



VisionLabs
MACHINES CAN SEE

VISIONLABS LUNA PASS

Инструкция по эксплуатации

ООО «ВижнЛабс»

123458, г. Москва, ул. Твардовского д. 8, стр. 1



+7 (499) 399 3361



info@visionlabs.ru



www.visionlabs.ru

Оглавление

Глоссарий	3
Введение	4
1. Информация по эксплуатации Системы	5
1.1. Алгоритм работы	5
1.2. Эксплуатация Системы	6
1.2.1. Коды статусов и описание ошибок	7
2. Информация для настройки Системы	9
2.1. Подключение LUNA PASS к сайту	9
2.2. Подключение компонента LPMessageRenderer	10
2.3. Подключение компонента LPFaceEstimator	11
Приложение	12

Глоссарий

Термин	Определение
Liveness	Программный способ подтверждения витальности (живучести, жизненности) человека по одному или нескольким изображениям с целью предотвращения спуфинг-атак
Детекция	Обнаружение сервисом на изображении области (областей), которая содержит лица
Спуфинг-атака	Тип атаки, основанной на фальсификации передаваемых данных, в частности подмена живого человека на поддельное изображение (например, фотографию) с целью обмана системы

Введение

Настоящий документ описывает процесс эксплуатации программного обеспечения VisionLabs LUNA PASS.

ПО VisionLabs LUNA PASS (далее – Система) представляет собой набор библиотек, обеспечивающих возможность реализации работы в режиме реального времени для выполнения детекции лица в кадре, проверки витальности человека и передачи данных во внешнюю систему.

Система предназначена для реализации процесса приёма и обработки цветного видеопотока с устройства видеозаписи, проверки качества изображения, выбора лучшего кадра, детекции лица методом машинного вычисления по двум изображениям, проверки предъявляемого изображения Liveness-алгоритмами и защиты от подмены изображения макетами путём анализа карты глубин и последующей передачи лучшего кадра лица в системы интеграции устройств.

1. Информация по эксплуатации Системы

1.1. Алгоритм работы

Диаграмма взаимодействия компонентов LUNA PASS представлена на Рисунке 1.

