмітка даних електрокардіограми, препроцесорна та постпроцесорна обробка даних електрокардіограми.

1. ВСТУП

Захворювання серцево-судинної системи є однією з наймасовіших причин смертності у всьому світі. Якщо розглянути ситуацію в Україні, то смертність від серцево-судинних захворювань посідає перше місце та складає 64.3% від загальної кількості хворих [1]. З кожним роком середній вік хворих стабільно зменшується, що вказує на невтішну тенденцію, а кількість чинників серцево-судинних захворювань та їхній вплив тільки зростає. Одним з найефективніших видів неінвазивних методів дослідження

хвороб серцево-судинної системи є електрокардіограма (ЕКГ) за Холтером [2]. Прилад Холтера дозволяє виявляти хворобу на початковій стадії, та може виявити симптоми, які можуть проявлятися лише у вузькому часовому діапазоні або під звичним навантаженням, саме тому виникає необхідність у створенні програмних засобів, які дозволяють ефективно виявляти ознаки хвороби серцево-судинної системи, для правильної постановки діагнозу, та призначення ефективного лікування.

Об'єктом дослідження є аналіз наявних методів виявлення серцево-судинних захворювань за допомогою ЕКГ за Холтером.

Предметом дослідження є розробка програмного комплексу для маніпулювання даними ЕКГ з подальшим аналізом.

Метою дослідження є програмний комплекс, який надасть можливості візуалізації, розмітки даних ЕКГ для виділення ефективних характеристичних ознак.

Збір даних ЕКГ за Холтером відбувається здебільшого протягом тривалого проміжку часу (24 години та більше). Це дозволяє виявляти порушення в роботі серця: епізоди ішемії міокарда, порушення ритму і провідності серця тощо. Однак дані записані цим приладом мають великий обсяг і вимагають спеціальних методів їх обробки, що вимагає використання значного ресурсу комп'ютерної техніки [3]. З огляду на це, необхідно мати ефективні алгоритми для попередньої обробки даних та подальшого аналізу. Розробка алгоритмів та програмного забезпечення автоматизованого аналізу показників роботи серця дозволить лікарям зменшити час обробки даних ЕКГ. Обробку даних можна виконувати як під час їх отримання, так і після завершення збору даних серцевої діяльності, що може пришвидшити встановлення діагнозу.

В першому розділі даної публікації наведено опис програмного забезпечення, його характеристик та його окремих складових. В другому розділі проведено короткий огляд інтерфейсу програмного забезпечення, розглянуто його функціонал на прикладі реалізованого порогового алгоритму для виділення R-піків.

2. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Дані окремих відведень (дані серцебиття в певний момент часу, що вимірюються в мілівольтах) отримуються за певний проміжок часу, які найчастіше збираються з 3 датчиків, протягом однієї або кількох діб. Точність вимірювань визначаються кількістю записаних значень за певну одиницю часу, саме тому вихідні файли Холтера займають достатньо багато пам'яті. В більшості випадків дані записуються протягом доби, за трьома каналами, кожен з яких реєструє 250-500 вимірювань за секунду, що становить близько сто мільйонів значень, або 100-150 мегабайт даних. Більшість приладів Холтера оснащені програмним забезпеченням, що дозволяє запаковувати дані у загальноприйняті формати такі як EDF, BDF, EDF+ [4, 5]. Деякі лікарні використовують Холтер без такого забезпечення, тому на виході отримується звичайний бінарний файл. У таких випадках треба створювати власні декодери. Для розробки програмного забезпечення, яке б

дозволяло автоматизувати процес аналізу ЕКГ було обрано програмне середовище Python версії 3.10 [6], через велике різноманіття готових до використання бібліотек, хорошу швидкодію та відносно просте написання коду. Розглянемо основні складові нашого програмного комплексу (Рис. 1).

