



Рисунок 5 — Пример интерфейса

Заключение. Figma является мощным инструментом для дизайна, который позволяет работать в реальном времени, делиться проектами и комментариями с другими участниками команды. Были рассмотрены основные возможности Figma, такие как создание макетов, прототипирование, работа с компонентами и библиотеками, а также совместная работа в режиме реального времени.

Был рассмотрен пример использования Figma в области разработки дизайна веб-сайта. Описанные возможности Figma позволяют значительно упростить и ускорить процесс дизайна, а также повысить качество и эффективность работы дизайнеров и команды в целом.

Список цитируемых источников

1. Figma: преимущества и возможности приложения для дизайна [Электронный ресурс]. — Режим доступа: <https://emailmatix.ru/blog/figma/>. — Дата доступа: 29.04.2023.
2. Официальный сайт программного продукта Figma [Электронный ресурс]. — Режим доступа: <https://www.figma.com/>. — Дата доступа: 29.04.2023.

УДК 004.93'11

В. А. Бертош, Е. Г. Шапович

*Учреждение образования «Барановичский государственный университет», Барановичи,
Республика Беларусь*

УПРАВЛЕНИЕ ГРОМКОСТЬЮ И ЯРКОСТЬЮ КОМПЬЮТЕРА ПРИ ПОМОЩИ ЖЕСТОВ

Введение. Жесты рук — мощное средство коммуникации для взаимодействия человека с компьютером. Несколько входных данных устройства доступны для взаимодействия с компьютером, такие как клавиатура, мышь, джойстик и сенсорный экран, но эти устройства не обеспечивают самый простой способ коммуникации. При этом предлагаемая система будет состоять из интерфейса настольного компьютера или ноутбука, а жесты рук могут использоваться пользователями для управления настройками компьютера, такие как звук системы и яркость экрана.

Первым и наиболее важным шагом к любой системе распознавания жестов рук является внедрение системы отслеживания рук. Некоторые сенсорные устройства обычно используются в методах на основе перчаток виртуальной реальности для оцифровки движений рук и пальцев в многопараметрические данные. Для разрабатываемого метода управления потребуется только веб-камера, чтобы можно было реализовать естественное взаимодействие между человеком и компьютером без использования каких-либо дополнительных устройств.

Сложностью такой системы состоит в том, что восприятие рук в реальном времени является явно сложной задачей компьютерного зрения, поскольку руки часто закрывают себя или друг друга (например, окклюзии пальцев, ладоней и рукопожатия) и не имеют высококонтрастных паттернов.

Основная часть. В компьютерном зрении жесты рук являются активной областью исследований с целью распознавания языка жестов и взаимодействия человека с компьютером. В данном исследовании используются некоторые алгоритмы и модули для обнаружения жестов человека, и эти жесты принимаются в качестве входных данных в системе. Здесь используется несколько модулей, такие как OpenCV-Python, NumPy, MediaPipe, imutils, SciPy для отслеживания жестов.

OpenCV — это библиотека компьютерного зрения с открытым исходным кодом, распространяющаяся бесплатно, которая предназначена для анализа, классификации и обработки изображений [1].

MediaPipe — кроссплатформенный фреймворк для запуска пайплайнов машинного обучения. MediaPipe позволяет из калькуляторов составлять необходимый пайплайн для запуска модели, а затем просто встраивать его в приложения на разных платформах. В MediaPipe есть встроенные калькуляторы для запуска TensorFlow и TFLite моделей [2].

В данной работе использовались решения машинного обучения MediaPipe Hands.

MediaPipe Hands — это высокоточное решение для отслеживания рук и пальцев. Оно использует машинное обучение, чтобы вывести 21 трехмерный ориентир руки всего из одного кадра. В то время как современные подходы в основном полагаются на мощные настольные среды для логического вывода, данный метод обеспечивает производительность в реальном времени на мобильном телефоне и даже масштабируется для нескольких рук.

MediaPipe Hands использует конвейер машинного обучения, состоящий из нескольких моделей, работающих вместе: модель обнаружения ладони, которая работает с полным изображением и возвращает ориентированную ограничивающую рамку руки. Модель ориентира руки, которая работает с обрезанной областью изображения, определяемой детектором ладони, и возвращает высокоточные трехмерные ключевые точки руки.

