

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет комп'ютерних наук та кібернетики
Кафедра інтелектуальних програмних систем

Кваліфікаційна робота


на здобуття ступеня бакалавра

за спеціальністю 121 Інженерія програмного забезпечення

на тему:

**РОЗРОБКА ІНСТРУМЕНТУ ДЛЯ КЕРУВАННЯ ДЗЕРКАЛЬНОЮ КАМЕРОЮ НА
ОСНОВІ CANON EOS DIGITAL SDK**

Виконав студент 4-го курсу
Євгеній КУЦЕНКО




(підпис)

Науковий керівник:
кандидат фіз.-мат. наук, доцент
Олексій ЧЕНЦОВ

(підпис)

Засвідчую, що в цій роботі немає запозичень
з праць інших авторів без відповідних
посилань.

Студент



(підпис)

Роботу розглянуто й допущено до захисту
на засіданні кафедри інтелектуальних
програмних систем

«25» травня 2022р.

Протокол № 10

Завідувач кафедри

Олександр ПРОВОТАР

(підпис)

РЕФЕРАТ

Обсяг роботи 25 сторінок, 6 рисунків, 2 лістинги, 1 таблиця та 11 джерел посилань.

Ключові слова: БІБЛІОТЕКА, ДЗЕРКАЛЬНА КАМЕРА, КЕРУВАННЯ ДЗЕРКАЛЬНОЮ КАМЕРОЮ, ПРОГРАМНІ ІНТЕРФЕЙСИ, ОБГОРТКА, CANON, EDSDK, EOS

Об'єктом роботи є керування дзеркальною камерою Canon за допомогою іншого пристрою. Предметом роботи є засіб для керування дзеркальною камерою Canon за допомогою комп'ютера через дротове з'єднання.

Метою роботи є створення обгортки до бібліотеки Canon Eos Digital SDK для спрощення використання можливостей бібліотеки шляхом спрощення інтерфейсів та приховування деталей взаємодії з об'єктами бібліотеки.

Інструменти розробки: інтегроване середовище розробки для C та C++ CLion 2022.1, набір інструментів CMake 3.23.1, система контролю версій Git, мова програмування C++. Тестування результатів роботи виконувалось з камерою Canon EOS 50D з об'єктивами EF50mm f/1.4 USM та EF24-105mm f/4L IS USM.

У результаті виконання роботи було проведено дослідження існуючих засобів для керування дзеркальними камерами, розглянуто можливості SDK та API, запропоновані різними виробниками дзеркальних камер для розробки додатків для керування камерами, досліджено можливості та принципи роботи Canon EOS Digital SDK, створено обгортку для основних можливостей запропонованої бібліотеки, які підтримуються наявною для тестування камерою, а також розроблено простий додаток з консольним інтерфейсом для демонстрації прикладу використання розробленої обгортки.

Під час виконання роботи була також спроба використання розробленої обгортки та простої програми, що використовувала її, на пристрої Raspberry Pi,

але виникла проблема з тим, що наявний пристрій має обмежену підтримку операційної системи Windows (а саме її спрощеної версії), необхідної для роботи додатків, які використовують Canon EOS Digital SDK.

Розроблену обгортку можна доповнювати, додаючи більше можливостей, запропонованих Canon EOS Digital SDK, і можна використовувати для розробки додатків для керування камерами Canon серії EOS, а також деякими камерами серії PowerShot, роботу з якими підтримує відповідний SDK.

ЗМІСТ

СКРОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ.....	5
ВСТУП.....	6
1. ЗАСОБИ ДЛЯ КЕРУВАННЯ КАМЕРОЮ	10
1.1 Фізичні засоби.....	10
1.2 Програмні засоби	10
1.3 SDK/API.....	11
2. РІШЕННЯ, ЯКІ ПРОПОНУЄ CANON ДЛЯ РОЗРОБКИ ВЛАСНИХ ЗАСОБІВ ДЛЯ КЕРУВАННЯ ДЗЕРКАЛЬНИМИ КАМЕРАМИ.....	13
2.1 Camera Control API	13
2.2 EOS Digital SDK.....	14
3. СТВОРЕННЯ ОБГОРТКИ ДЛЯ EDSDK.....	15
3.1 Загальний огляд.....	15
3.2 Структура обгортки	16
3.3 Робота з параметрами	16
3.4 Контроль зйомки.....	18
3.5 Обробка сигналів камери.....	19
3.6 Використання результатів	20
3.7 Приклад додатку з консольним інтерфейсом.....	22
ВИСНОВКИ	24