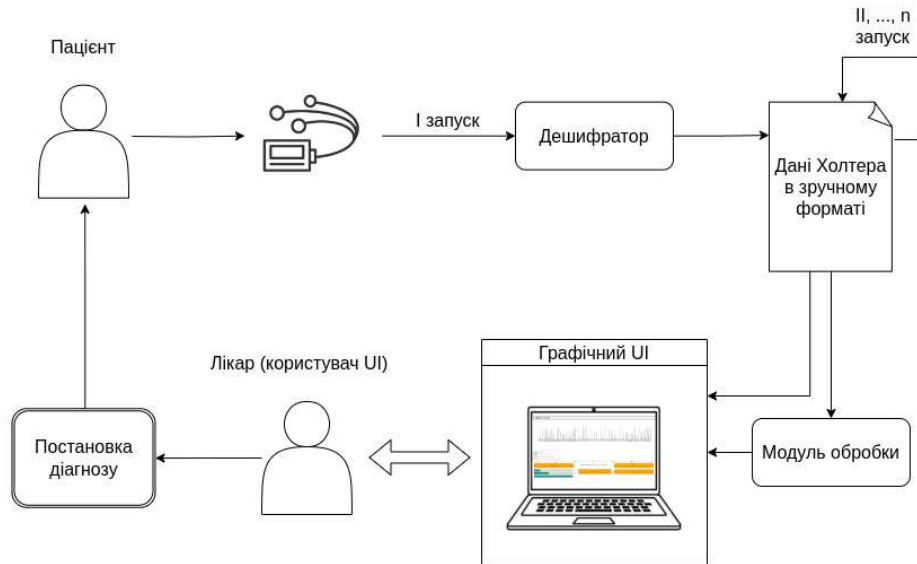


Рис. 1. Схема взаємодії складових програмного забезпечення.

Розроблений застосунок можна розділити на такі блоки:

Блок 1: До нього належить програмний модуль дешифрування каналу, який знімається з датчиків і може залежати від формату даних та типу приладу. Дешифратор може використовувати усі загальноприйняті формати. Користувач системи може самостійно додати власні алгоритми дешифрування, в залежності від конкретного використовуваного прилада Холтера. На вхід дешифратора передаються зашифровані дані (модуль може працювати з файлами у форматі edf, edf+ та інші), як результат модуль створює цілісну структуру даних, зберігаючи окремо відцифровані дані каналу та метадані, що прив'язані до них. Дані ЕКГ, що приходять до цього блоку можна розділити на три категорії: XML-подібні, JSON-подібні та бінарні. XML- та JSON-подібні, мають читабельну структуру [7]. Бінарні формати за різними правилами розділяються на дві підструктури: перша відповідає за опис каналу, друга - за сам канал. Наведемо приклад правила, як кодується файл формату edf [4]:

Таблиця 1

Опис набору даних ЕКГ збережених у файлі EDF формату

1. Опис набору даних ЕКГ
252 байти в Ascii кодуванні - опис набору даних: n – кількість вимірювань у каналі, дати зняття даних, ідентифікація вимірювань (в яких величинах знаходяться вимірювання), тощо.
4 байти Int - кількість каналів (NS)
256 x NS байтів в Ascii кодуванні - опис каналів
2. Дані вимірювань:
$[a_{ij}], i = 1, \dots, n, j = 1, \dots, NS.$

Варто зауважити, що існують формати, які мають крім основного файлу ще файл з метаданими, де зберігається інформація, важлива для його декодування [7]. При розробці універсального декодера, метою якого було передавати у застосунок різні формати, крім інтерфейсу для загальноприйнятих бінарних файлів, було розроблено окремо інтерфейс, який у налаштуваннях обирає за яким правилом декодувати отриманий файл, якщо жодне з правил, що реалізовані у програмі не спрацювало до ЕКГ каналу. Правила наразі описують, з якого байту у закодованому файлі починається опис, а де самі дані, їх кількість — тощо. В залежності від обраного правила використовується конкретний алгоритм декодування, що дає на виході різну кількість даних в залежності від кількості каналів. В такому випадку для кожного потенційного приладу зі своїм форматом, треба буде додавати реалізацію в інтерфейс. Наразі розробляється архітектура більш гнучкого варіанту.