Предоставление точно обрезанного изображения руки модели ориентира руки резко снижает потребность в дополнении данных (например, вращении, перемещении и масштабировании) и вместо этого позволяет сети направить большую часть своих возможностей на точность прогнозирования координат. Кроме того, в данном пайплайне обрезки также могут быть сгенерированы на основе ориентиров руки, определенных в предыдущем кадре, и только когда модель ориентира больше не может идентифицировать присутствие руки, вызывается обнаружение ладони для повторной локализации руки [3].

Библиотека SciPy предоставляет алгоритмы оптимизации, интеграции, интерполяции, задач на собственные значения, алгебраических уравнений, дифференциальных уравнений, статистики и многих других классов задач. Расширяет функции библиотеки NumPy, предоставляя дополнительные инструменты для вычисления массивов и предоставляет специализированные структуры данных, такие как разреженные матрицы и k-мерные деревья [4].

Библиотека imutils — это ряд удобных функций, упрощающих базовые функции обработки изображений, такие как перевод, вращение, изменение размера, скелетирование, отображение изображений Matplotlib, сортировка контуров, обнаружение краев и многое другое с OpenCV и Python 2.7 и Python 3 [5].

Для написания данной системы был выбран язык программирования Python.

Python — высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ [6].

В качестве среды разработки была выбрана IDE PyCharm Community Edition.

PyCharm — это кроссплатформенная интегрированная среда разработки для языка программирования Python, разработанная компанией JetBrains на основе IntelliJ IDEA. Предоставляет пользователю комплекс средств для графической отладки и работы с кодом. Продукт доступен в двух версиях: PyCharm Community Edition — бесплатная версия, находится под лицензией Apache License, и PyCharm Professional Edition — расширенная версия продукта, обладающая дополнительной функциональностью [7].

Разработанное приложение представляет собой окно, в котором отображается в режиме реального времени видеопоток с вебкамеры. Кроме этого, оно отображает текущий уровень громкости системы компьютера и количество сменяемых кадров в секунду. Данное окно представлено на рисунке 1.

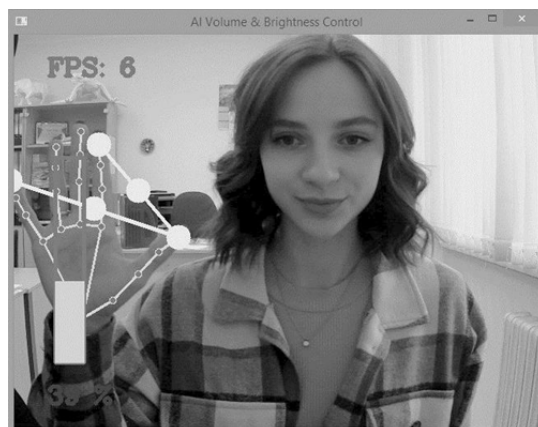


Рисунок 1 — Главное окно

Для изменения уровня громкости системы используется отслеживание положения указательного пальца по отношению к большому. Чем ближе указательный палец к большому — тем ниже уровень громкости (рисунок 2, а), чем дальше — тем выше уровень громкости (рисунок 2, б).

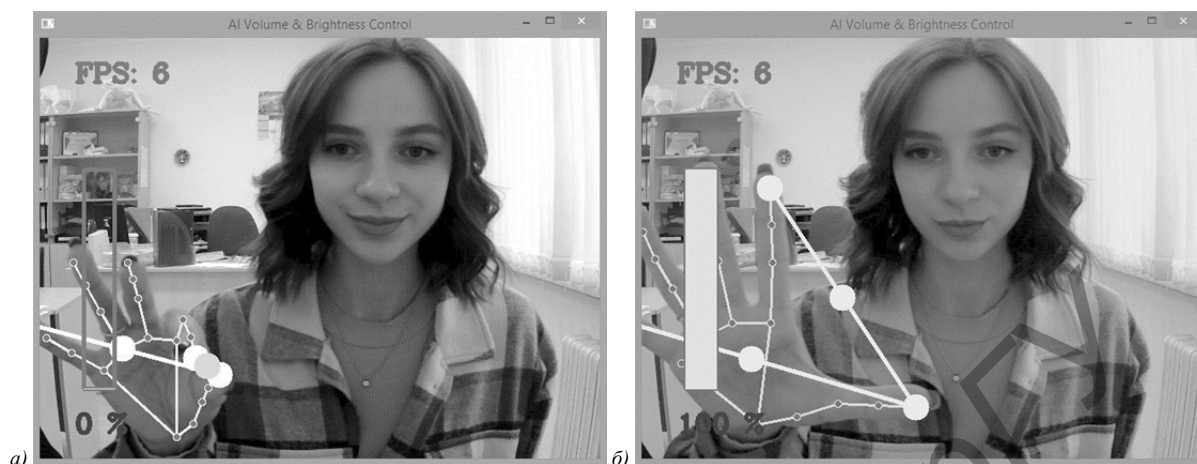


Рисунок 2 — Изменение громкости системы: а — уменьшение; б — увеличение

Для изменения яркости системы используется отслеживание положения мизинца по отношению к большому пальцу. Чем ближе мизинец к большому пальцу — тем ниже яркость (рисунок 3, а), чем дальше — тем выше яркость (рисунок 3, б).