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

- ОС – операційна система;
- API – Application Programming Interface, прикладний програмний інтерфейс;
- EDSDK – EOS Digital SDK;
- EOS – Electro-Optical System, лінійка плівкових та цифрових однооб’єктивних дзеркальних камер Canon;
- LiveView – режим роботи камери, який дозволяє відображати на дисплеї камери те, що бачить камера в режимі реального часу;
- PTP – Picture Transfer Protocol, протокол передачі зображень;
- RAI – Resource acquisition is initialization, ідіома програмування, відповідно до якої ресурс є захопленим, доки існує об’єкт;
- SDK – Software Development Kit, набір із засобів розробки, утиліт і документації, який дозволяє програмістам створювати прикладні програми за визначеною технологією або для певної платформи;
- USB – Universal Serial Bus, послідовний інтерфейс для підключення периферійних пристроїв.

ВСТУП

Сучасні дзеркальні камери та фотографія в цілому мають досить велику історію розвитку. Вважається, що важливим елементом, на якому були засновані подальші революційні розробки у сфері фотографії, є камера обскура, згадки про яку були у Аристотеля ще у VI столітті до нашої ери. Першу фотографію на камеру вдалося зробити у 1826 році, для цього використовувався олов'яний лист покритий бітумом, і зйомка потребувала витримку 8 годин. З тих пір технологія фотографії швидко розвивалась та вдосконалювалась. У 1839 році розпочалось комерційне використання фотографії для портретів, до цього часу необхідну витримку вдалося скоротити до кількох секунд. Ще через 50 років з'явилась плівка, що дозволило знімати кілька послідовних кадрів, а у 1925 році світу було представлено перший прототип компактної камери Leica, який у порівнянні з громіздкими ящикоподібними камерами став дуже великим проривом. Наступним визначним кроком у розвитку фотографії стало винайдення кольорової плівки у 1936 році. Перші цифрові елементи з'явилися на камерах у 1960-х роках. Спочатку це було автоматичне визначення лише деяких параметрів зйомки чи автофокус, а вже у 1974 році німецька компанія Rollei представила першу повністю автоматичну камеру. У кінці 20-го століття аналогові носії були замінені цифровими, і таким чином камери набули такого вигляду, якими вони є зараз. Більш детально з історією фотографії можна ознайомитися в [1].

Більшість фотографій робиться відносно просто – встановлюються певні значення відповідних параметрів на камері, частина значень залишається на визначення автоматики, і натискається кнопка спуску затвору. Проте, для отримання деяких знімків або необхідні значення параметрів не можливо встановити, або процес зйомки вимагає виконання певної послідовності дій, або наявні інші певні особливості при отриманні знімка. Розглянемо кілька подібних прикладів.

На першому кадрі, відображеному на рис. 1, зображена блискавка над нічним містом. Під час грози блискавка з'являється на небі менше ніж на секунду, тому вичікувати появу блискавки і намагатись її зловити виявляється не дуже ефективним. Для того, щоб створити такий знімок, використовується наступна техніка – трохи закривається діафрагма об'єктиву для зменшення кількості світла, що потрапляє на матрицю, та встановлюється значення витримки рівне кільком секундам (для знімку, зображеному на рис. 1, значення витримки було рівним 10 секундам). Блискавка під час своєї появи є достатньо яскравою, щоб її було видно на фінальному знімку незважаючи на те, що насправді її було видно лише малу частину часу порівняно з тривалістю витримки. Решта ж кадру не є пересвіченою через закриту діафрагму. Але протягом часу, який відповідає значенню витримки, блискавка може не з'явитись, тому для отримання схожого знімку потрібно періодично натискати кнопку спуску затвору, і в результаті на черговому кадрі вдасться спіймати момент з блискавкою.



Рисунок 1. Приклад фотографії

На наступному знімку, зображена на рис. 2, можна бачити розмите зоряне небо при статичних інших об'єктах у кадрі. Такий знімок з'являється у результаті використання тривалої витримки, що дозволяє зробити рух Землі помітним. Для отримання зображеного знімку значення витримки дорівнює 1142 секунди, що складає близько 19 хвилин. Встановити таке значення витримки на камері неможливо. Довільна витримка встановлюється значенням «Bulb», але при такому значенні тривалість витримки визначається тривалістю затиснення кнопки спуску затвору. Тримати кнопку 19 хвилин по-перше важко, а по-друге жодна людина не зможе непорушно тримати руку, не рухаючи камеру протягом такого часу, що призведе до псування фінального знімку.