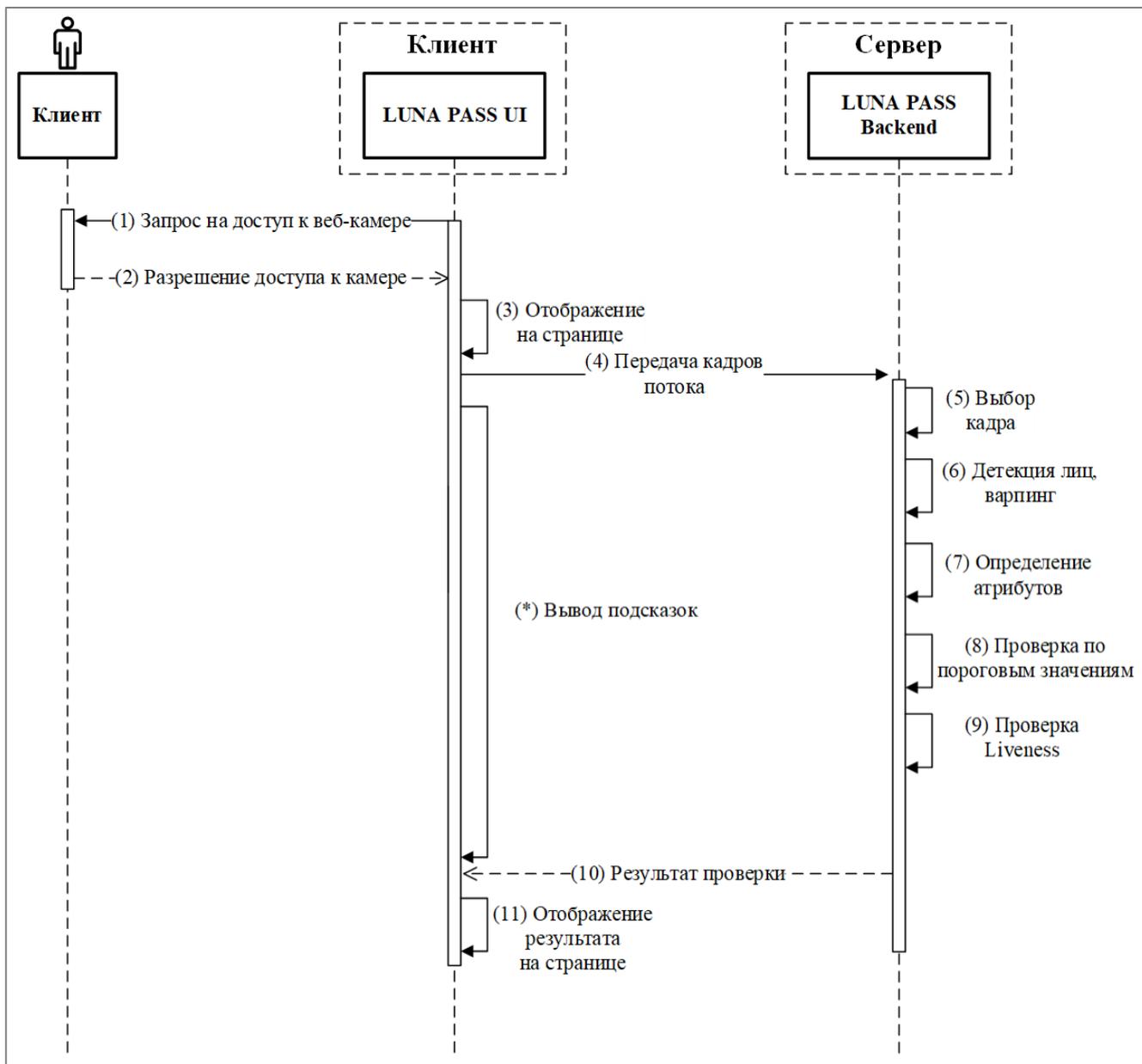


Рисунок 1. Диаграмма взаимодействия компонентов решения LUNA PASS

Описание процесса взаимодействия пользователя с сервисом LUNA PASS:

- (0) — Пользователь получает доступ к фронтальной части сервиса LUNA PASS, открыв страницу в веб-браузере.
- (1) — Сервис запрашивает доступ к веб-камере.

(2) — Пользователь должен разрешить доступ, чтобы продолжить пользоваться сервисом.

(3) — Происходит захват изображения с веб-камеры, пользователь видит видеопоток с камеры в режиме реального времени.

(4) — LUNA PASS UI посредством защищённого протокола Web Socket направляет кадры видеопотока на сервер в LUNA PASS Backend.

(5) — На сервере в backend части в составе LUNA PASS установлена система LUNA SDK 5, которая проводит выбор кадра.

(6) — При помощи LUNA SDK 5 происходит подготовка кадра к обработке — детекция лица на кадре и нормализация кадра (варпинг).

(7) — LUNA SDK определяет атрибуты кадра (контрольные значения кадра).

(8) — Происходит проверка атрибутов кадра по допустимым пороговым значениям.

Если результат проверки не удовлетворительный, LUNA PASS Backend возвращает запрос на получение нового кадра.

(9) — Компонент LUNA SDK MOBILE LIVENESS в составе LUNA PASS производит проверку, является ли человек в кадре живым или нет.

Если результат проверки не удовлетворительный, LUNA PASS Backend возвращает запрос на получение нового кадра.

(10) — LUNA PASS Backend возвращает результат проверки в LUNA PASS UI.

(11) — Пользователь видит результат в формате сообщения в веб-браузере.