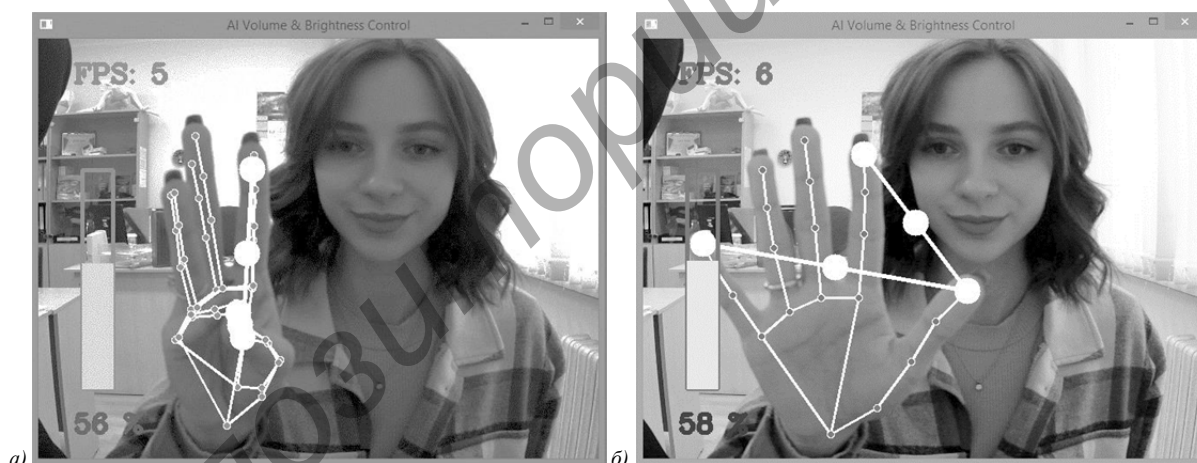


Рисунок 3 — Изменение яркости монитора: а — уменьшение; б — увеличение

В целом принцип работы прост. После захвата ввода от пользователя изображение используется в системе отслеживания рук для проверки размеров и формы жеста, полученного в системе. После этого жесты идентифицируются из обученных данных, и на основе этих данных жесты распознаются и используются для обработки системы для реализации таких функций, как увеличение и уменьшение громкости или яркости.

Заключение. Этот проект представляет программный модуль, который позволяет пользователю жестами рук управлять программным обеспечением. Регулятор громкости на основе жестов не требует какого-либо определенного типа маркеров, и ими можно управлять в нашей реальной жизни на простом персональном компьютере. Для работы подойдут компьютеры с очень дешевыми камерами, поскольку для обнаружения или записи жестов рук не требуются камеры очень высокого разрешения.

В частности, система отслеживает положения кончиков мизинца, большого и указательного пальцев каждой руки. Данное исследование показывает, что можно автоматизировать некоторые функции в системе для управления жестами. Полученные результаты можно использовать в системах управления в виртуальной реальности.

Основная цель достигнута. Продемонстрировано взаимодействие человека с компьютером, а также разработано приложение с использованием языка Python, библиотек MediaPipe и OpenCV.

Список цитируемых источников

1. Platforms — OpenCV [Электронный ресурс]. — Режим доступа: <https://opencv.org/platforms/>. — Дата доступа: 02.01.2023.
2. MediaPipe [Электронный ресурс]. — Режим доступа: <https://mediapipe.dev/>. — Дата доступа: 03.01.2023.
3. Hands mediapipe [Электронный ресурс]. — Режим доступа: <https://google.github.io/mediapipe/solutions/hands>. — Дата доступа: 03.01.2023.
4. SciPy [Электронный ресурс]. — Режим доступа: <https://scipy.org/>. — Дата доступа: 04.01.2023.
5. imutils PyPI [Электронный ресурс]. — Режим доступа: <https://pypi.org/project/imutils/>. — Дата доступа: 05.01.2023.
6. Северанс, Ч. Р. Python для всех / Ч. Р. Северанс ; пер. с англ. А. В. Снастина. — Москва : ДМК Пресс, 2022. — 262 с.
7. PyCharm: IDE для профессиональной разработки на Python от JetBrains [Электронный ресурс]. — Режим доступа: <https://www.jetbrains.com/ru-ru/pycharm/>. — Дата доступа: 05.01.2023.