Рисунок 2. Приклад фотографії

На останньому знімку, зображеному на рис. 3, зображено Місяць. Цей кадр було знято з використанням телеоб'єктиву з фокусною відстанню 300 мм та камерою з матрицею, що має кроп-фактор рівний 1.6, що разом дає дуже значне збільшення. Оскільки подібні фотографії робляться у темну пору доби, то значення витримки має бути рівним не сотим чи тисячним долям секунди, а десятим чи навіть бути більше секунди. При таких значеннях витримки та значному збільшенню навіть незначні рухи камери, спричинені натисканням кнопки спуску затвору, можуть мати негативний вплив на фінальний знімок у вигляді розмивання.

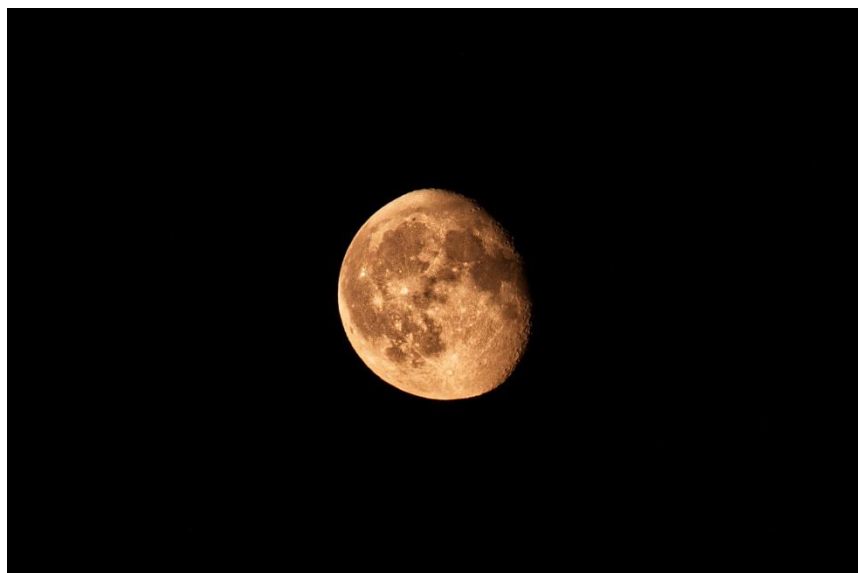


Рисунок 3. Приклад фотографії

Розглянуті випадки показують необхідність використання інших засобів керування камерою для встановлення особливих значень певних параметрів, задання певних сценаріїв роботи камери чи просто для дистанційного керування камерою, для уникнення випадкових зайвих рухів камери.

1. ЗАСОБИ ДЛЯ КЕРУВАННЯ КАМЕРОЮ

1.1 Фізичні засоби

Найпростішими засобами для керування камерою є фізичні засоби. Це невеликі пристрої, які дозволяють проводити зйомку не дотикаючись безпосередньо до камери. Такі пристрої бувають кількох типів. Перший тип це пристрої, які напряду підключаються до камери через спеціальний роз'єм. Другий – пристрої для бездротового керування, їх називають бездротовими чи дистанційними тригерами або дистанційними спусками затвору. Пристрої другого типу складаються з двох частин – приймача, який підключається до камери через спеціальний роз'єм та іноді має кріплення для встановлення на корпус камери (схоже на пристрої-синхронізатори для спалахів), та передатчика який являє собою маленький пульт. З'єднання передатчика та приймача може здійснюватися за допомогою інфрачервоних сигналів або радіохвиль. Останній тип - це повністю бездротовий пульт, але такі пристрої можуть працювати лише з камерами, які підтримують мережеве з'єднання.

Вагомим недоліком таких пристроїв є дуже обмежені функціональні можливості. Переважна більшість моделей дозволяє лише керувати кнопкою спуску затвору, деякі моделі пристроїв додатково дозволяють проводити відкладену зйомку з таймером або серійну зйомку, але зрештою цього все ще може бути недостатньо.