На виході, отримуємо структуру даних, що є легкою у читанні та подальшому використанні. Кожен канал окремо зберігається у текстовому файлі, як звичайна матриця $1 \times N$. Ці текстові файли і є результатом роботи першого блоку. Отриману структуру надалі можна обробляти різними методами обробки даних.

Блок 2: Містить набір файлів певної структури, в яких окремо зберігаються дані кожного каналу, а також їхнє графічне зображення. Самі дані зберігаються у форматі txt, зображення - у pdf форматі для можливості збільшення отриманого графіка без втрати чіткості зображення.

Блок 3: Складається з програмних модулів, що відповідають за обробку отриманої структури даних від блоку 1. Наразі додана можливість для впровадження алгоритму пошуку R-піків, тому є можливість приєднати алгоритми, які були реалізовані в роботах [3, 8]. Надалі програмний комплекс буде наповнюватись статистичними, евристичними та іншими методами обробки даних ЕКГ в тому числі і в реальному часі.

Блок 4: Містить модулі графічного інтерфейсу користувача (GUI). Даний блок відповідає за відображення каналу як у первинному вигляді, так і його частин, також графічний інтерфейс відображає оброблений канал та показує інформацію, яка може знадобитись лікарю для діагнозу. Цей блок забезпечує взаємодію користувача застосунку з іншими блоками програми.

Блочна структура розробленого програмного комплексу дозволяє незалежно змінювати, покращувати та доповнювати методи обробки кардіограм, розширювати сумісність з різними форматами без порушення цілісності програми. При зчитуванні та подальшому збереженні даних ЕКГ від окремих відведень, які записує пристрій Холтера в стандартному форматі, можна використовувати бібліотеки Python 3.10. Серед проаналізованих бібліотек, які працюють з різними форматами ЕКГ, можна виділити такі як WFDB [9] та MNE-tools [10]. Ці бібліотеки містять не тільки програмні засоби зчитування, але й обробки даних самих каналів. Для зчитування даних ЕКГ загальноприйнятих форматів використовується бібліотека PythonEDFLib [11]. Засоби MNE-tools для зчитування використовує методи, які представлені у PythonEDFLib.

У випадку, якщо на вхід подається канал не у стандартизованому бінарному форматі, без анотацій та метаданих щодо нього, дані обробляються розробленим декодером, який приймає конфігурабельні дані від користувача. Тобто користувач системи, заздалегідь задає параметри дешифрування формату (наприклад це є формат, що притаманний лише конкретному приладу та не є загальнодоступним) зазвичай, клініки що надають дані про ЕКГ у таких форматах, надають інформацію як саме цей канал закодовано. Якщо такої інформації немає, декодер намагається підібрати її належним чином самостійно. Для цього реалізовано програмний інтерфейс, що має різні реалізації, одна для стандартизованих форматів, інша приймає налаштування для декодування та намагається декодувати канал.

Дані з приладу Холтера записуються протягом тривалого часу – така кількість записаних даних (близько 100 мільйонів) вимагає велику кількість ресурсів, як пам'яті, так і часу (середня кількість пам'яті для збереження даних одного пацієнта - 100-150 мегабайт, середній час від запуску до отримання результатів – 20-60 секунд, залежить від обчислювальної потужності пристрою та чи дані попередньо розшифровані), тому дані кожного каналу розбиваються на сегменти рівної довжини (довжина сегмента за замовчуванням – 1 хвилина, може бути обрана вручну), а саме на інтервали близько 5-10 тисяч точок, які надалі передаються до графічного інтерфейсу або на обробку алгоритму. Це дозволяє оптимізувати використання ресурсів без втрати точності. При повторному запуску робити розшифрування даних є неефективним. Експериментальним шляхом було з'ясовано, що розшифрування даних займає близько 40 секунд (на процесорі Intel(R) Core(TM) i5-6200U CPU @ 2.30G), в той час, як зчитування розшифрованих даних займає близько 20 секунд, що збільшує час для першого запуску на 50%, але при кожному наступному запуску не потребує розшифрування, та займає тільки час для зчитування даних, що в свою чергу на 50% краще від випадку, коли кожен раз дані розшифровуються, саме тому було вирішено для кожного нового пацієнта створювати файлову структуру для збереження даних та можливістю багаторазового повторного використання.