Если LUNA PASS не может определить лицо человека в поступающем от LUNA PASS UI в LUNA PASS Backend потоке, не может провести проверку Liveness или определить атрибуты (например, пользователь далеко сидит от камеры или его голова повернута на слишком большой угол), на экране монитора в веб-браузере будут отображаться указания в виде подсказок, как пользователь должен изменить положение лица относительно веб-камеры.

1.2. Эксплуатация Системы

Для Системы VisionLabs LUNA PASS необходимо ПО VisionLabs LUNA SDK. Запросы отправляют в ПО VisionLabs LUNA SDK по WebSocet.

Эстиматоры, заращиваемые из VisionLabs LUNA SDK, представлены в Таблице 1.

Таблица 1. Пороговые значения параметров изображения

Эстиматор	Описание
liveness_estimator	Эстиматор определяет, настоящее ли лицо человека или поддельное (фотография, распечатанное изображение, экран смартфона).

Эстиматор	Описание
quality_estimator	Эстиматор оценки параметров изображения (пересвеченность, затемненность, размытие, равномерность освещения и наличие блики)
head_pose_estimator	Эстиматор положения головы определяет углы поворота головы по трём осям.
glasses_estimator	Эстиматор определяет наличие и тип очков у человека (солнцезащитные или обычные)
mouth_estimator	Эстиматор рта предсказывает преобладающее состояние рта (открыт/закрыт, наличие улыбки и перекрытие рта).

1.2.1. Коды статусов и описание ошибок

Коды статусов и описание ошибок в полезных нагрузках при выполнении проверки Liveness приведены в Таблице 2.

Таблица 2. Коды статусов и описание ошибок

Код ошибки	Описание
0	Ошибки нет, кадр прошел проверку
1	Не удалось обработать кадр
2	Не удалось получить оценку качества
3	Кадр смазан
4	Кадр засвечен
5	Кадр затемнен
6	Кадр неравномерно освещен, есть засветы
7	Кадр имеет блики
8	Кадр не содержит лиц
9	Кадр содержит больше чем одно лицо
10	Ширина кадра недостаточна
11	Высота кадра недостаточна
12	Ширина кадра недостаточна
13	Высота кадра недостаточна
14	Лицо слишком близко к верхней границе кадра
15	Лицо слишком близко к правой границе кадра
16	Лицо слишком близко к нижней границе кадра
17	Лицо слишком близко к левой границе кадра

Код ошибки	Описание
18	Не удалось получить оценку положения головы
19	Голова слишком сильно повернута вправо
20	Голова слишком сильно повернута влево
21	Голова слишком сильно наклонена вправо
22	Голова слишком сильно наклонена влево
23	Голова слишком сильно опущена вниз
24	Голова слишком сильно запрокинута назад
25	Не удалось получить оценку рта
26	Рот не виден в кадре
27	Не удалось получить оценку очков
28	Тип очков не соответствует допустимому
29	Не удалось получить оценку AGS
30	Оценка AGS слишком низкая
31	Не удалось получить оценку Liveness
32	Оценка Liveness слишком низкая
33	Оценка качества оценки Liveness слишком низкая
34	Не удалось произвести детекцию лица в кадре

2. Информация для настройки Системы

2.1. Подключение LUNA PASS к сайту

Для использования LUNA PASS на сайте необходимо выполнить следующие команды:

1. Подключите скрипт к странице:

```
<script src="luna-pass.{version}.js"></script>
```