1.2 Програмні засоби

Обмеження функціональних можливостей фізичних пристроїв для керування камерами зумовлене в першу чергу тим, що збільшення функціоналу вимагатиме ускладнення будови цих пристроїв – додавання певної пам'яті, підтримки більш складних апаратних інтерфейсів для передачі більшої кількості можливих команд та інше. Таке ускладнення будови

пристроїв однозначно призведе до значного збільшення їх вартості. Але можна використовувати обчислювальні можливості та апаратні інтерфейси пристроїв, які у всіх є - смартфонів та комп'ютерів для керування камерами. Так компанії виробники камер та деякі сторонні розробники пропонують рішення у вигляді додатків для ряду операційних систем. Такі додатки дозволяють з'єднання з камерою через USB інтерфейс або мережеве з'єднання для тих камер, які відповідно мають мережевий інтерфейс. Функціональні можливості таких додатків значно більші, ніж у пристроїв, описаних у попередньому підрозділі – у таких додатках можуть бути наявні можливості для зміни значень параметрів зйомки та деяких параметрів камери, можливості для завантаження фотографій з камери, керування режимом LiveView, а також іноді додаються інструменти для певної обробки фотографій. На рис. 4 наведено приклади інтерфейсів описаних програмних засобів для керування камерами Canon.

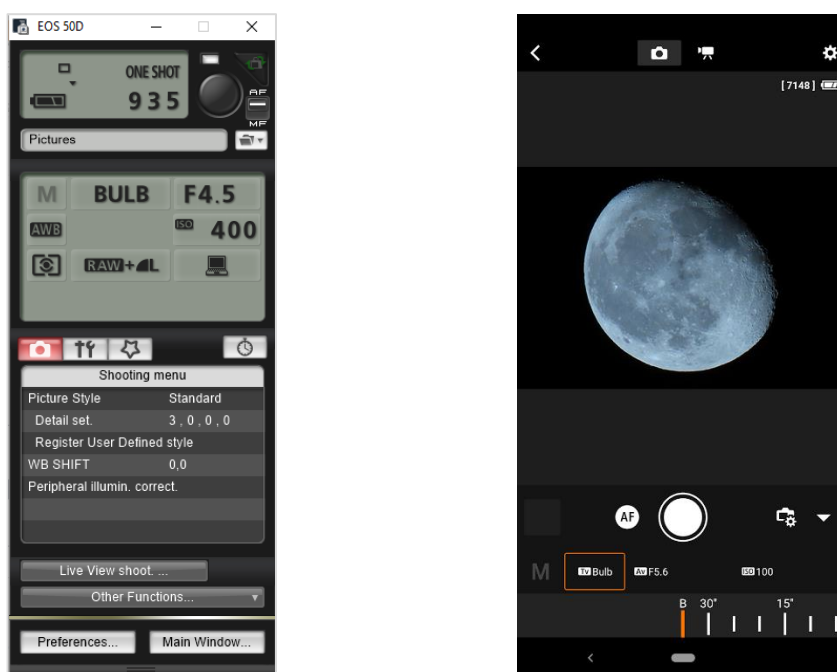


Рисунок 4: а – EOS Utility для ОС Windows [2]; б – Canon Camera Connect для ОС Android [3]

1.3 SDK/API

Хоча програмні засоби, описані у попередньому підрозділі, надають широкий спектр можливостей, все одно як у приватних фотографів, так і у

комерційних підприємств, можуть виникати сценарії, коли існуючих програмних рішень недостатньо, або вони не підходять з певних причин. Для цього виробники пропонують розробникам власні SDK та API для розробки власних рішень з використанням наданих бібліотек чи модулів. Зокрема це стосується найбільших виробників камер - компаній Canon, Nikon, Fujifilm, Sony [4] - [7]. Раніше свій SDK пропонувала також компанія Kodak, проте кілька років тому програму було закрито [8]. Рішення кожного виробника має свої особливості, що зумовлено відмінностями у принципах роботи та характеристиках камер різних виробників.

2. РІШЕННЯ, ЯКІ ПРОПОНУЄ CANON ДЛЯ РОЗРОБКИ ВЛАСНИХ ЗАСОБІВ ДЛЯ КЕРУВАННЯ ДЗЕРКАЛЬНИМИ КАМЕРАМИ

Для розробки власних засобів для керування дзеркальними камерами компанія Canon пропонує приватним розробникам та комерційним підприємствам два наступні рішення: Camera Control API [9] та EOS Digital SDK [9]. Особливості та можливості відповідних інструментів буде розглянуто у наступних підрозділах.