3. Огляд ІНТЕРФЕЙСУ ОБРОБКИ ДАНИХ ЕКГ

Дані ЕКГ збираються протягом великого проміжку часу (найчастіше протягом однієї доби, в деяких випадках до одного тижня), тому для зручної візуалізації через неможливість повноцінного одночасного перегляду загальний інтервал поділяється на рівні сегменти.

Для взаємодії користувача з даними було розроблено графічний інтерфейс, який дозволяє переходити між інтервалами з можливістю їх перегляду за лінійний час (час відповіді не залежить від віддаленості інтервалу). Розглянемо детальніше функціонал меню керування (Рис. 2).

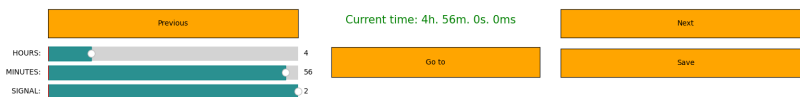


Рис. 2. Функціонал меню керування.

Це меню дозволяє користувачу взаємодіяти з графічним відображенням даних ЕКГ, посередині можна побачити поточний час початку даного інтервалу. Для переходу до наступних інтервалів можна скористатись кнопками Previous, Next, які візуалізують наступний та попередній інтервал відповідно. Для переходу до віддалених інтервалів можна користуватися слайдерами HOURS, MINUTES, які переводять графік в необхідне положення після натискання кнопки Go to. Такий метод переходу зумовлений підвищенням ефективності перегляду даних, адже якщо при перетягуванні слайдерів в реальному часі графік на кожному новому значенні буде перезавантажуватись – це знизить швидкодію програми перегляду. Використання вбудованих засобів та бібліотек мови Python надає можливість взаємодіяти з графіком, а саме робити масштабування, виконувати виділення окремої частини побудованого інтервалу, його прокрутка та ін.

За збереження даних окремих відведень за вказаний проміжок відповідає кнопка Save, яка зберігає графік у форматі png, а також дані кожного відведення окремо, що є корисним для виділення потенційно підозрілих інтервалів даних вручну та їхнього повторного перегляду. При кожному збереженні відбувається створення певної структури папок (Рис. 3).

Розглянемо детальніше графік ЕКГ. Його довжина може встановлюватись користувачем, або становити 60 секунд за замовчуванням. Розміри вікна підлаштовуються під розміри екрана, розмір координатної площини графіку залежить від розмаху даних і визначається симетрично відносно середнього значення. Кількість значень на кожному інтервалі залежить від загальної кількості значень за секунду, а також від коефіцієнта який може обрати користувач для зменшення кількості даних для відображення. На рис. 4 показано зображення, з використанням 50 точок на секунду.

В програмі є можливість переглянути як необроблені дані, так і оброблені за певним алгоритмом (різноманітні алгоритми фільтрації, ...), будь-який з цих варіантів можна запустити незалежно один від одного. Проте

переглядати та аналізувати дані вручну для кожного пацієнта не є ефективним, саме тому було розроблено інтерфейс з можливістю автоматизованого виділення підозрілих фрагментів. Для збереження оброблених даних додана ще одна можливість, яка дозволяє зберігати ефективні характеристичні ознаки, та видаляти непотрібні дані. На рисунку 5 зображено приклад збережених даних, оброблених за пороговим алгоритмом, проте загальний вигляд зображення даних у файлі не залежить від алгоритму, і є універсальним.