УДК 629.735

Э. А. Веракса, Г. М. Раковцы

Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь

РАЗРАБОТКА ДРОНА НА ДИСТАНЦИОННОМ УПРАВЛЕНИИ С ИСПОЛЬЗОВАНИЕМ ARDUINO

Введение. Разработка дрона на дистанционном управлении является одной из наиболее актуальных тем в современном мире. Дроны широко используются в различных сферах, начиная от промышленной и заканчивая бытовой. Они могут выполнять разнообразные задачи, такие как съемка видео и фотографий, доставка грузов, контроль за территориями и объектами, а также использоваться в научных исследованиях. Разработка дрона на дистанционном управлении требует знаний в области электроники, программирования и механики, а также умения интегрировать эти знания для создания работающего прототипа.

Arduino была выбрана для разработки дрона из-за ее открытости, доступности и гибкости. Arduino позволяет быстро и легко создавать прототипы устройств, а также облегчает управление подключенными датчиками и модулями. Кроме того, Arduino имеет большое сообщество пользователей, что облегчает поиск решений и поддержку проекта.

Основная часть. Целью работы является создание недорогого квадрокоптера, обладающего автономным питанием, алгоритмом стабилизации полета, а также возможностью управления с помощью смартфона.

При разработке устройство ставились задачи создать устройство, которое будет контролироваться оператором в ручном режиме. Оператор должен иметь возможность управлять полетом дрона, изменять его высоту, направление и скорость движения. Важным аспектом при создании дрона на дистанционном управлении является обеспечение безопасности полета, поэтому в ручном режиме должны быть предусмотрены системы автоматического возврата на базу и аварийного приземления в случае потери связи с оператором. Дополнительно была предусмотрена функция автоматического режима для наблюдения за территорией в течение некоторого времени при условии дальнейшей доработки и установки камеры видеонаблюдения.

В качестве средств реализации проекта была использована среда разработки Arduino IDE с использованием C++ подобного языка программирования. Arduino IDE — это интегрированная среда разработки (IDE) для программирования микроконтроллеров на платформе Arduino. Она предоставляет пользователю набор инструментов для написания, компиляции и загрузки программного кода на плату Arduino. Среда включает в себя такие компоненты как: редактор кода — встроенный текстовый редактор, который позволяет создавать и редактировать программный код на языке Arduino; компилятор — преобразует исходный код в машинный код, который может быть загружен на плату Arduino; загрузчик — загружает скомпилированный код на плату Arduino через USB-порт; серийный монитор — позволяет пользователю просматривать вывод программы в реальном времени; библиотеки — коллекция готовых модулей и функций, которые можно использовать для упрощения разработки программного кода.

Для симуляции проекта была использована система автоматизированного проектирования Proteus позволяющая виртуально смоделировать работу огромного количества аналоговых и цифровых устройств. Программный пакет Proteus VSM позволяет собрать схему любого электронного устройства и симулировать его работу, выявляя ошибки, допущенные на стадии проектирования и трассировки. Программа состоит из двух модулей. ISIS — редактор электронных схем с последующей имитацией их работы. ARES — редактор печатных плат, оснащенный автотрассировщиком Electra, встроенным редактором библиотек и автоматической системой размещения компонентов на плате. Кроме этого ARES может создать трехмерную модель печатной платы. Proteus VSM включает в себя более 6000 электронных компонентов со всеми справочными данными, а также демонстрационные ознакомительные проекты. Программа имеет инструменты USBCONN и COMPIB, которые позволяют подключить виртуальное устройство к портам USB и COM компьютера. При подсоединении к этим портам любого внешнего прибора виртуальная схема будет работать с ним, как если бы она существовала в реальности. Proteus VSM поддерживает следующие компиляторы: CodeVisionAVR и WinAVR (AVR), ICC (AVR, ARM7, Motorola), HiTECH (8051, PIC Microchip) и Keil (8051, ARM). Существует возможность экспорта моделей электронных компонентов из программы PSpice [1].