Усі SDK та API, які Canon пропонує розробникам, не тільки ті, які можуть бути використані у задачі керування камерами, вважаються закритою інформацією компанії. Для того щоб отримати доступ до відповідних рішень, потрібно надіслати запит до відповідного регіонального центру підтримки компанії, зазначивши окрім персональних даних ідею майбутньої розробки, її мету та сферу застосування. У разі схвалення запиту, компанія надає доступ до відповідних рішень.

2.1 Camera Control API

Першим рішенням, яке пропонує компанія Canon для розробки власних засобів для керування дзеркальними камерами є Camera Control API. Це рішення підходить для камер, які підтримують мережеве з'єднання. При використанні даного рішення керування камерою відбувається за допомогою відправлення HTTP запитів. Отже додаток, який використовуватиме запропонований API може бути розроблений за допомогою будь-яких засобів та мов програмування, які підтримують відправлення та приймання пакетів відповідного мережевого протоколу прикладного рівня. Одержувачем запитів у такому випадку виступає безпосередньо камера. Для запуску даної функції на камері необхідно використати спеціальну програму-активатор, яка доступна для операційних систем Windows та MacOS, для деяких камер може знадобитись оновлення прошивки

До можливостей, які можна реалізувати з використанням даного API входять налаштування параметрів зйомки та параметрів камери, робота з режимом LiveView, контроль за спуском затвору та завантаження отриманих знімків з камери.

2.2 EOS Digital SDK

Другим рішенням, запропонованим Canon для вирішення поставленої задачі є EOS Digital SDK, або скорочено EDSDK. Дане рішення розраховане на підключення камери до хост-комп'ютера за допомогою інтерфейсу USB. Комунікація між камерою та комп'ютером відбувається за допомогою протоколу PTP. У SDK входить бібліотека, написана на мові C, яка складається з кількох файлів-заголовків та скомпільованих бібліотек для операційних систем Windows та MacOS і надає інтерфейси для виклику функцій бібліотеки. Спрощена схема дизайну системи, що включає у себе надану бібліотеку та додаток, що її використовує, зображено на рис. 5.

Можливості, які надають функції бібліотеки, яку включає SDK, аналогічні до можливостей, які дозволяє реалізувати Camera Control API, а саме налаштування параметрів зйомки та параметрів камери, робота з режимом LiveView, контроль за спуском затвору та завантаження отриманих знімків з камери.

Принципи роботи та усі можливості детально описані у [10, с. 11-32].



Рисунок 5. Спрощена схема дизайну системи, що включає у себе бібліотеку EDSDK та додаток, що її використовує

3. СТВОРЕННЯ ОБГОРТКИ ДЛЯ EDSDK

3.1 Загальний огляд

Бібліотека EDSDK пропонує багато інтерфейсів до функцій, які є відносно низькорівневими – необхідно явно ініціалізувати та завершувати роботу SDK, звільнювати об’єкти, отримані в результаті виконання деяких функцій, відкривати та закривати сесію з’єднання з камерою та інше, а для зчитування значення якогось параметра потрібно спочатку явно за допомогою виклику спеціальної функції визначити розмір значення параметра за його ідентифікатором. Щоб збільшити читабельність коду, що використовуватиме бібліотеку та спростити використання бібліотеки, приховавши взаємодію зі складними інтерфейсами та спеціальними об’єктами, що використовує бібліотека, можна застосувати шаблон проектування фасад.

Спрощена схема дизайну системи, що включає у себе надану бібліотеку та додаток, що її використовує, зображеного на рис. 5, набуде вигляду, зображеного на рис. 6.

Для створення обгортки було виділено три найважливіші можливості бібліотеки – робота з параметрами камери, контроль зйомки та обробка сигналів камери, детальніше про кожну з них описано у підрозділах 3.3, 3.4 та 3.5, відповідно. Мовою програмування було обрано C++, щоб залишити можливість використання бібліотеки під час розробки додатків з використанням мов, що компілюються.



Рисунок 6. Спрощена схема дизайну системи, що включає у себе бібліотеку EDSDK, її обгортку та додаток, що її використовує

3.2 Структура обгортки

Усі реалізовані можливості бібліотеки у обгортці додані у якості полів та методів класу EDSDK, який є реалізацією шаблону проектування одинак (singleton). Також цей клас являє собою RAII-обгортку, що ініціалізує бібліотеку при створенні об'єкту та завершує роботу бібліотеки при видаленні. Оскільки клас реалізує шаблон одинак (singleton), то ініціалізація бібліотеки відбудеться при першому зверненні до її функцій, а завершення роботи з бібліотекою відбудеться при завершенні роботи програми та звільнення пам'яті.