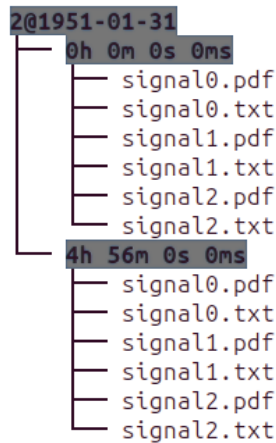


Рис. 3. Функціонал меню керування.

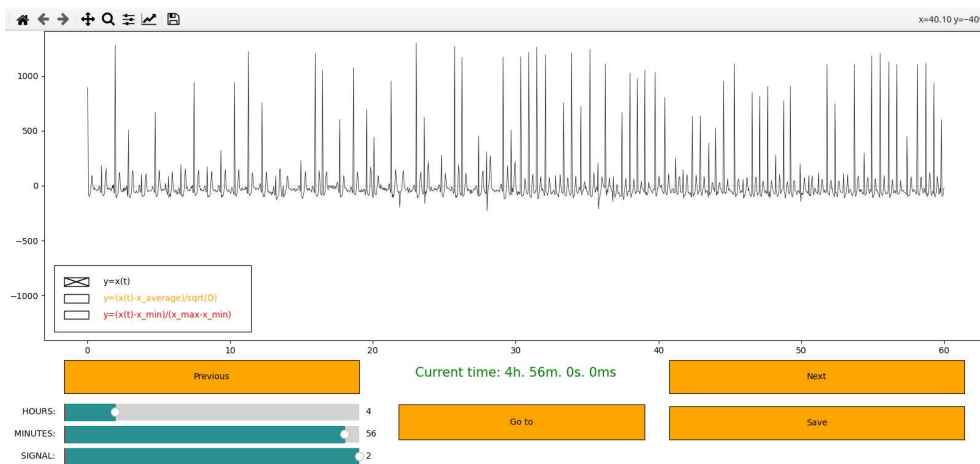


Рис. 4. Дані Холтера перед обробкою.

Date	Signal 0	Signal 1	Signal 2
DELETED FROM Time1:1h 0m 0s 0ms TO Time2:1h 17.0m 48s 68ms PERIOD;			
1h 17m 48s 72ms	1328.0	1556.0	-1282.0
1h 17m 48s 76ms	1176.0	1314.0	-1830.0
1h 17m 48s 80ms	1290.0	1418.0	-1328.0
1h 17m 48s 84ms	1364.0	1464.0	1176.0
1h 17m 48s 88ms	1500.0	1368.0	1596.0
1h 17m 48s 92ms	1344.0	1136.0	1376.0
1h 17m 48s 96ms	1278.0	1132.0	1422.0
1h 17m 49s 0ms	1106.0	892.0	1044.0
1h 17m 49s 3ms	22.0	910.0	1402.0
1h 17m 49s 7ms	674.0	588.0	1202.0
1h 17m 49s 11ms	726.0	630.0	1362.0
1h 17m 49s 15ms	698.0	510.0	1304.0
1h 17m 49s 19ms	498.0	434.0	1344.0
1h 17m 49s 23ms	476.0	570.0	1528.0
1h 17m 49s 27ms	544.0	482.0	1666.0
1h 17m 49s 31ms	534.0	582.0	1716.0
1h 17m 49s 35ms	508.0	596.0	1598.0
1h 17m 49s 39ms	580.0	472.0	1598.0
1h 17m 49s 43ms	650.0	478.0	1558.0
1h 17m 49s 47ms	702.0	646.0	1564.0
1h 17m 49s 52ms	556.0	468.0	1298.0
1h 17m 49s 56ms	478.0	454.0	1164.0
1h 17m 49s 60ms	496.0	518.0	1028.0
1h 17m 49s 64ms	462.0	458.0	848.0

Рис. 5. Зображення оброблених даних з видаленими періодами.

Варто зазначити, що при запуску алгоритму пошуку інтервал на якому він обробляє дані, обирається вручну, що дає змогу за потреби аналізувати як лише необхідну частину даних, так і весь загальний інтервал. Після заповнення файлу з даними відбувається їхня візуалізація, приклад наведений на рис. 6.

