

VisionLabs LUNA ID

Инструкция по установке



Содержание

Глоссарий	3
Введение	5
Аппаратные и программные требования	6
Минимальные аппаратные требования	6
Требования к инструментам разработки	6
Операционные системы	6
Первоначальная настройка VL-LUNA-ID для Android	7
Скачивание дистрибутива	7
Предоставление учетных данных пользователя	8
Добавление файла .aar в качестве зависимости	8
Инициализация VL-LUNA-ID	9
Вызов функций VL-LUNA-ID	9
Первоначальная настройка VL-LUNA-ID для iOS	12
Скачивание дистрибутива	12
Добавление библиотек XCFramework	12
Включение проверки OneShotLiveness	13
Активация лицензионного ключа	13
Создание экрана распознавания лиц	14
Установка демонстрационного приложения LUNA ID Demo	15

Глоссарий

Термин	Определение
VisionLabs LUNA ID	Набор средств разработки (Software Development Kit, SDK), включающий в себя библиотеки и нейронные сети для анализа изображений в мобильном приложении.
VisionLabs LUNA PLATFORM	Автоматизированная система распознавания лиц и тел компании VisionLabs.
AGS	Абсолютное значение "качества" лица, определяемое нейронной сетью до извлечения дескриптора.
API	Набор готовых классов, процедур, функция, структур и констант, предоставляемых приложением (библиотекой, сервисом) или операционной системой для использования во внешних программных продуктах.
OSL	Программный способ, позволяющий подтвердить витальность (живучесть, жизненность) человека по одному или нескольким изображениям с целью предотвращения спуфинг-атак.
REST	Архитектурный стиль взаимодействия компонентов распределенного приложения в сети, основанный на протоколе HTTP. Обмен данными между клиентом и сервером происходит в форматах JSON, XML, HTML и т.д. REST также подразумевает, что каждый ресурс имеет уникальный идентификатор (URI) и может быть доступен через стандартные методы HTTP.
SDK	Набор программных инструментов, который позволяет разработчикам создавать приложения для определенной платформы.
БО	Нормализованное (центрированное и обрезанное) изображение, полученное после детекции объекта (лица или тела), предшествующее извлечению биометрического шаблона.
БШ	Набор данных в закрытом, двоичном формате, подготавливаемый системой распознавания на основе анализируемой характеристики.
Лучший кадр	Кадр видеопотока, на котором лицо зафиксировано в оптимальном ракурсе для дальнейшей обработки.
Угол наклона	Угол вращения вокруг горизонтальной оси X, направленной слева-направо.
Угол поворота	Угол вращения вокруг вертикальной оси Y.
Угол отклонения	Угол вращения вокруг горизонтальной оси Z, направленной вперед.

Сокращённое название	Расшифровка
LP	VisionLabs LUNA PLATFORM
API	API
VL-LUNA-ID	VisionLabs LUNA ID
OSL	OneShotLiveness
SDK	Software Development Kit
REST	Representation State Transfer
БО	Биометрический образец
БШ	Биометрический шаблон

Введение

VL-LUNA-ID предназначен для решения следующих задач:

- Получение лучшего кадра:
 - Оценка лучшего кадра по следующим критериям:
 - количество лиц в кадре;
 - размер ограничивающей рамки обнаружения лиц;
 - расстояние от краев кадра до ограничивающей рамки обнаружения лиц;
 - состояние глаз (открыты, закрыты или перекрыты);
 - положение головы (углы наклона, поворота и отклонения);
 - AGS;
 - качество изображения (слишком большая экспозиция, недостаточная экспозиция и размытость).
 - Отправка изображения с обнаруженным лицом в LUNA PLATFORM 5 для выполнения проверки OneShotLiveness на бэкенде.
 - Отправка исходных изображений в LUNA PLATFORM 5 для сопоставления биометрических шаблонов на бэкенде.

Данный документ описывает первоначальную настройку VL-LUNA-ID и установку демонстрационного приложения LUNA ID.

Демонстрационное приложение LUNA ID Demo предназначено для демонстрации возможностей VL-LUNA-ID для реализации сценариев регистрации и авторизации.

Данный документ включает в себя информацию о первоначальной настройке VL-LUNA-ID и установке демонстрационного приложения LUNA ID Demo.