Також у структурі коду наявний клас, що відповідає за об'єкт камери. Він містить усі значення параметрів зйомки та параметрів камери, множини можливих значень для змінних параметрів з урахуванням виставленого режиму зйомки та об'єктиву, що використовується, та надає інтерфейси для керування параметрами, а також станами камери.

Під час розробки обгортки було створено та досліджено прототип обгортки, який включав реалізацію шаблону проектування команда та потокобезпечну чергу, використання яких мало дозволити створення циклу подій у окремому потоці для об'єкту камери. Через особливості роботи бібліотеки EDSDK реалізований код працював не дуже стабільно та зависав на певних ітераціях циклу, тому від впровадження цієї ідеї довелось відмовитись.

3.3 Робота з параметрами

Першою складовою розробленої обгортки до EDSDK є підтримка роботи з параметрами зйомки та параметрами камери. Було обрано найважливіші параметри та умовно розподілено на три групи – незмінні, доступні тільки для читання та параметри налаштування зйомки.

Незмінними є параметри камери, які або ідентифікують камеру, і не можуть бути змінені жодним чином, або можуть бути змінені лише після відключення

камери від хост-комп'ютера та проведення певних дій (наприклад, заміна карти пам'яті). Так до незмінних відносяться наступні параметри:

- назва камери;
- серійний номер;
- версія програмного забезпечення;
- тип карти пам'яті.

Для незмінних параметрів реалізоване зчитування значень з камери при підключенні та функції, що повертають відповідні значення.

До параметрів тільки для читання відносяться ті, які не можуть бути змінені програмним чином, але значення відповідних параметрів можуть бути змінені шляхом проведення певних дій з камерою, наприклад, заміни об'єктива. До цієї групи параметрів відносяться:

- режим камери;
- якість зображення;
- статус об'єктиву;
- назва об'єктиву.

Для параметрів тільки для читання реалізовано функціонал аналогічний тому, що реалізовано для незмінних параметрів – зчитування значень при підключенні камери та функції, що повертають відповідні значення, а також додатково реалізована обробка подій зміни значень, про що детальніше описано у підрозділі 3.5.

До останньої групи параметрів – параметрів зйомки, відносяться параметри, які застосовуються при зйомці кадру. До таких параметрів віднесено:

- баланс білого (WB);
- температура (колір);
- простір кольорів;
- режим зйомки;
- режим вимірювання експозиції;

- режим автофокусу;
- значення діафрагми (Av);
- значення витримки (Tv);
- світлочутливість. (ISO);

Для цієї групи реалізоване зчитування значень при підключенні камери, можливість встановлення нових значень, функції, що повертають значення відповідних параметрів та обробка подій зміни значень. Також для кожного параметра цієї групи зберігається множина можливих значень відповідного параметра при даній конфігурації камери (режим зйомки та характеристики встановленого об'єктиву).

Для багатьох параметрів у бібліотеці їх значенням відповідають деякі наперед задані числові значення, які не завжди можна інтуїтивно співставити. Так, наприклад, значенню витримки, рівному одній секунді, відповідає числове значення 0x38, а значенню витримки, рівному 1/3000, відповідає число 0x94. Тому для спрощення використання наперед заданих числових значень реалізоване їх представлення у рядковій зрозумілій формі.

3.4 Контроль зйомки

Наступною складовою обгортки є контроль зйомки, а саме робота з кнопкою спуску затвору. Кнопка спуску затвору може знаходитись у трьох станах: відпущена, натиснута наполовину, натиснута повністю, та переходити з одного стану в інший шляхом відправки відповідних команд на камеру. Натиснення кнопки спуску затвору наполовину використовується для фокусування, особливо корисне при динамічному автофокусуванні. У свою чергу окремі функції, що відповідають за натиснення та відпускання кнопки спуску затвору, дозволяють робити знімки з довільною тривалістю витримки. У обгортці реалізовані функції для переходу у будь-який з трьох станів, а також окрема функція для імітації одноразового короткочасного натиснення на кнопку.

3.5 Обробка сигналів камери

Третьою складовою обгортки є сигнали камери, які виникають внаслідок певних подій. У EDSDK виділяється три групи подій: події стану, події параметрів та події об'єктів. У розробленій обгортці виконується обробка сигналів, що з'являються у результаті подій параметрів та подій стану.