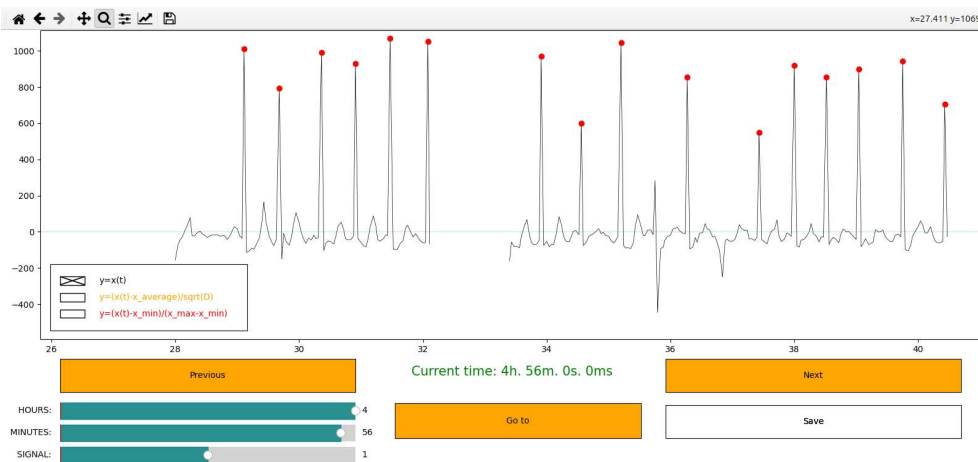


Рис. 6. Зображення оброблених даних з видаленими періодами.

Для уникнення дублювання коду реалізації інтерфейсу, загальна програма має один базовий інтерфейс, який дозволяє переглядати як оброблені певним алгоритмом дані, так і дані без обробки. Необроблені дані можна відображати з початку відліку загального інтервалу, а оброблені – з будь-якої точки. Саме тому алгоритм відображення був модифікований таким чином, щоб порожні інтервали не висвічувались, а відображення починалось з інтервалу, де наявні характеристичні ознаки хвороби серцево-судинної системи.

Одним з поширених захворювань серцево-судинної системи є фібриляція передсердь. Це ефект при якому протягом декількох секунд відбувається

сильно пришвидшене скорочення м'язів серця. На даному етапі реалізації програми, за основу виділення значущих проміжків був обраний пороговий алгоритм, який полягає в виділенні відстані між R-піками в сусідніх QRS-комплексах.

Проміжок залишається, якщо відстань між R-піками менша або рівна за задану програмно, і видаляється в усіх інших випадках. Для знаходження R-піків певних QRS комплексів використовується алгоритм, розроблений в роботі [8]. Апробація запропонованого методу була проведена на базі даних аритмій MIT-BIH від PhysioNet [12], результати тестування якого свідчать про його ефективність.

На рисунку 6 наочно продемонстровано графічний інтерфейс, що зображує результат роботи алгоритму для пошуку ефективних характеристичних ознак.

При розробці інших алгоритмів для аналізу ЕКГ достатньо до програмного комплексу під'єднати програмний модуль, відповідальний за роботу алгоритму. Це відбувається без порушення роботи інших блоків.

6. ВИСНОВОК

Дослідження діяльності серцево-судинної системи є потужним інструментом виявлення початкової стадії захворювання. Автоматизація процесу виявлення вад роботи серця за допомогою комп'ютерних систем є важливою та актуальною задачею на цей момент.

В даній статті було розглянуто частину програми, яка реалізує повний цикл аналізу, від передачі та розшифрування вихідного файлу Холтера, до отримання результату в вигляді оброблених даних, а також зручного графічного відображення.

Розробка та покращення методів автоматичного виявлення підозрілих на наявність ознак хвороби проміжків є важливим завданням, адже це дозволить значно скоротити час на обробку даних, а також виявляти хворобу на початковій стадії та призначити ефективне лікування, що надалі може допомогти уникнути серйозних ускладнень, або навіть і смерті.