2. Добавьте на страницу следующие функции:

```
a async function main() {
  // Создание компонента
  const lunapass = new LunaPass(
    // DOM узел, к которому монтируется компонент и внутри которого будет
отображаться
    // видео с подключенной камеры.
    rootNode // document.getElementById('vllunapass')
  );

  // Инициализация компонента

  // Установка соединения
  // Для передачи идентификатора сессии в LunaPass Backend необходимо
использовать cookie,
  // установленные в браузере cookie передаются автоматически, если
необходимо дополнительные,
  // то установите их средствами браузера, например:
  // document.cookie = 'LUNA_PASS_CLIENT_ID={required client id}; path=/
  await lunapass.connectWS(
    connectionString // Адрес LunaPassBackend, вида
`wss://{login}:{pass}@{host}/ws`
  );

  // Подключение камеры
  await lunapass.attachCamera();

  // Коллбек для обработки ответа результата от сервера
  const onMessage = (frameStatus) => {
    // Выполнить клиентский код
    console.log(frameStatus);
  }

  // Проверить Liveness автоматически
  // В результате проверки возвращается ответ сервера с результирующим JWT
сообщением
  const result = await lunapass.checkLiveness(onMessage);
  if (result.isOk) {
    // Выполнить клиентский код
    alert(result)
  }
}
```

```
// Проверить текущий кадр на Liveness
const frameStatus = await lunapass.checkFrame();

// Разорвать соединение с LunaPass Backend
await lunapass.disconnectWS();

// Отключить камеру
await lunapass.detachCamera();
}
```

2.2. Подключение компонента LPMessageRenderer

Компонент предназначен для отображения пользовательских подсказок.

Для использования LPMessageRenderer необходимо выполнить следующие команды:

1. Подключите скрипт к странице:

```
<script src="luna-pass.{version}.message-renderer.js"></script>
```

2. Добавьте на страницу следующие функции:

```
async function main() {
  // Создание и инициализации LunaPass UI
  const rootNode = document.getElementById('vllunapass');
  const lunapass = new LunaPass(rootNode);
  await lunapass.connectWS(connectionString);
  await lunapass.attachCamera();

  // Создание компонента для рендера подсказок
  const msgRenderer = new LPMessageRenderer({
    test: true, // отображать текстовые подсказки?
    graphics: true, // отображать графические подсказки?
  });
  // Инициализация компонента для рендера подсказок
  // Данный шаг необходим для корректной настройки размеров области подсказок
  // Выполняется только после подключения камеры и отображения видеопотока с
камеры
  await msgRenderer.init(rootNode);

  // Коллбек для обработки ответа результата от сервера
  const onMessage = (frameStatus) => {
    // Отобразить информацию о результате проверки текущего кадра
    msgRenderer.render(frameStatus);
  }

  const result = await lunapass.checkLiveness(onMessage);
  if (result.isOk) {
    // Выполнить клиентский код
    alert(result)
  }

  // Отключение LPMessageRenderer
  msgRenderer.destroy();
}
```

```
// Отключение LunaPass UI
await lunapass.disconnectWS();
await lunapass.detachCamera();
}
```

2.3. Подключение компонента LPFaceEstimator

Компонент предназначен для детекции лиц на кадре в браузере.

Для использования LPFaceEstimator необходимо выполнить следующие команды:

1. Подключите скрипт к странице:

```
<script src="luna-pass.{version}.face-estimator.js"></script>
```

2. Добавьте на страницу следующие функции:

```
async function main() {
  const rootNode = document.getElementById('vllunapass');
  const lunapass = new LunaPass(rootNode);
  await lunapass.connectWS(connectionString);
  await lunapass.attachCamera();
  const msgRenderer = new LPMessageRenderer();
  await msgRenderer.init(rootNode);

  // Создание компонента для детекции лиц в браузере
  const faceEstimator = new LPFaceEstimator();
  await faceEstimator.init();

  // Коллбек для обработки ответа результата от сервера
  const onMessage = (frameStatus) => msgRenderer.render(frameStatus)

  // Коллбек, выполняемый до отправки кадра в LunaPass Backend
  const before = (img) => faceEstimator.estimate(img);

  const result = await lunapass.checkLiveness(onMessage, before);
  alert(result);

  msgRenderer.destroy();
  await lunapass.disconnectWS();
  await lunapass.detachCamera();
}
```

Приложение.

Приложение 1. История изменений

Дата	Версия	Описание
12.12.2021	1.0	Первичная версия документа