До подій параметрів відносяться зміна значення певного параметра та зміна множини можливих значень для певного параметру. Зміна множини можливих значень для певного параметра відбувається як наслідок впливу зміни режиму камери або заміни об'єктиву. У свою чергу зміна безпосередньо значення певного параметра може відбуватись через зміну значення безпосередньо через мануальний інтерфейс самої камери, або внаслідок зміни множини можливих значень, якщо попереднє значення не належить оновленій множині.

Під час виникнення таких подій, бібліотека надає ідентифікатор параметра та конкретний тип події (зміна значення чи зміна множини можливих значень), відповідно до цього обробник сигналу зчитує з камери оновлене значення чи множину можливих значень для визначеного параметра.

До подій стану, які оброблюються розробленою обгорткою відносяться події, що сповіщають хост-комп'ютер про відключення камери, перехід у режим сну через короткий проміжок часу та помилку у створенні знімку, наприклад, через помилку автофокусування.

Якщо комп'ютер отримує сигнал про відключення камери, то об'єкт, який відповідає за збереження камери видаляється і звільнюється пам'ять. При отриманні сигналу про перехід у режим сну, функція обгортки надсилає команду для продовження часу роботи камери.

Обробка подій виконується за рахунок реєстрації функцій зворотного зв'язку (callbacks) у якості обробників певних подій після відкриття з'єднання з камерою. Для зчитування сигналів від подій необхідно час від часу викликати

спеціальну функцію, яка зчитує усі сигнали, утворені з моменту попереднього виклику цієї функції.

3.6 Використання результатів

Розроблену обгортку можна зібрати у вигляді бібліотеки та разом зі скомпільованими файлами динамічних бібліотек EDSDK використовувати для розробки будь-яких додатків засобами, які підтримують роботу з інтерфейсами мови C++.

Як вже зазначалось у попередніх розділах, використання EDSDK дозволяє створювати програми, за допомогою яких можна зручно керувати камерою з іншого пристрою та задавати певні автоматизовані сценарії роботи камери. Розроблена обгортка значно спрощує використання можливостей бібліотеки EDSDK. Так у лістингах 1 та 2 наведено порівняння однакової послідовності дій – ініціалізація бібліотеки, підключення камери та натискання кнопки спуску затвора без використання обгортки [10, с. 157-160, 163] та з нею, відповідно.

Лістинг 1

```
void foo() {
    EdsError err = EDS_ERR_OK;
    EdsCameraRef camera = NULL;
    err = EdsInitializeSDK();

    if (err == EDS_ERR_OK) {
        EdsCameraListRef cameraList = NULL;
        EdsUInt32 count = 0;
        err = EdsGetCameraList(&cameraList);
        if (err == EDS_ERR_OK)
            count = EdsGetChildCount(cameraList, &count);

        if (err == EDS_ERR_OK)
            err = EdsGetChildAtIndex(cameraList, 0, &camera);

        if (cameraList)
            EdsRelease(cameraList)
    }
}
```

```
err = EdsOpenSession(camera);

if (err == EDS_ERR_OK) {
    err = EdsSendCommand(camera,
        kEdsCameraCommand_PressShutterButton,
        kEdsCameraCommand_ShutterButton_Completely
    );
    err = EdsSendCommand(camera,
        kEdsCameraCommand_PressShutterButton,
        kEdsCameraCommand_ShutterButton_OFF
    );
}

EdsCloseSession(camera);
}

if(camera)
    EdsRelease(camera);

EdsTerminateSDK();
}
```

Лістинг 2

```
void foo{
    edsdk_w::EDSDK &eds = edsdk_w::EDSDK::get_instance();
    assert(eds.set_camera(0));
    edsdk_w::EDSDK::Camera &camera = eds.get_camera().value();
    camera.shutter_button();
}
```

З повним вихідним кодом розробленої обгортки можна ознайомитись у [11].

3.7 Приклад додатку з консольним інтерфейсом

Для демонстрації використання розробленої обгортки було створено додаток з простим консольним інтерфейсом. Робота програми є дуже простою – циклічно очікується команда на введення, перевіряється коректність команди, якщо команда коректна, то виконується відповідна дія, виводиться результат та викликається команда для зчитування сигналів від подій з камери. У табл 1. наведено список доступних команд для роботи з програмою.