Варто зазначити, що остаточне рішення щодо діагнозу має робити лікар, а використання програмних засобів, а також автоматизованих методів аналізу даних ЕКГ зводить ймовірність людської помилки до мінімуму, що безумовно покращує точність діагнозу.

В даній роботі було продемонстровано інтерфейс програми, проведено огляд основних функцій та можливостей, а також розглянуто роботу порогового алгоритму, який не є витратним по ресурсах і є ефективним методом виявлення фібриляції передсердь. Програма може бути корисна лікарям-кардіологам для виявлення аномалій серцевого ритму та подальшій постановці діагнозу на основі отриманих даних.

ЛІТЕРАТУРА

1. Сіренко Ю.М. Стан проблеми серцево-судинної захворюваності та смертності в Україні. *Ліки України*. 2022, 2(258), 11–14. [https://doi.org/10.37987/1997-9894.2022.2\(258\).264084](https://doi.org/10.37987/1997-9894.2022.2(258).264084)
2. Paudel B., Paudel K., The Diagnostic Significance of the Holter Monitoring in the Evaluation of Palpitation. *Journal of Clinical and Diagnostic Research*. 01 march 2013. p. 480-483. <https://www.researchgate.net/publication/236601003>.
3. Krak Yu., Stelia O., Pashko A., Yefremov M., Khorozov O. Electrocardiogram Classification Using Wavelet Transformations. 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), February 2020. <https://www.researchgate.net/publication/341243120>.
4. Kemp B., Värri A., Rosa A.C., Nielsen K.D., Gade J. A simple format for exchange of digitized polygraphic recordings, *Electroencephalogr. Clin. Neurophysiol.* Volume 82. May 1992. P. 391-393. [https://doi.org/10.1016/0013-4694\(92\)90009-7](https://doi.org/10.1016/0013-4694(92)90009-7).
5. Kemp B., Olivan J. European data format 'plus' (EDF+), an EDF alike standard format for the exchange of physiological data. *Clinical Neurophysiology*. Volume 114, Issue 9. September 2003. P. 1755-1761. ISSN 1388-2457. [https://doi.org/10.1016/S1388-2457\(03\)00123-8](https://doi.org/10.1016/S1388-2457(03)00123-8).
6. Python 3.10 [Електронний ресурс] – Режим доступу до ресурсу: <https://www.python.org/downloads/release/python-3100/>
7. Gonzalez D., Garcna-Vbzquez J., Bravo-Zanoguera M. et al. ECG Standards and Formats for Interoperability between mHealth and Healthcare Information Systems: A Scoping Review. *International Journal of Environmental Research and Public Health*. 2022, 19. 11941. <https://doi.org/10.3390/ijerph191911941>.
8. Крак Ю.В., Стеля О.Б., Єфремов М.С., Ляшко А.В. Метод обчислення першої похідної сигналу електрокардіограми для порогового знаходження R-піків. *Доповіді Національної академії наук України*, прийнята до друку. <https://doi.org/10.15407/dopovidi>.
9. Moody G. et al. "WFDB Software Package"(version 10.7.0). *PhysioNet* 2022, <https://doi.org/10.13026/gjvw-1m31>.
10. Gramfort A., Luessi M., Larson E. et al. MEG and EEG data analysis with MNE-Python. *Frontiers in Neuroscience*, 2013, 7, 267, 1–13. <https://doi.org/10.3389/fnins.2013.00267>.
11. Nahrstaedt H., Kern S., LucaCerina et al. holgern/pyedffib: v0.1.23 (v0.1.23). Zenodo.2020. <https://doi.org/10.5281/zenodo.5678481>.
12. MIT-BIH Arrhythmia Database [Електронний ресурс] – Режим доступу до ресурсу: <https://www.physionet.org/content/mitdb/1.0.0/>

Надійшла: 20.05.2024 / Прийнята: 24.06.2024