Таблиця 1. Список доступних команд для роботи з програмою

Команда	Параметри	Дія
<code>prop show <property></code>	<code><property></code> - назва параметра	Відображення множини можливих значень зазначеного параметра
<code>prop get <property></code>	<code><property></code> - назва параметра	Відображення поточного значення зазначеного параметра
<code>prop set <property> <num></code>	<code><property></code> - назва параметра <code><num></code> - порядковий номер значення у множині можливих значень	Встановлення відповідного значення зазначеного параметра
<code>ui lock</code>		Блокування мануального інтерфейсу камери
<code>ui unlock</code>		Розблокування мануального інтерфейсу камери
<code>session open</code>		Відкриття з'єднання з камерою

<code>session close</code>		Закриття з'єднання з камерою (з можливістю повторного відкриття)
<code>camera list</code>		Відображення списку камер, підключених до хост-комп'ютера
<code>camera set <num></code>	<num> - номер у списку камер, підключених до хост-комп'ютера	Створення об'єкта камери, що відповідає зазначеному номеру у списку камер, підключених до хост-комп'ютера
<code>camera reset</code>		Видалення об'єкта камери, звільнення пам'яті
<code>exit</code>		Завершення роботи програми

При використанні з'єднання з хост-комп'ютером за допомогою протоколу SSH¹ можна керувати камерою дистанційно.

¹ SSH, Secure Shell— мережевий протокол рівня застосунків, що дозволяє проводити віддалене управління комп'ютером.

ВИСНОВКИ

Під час виконання роботи було досліджено принципи роботи дзеркальних камер, та методи отримання деяких особливих фотографій, досліджено існуючі засоби керування дзеркальними камерами та проведено практичне порівняння використання засобів різних типів, ознайомлено з можливостями SDK та API запропонованими виробниками дзеркальних камер. Для отримання доступу до Canon EOS Digital SDK було надіслано запит до Європейського відділу підтримки Canon, який було схвалено. Після отримання доступу було ознайомлено з можливостями та принципами роботи бібліотеки EDSDK і створено обгортку для спрощення використання можливостей бібліотеки шляхом спрощення інтерфейсів та приховування деталей взаємодії з об'єктами бібліотеки, яка підтримує роботу з найважливішими серед запропонованих можливостей, а саме робота з параметрами камери, контроль зйомки та обробка сигналів від подій камери. Також було створено простий додаток з консольним інтерфейсом для демонстрації прикладу використання створеної обгортки.

Розроблену обгортку можна розвивати, додаючи підтримку решти можливостей, запропонованих бібліотекою EDSDK, серед яких завантаження фотографій з камери та робота з режимом LiveView. Також використовуючи розроблену обгортку можна створювати повноцінні додатки для керування камерами Canon, роботу з якими підтримує EDSDK.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The History of Photography [Електронний ресурс] / Lumas – Режим доступу: <https://www.lumas.com/history-photography/>
2. Canon EOS Utility [Електронний ресурс] – Режим доступу: <https://www.canon.ua/software/eos-utility/>
3. Canon Camera Connect у Google Play [Електронний ресурс] – Режим доступу: <https://play.google.com/store/apps/details?id=jp.co.canon.ic.cameraconnect>
4. Canon Imaging Solutions, Software Development Kit (SDK) [Електронний ресурс] – Режим доступу: <https://www.canon-europe.com/business/imaging-solutions/sdk/>
5. Nikon SDKs for Digital Imaging Products [Електронний ресурс] – Режим доступу: <https://sdk.nikonimaging.com/apply/>
6. FUJIFILM X Series and GFX System Digital Camera Control Software Development Kit [Електронний ресурс] – Режим доступу: <https://fujifilm-x.com/global/special/camera-control-sdk/>
7. Sony Camera Remote SDK [Електронний ресурс] – Режим доступу: <https://support.d-imaging.sony.co.jp/app/sdk/en/index.html>
8. How can I get the SDK for Kodak Pro SLRs? : повідомлення з форуму DPReview [Електронний ресурс] – Режим доступу: <https://www.dpreview.com/forums/post/42165866>
9. Canon Developer Programme, Camera Overview [Електронний ресурс] – Режим доступу: <https://developers.canon-europe.com/developers/s/camera>
10. EDSDK 13.14.40 API Programming Reference [Електронний ресурс] / Canon Inc. / 2022 – 195 с.
11. yevkk/camera-control-tool : GitHub репозиторій [Електронний ресурс] – Режим доступу: <https://github.com/yevkk/camera-control-tool>