Аппаратные и программные требования

Минимальные аппаратные требования

Следующие минимальные системные требования необходимы для работы с VL-LUNA-ID:

- CPU на Android: arm64-v8a, armeabi-v7a, x86_64, x86, на iOS: arm64;
- RAM 400 MB и больше.

Требования к инструментам разработки

Для работы с VL-LUNA-ID необходимы Android SDK 21 или выше на Android или XCode 13.2 или выше на iOS.

Операционные системы

VL-LUNA-ID поддерживает операционные системы Android 5.0 и выше, iOS 13 и выше.

Первоначальная настройка VL-LUNA-ID для Android

Скачивание дистрибутива

Дистрибутив представляет собой архив **lunaid-core-1.7.4**, где 1.7.4 – это числовой идентификатор, обозначающий версию VL-LUNA-ID.

Архив включает в себя минимальный набор библиотек и нейронных сетей, необходимый для встраивания VL-LUNA-ID в мобильное приложение.

Переместите архив в необходимую директорию.

Укажите учетные данные пользователя в файле `local.properties`.

Добавьте следующий фрагмент кода в блок `repositories` файла `settings.gradle.kts`:

Файл `settings.gradle.kts` расположен в корневом каталоге вашего проекта и определяет, какие проекты и библиотеки вам необходимо добавить в путь к классам вашего сценария сборки.

```
repositories {  
    ...  
  
    ivy {  
        url = java.net.URI.create("https://download.visionlabs.ru/")  
        patternLayout {  
            artifact ("releases/lunaid-[artifact]-[revision].[ext]")  
            setM2compatible(false)  
        }  
        credentials {  
            username = getLocalProperty("vl.login") as String  
            password = getLocalProperty("vl.pass") as String  
        }  
        metadataSources { artifact() }  
    }  
}
```

Предоставление учетных данных пользователя

Скачивать артефакты с <https://download.visionlabs.ru/> могут только авторизованные пользователи. Учетные данные пользователя приходят из VisionLabs.

Чтобы предоставить свои учетные данные пользователя, в файле `local.properties`:

Укажите свои учетные данные пользователя:

```
v1.login=YOUR_LOGIN  
v1.pass=YOUR_PASSWORD
```

Добавьте функцию получения логина и пароля:

```
fun getLocalProperty(key: String, file: String = "local.properties"): Any {  
    val properties = java.util.Properties()  
    val localProperties = File(file)  
    if (localProperties.isFile) {  
        java.io.InputStreamReader(java.io.FileInputStream(localProperties), Charsets.UTF_8).use { reader ->  
            properties.load(reader)  
        }  
    } else error("File from not found: '$file'")  
  
    if (!properties.containsKey(key)) {  
        error("Key not found '$key' in file '$file'")  
    }  
    return properties.getProperty(key)  
}
```

Мы рекомендуем добавить файл `local.properties` в `.gitignore`, поскольку система контроля версий не отслеживает этот файл.

Добавление файла `.aar` в качестве зависимости

Чтобы инициализировать VL-LUNA-ID в вашем проекте, добавьте файл `.aar` в качестве зависимости в файл `build.gradle.kts`. Файл `build.gradle.kts` определяет различные

VisionLabs LUNA ID: Инструкция по установке

параметры сборки, такие как зависимости, плагины, версии библиотек, параметры компиляции и тестирования и т. д. Все эти настройки влияют на то, как собирается проект и какой функционал он содержит.

Чтобы добавить файл .aar в качестве зависимости, добавьте следующий фрагмент кода в блок зависимостей файла build.gradle.kts:

```
dependencies {  
    ...  
    implementation("ai.visionlabs.lunaid:core:{VERSION}@aar")  
}
```

Вам необходимо обновить параметр `{VERSION}` при выпуске новой версии VL-LUNA-ID.

Инициализация VL-LUNA-ID

Укажите базовый класс `Application` и функцию `LunaID.init()` в файле build.gradle.kts:

```
class App : Application() {  
  
    override fun onCreate() {  
        super.onCreate()  
  
        LunaID.init(  
            app = this@App,  
            lunaConfig = LunaConfig.create(),  
            areDescriptorsEnabled = true  
        )  
    }  
}
```

Вызов функций VL-LUNA-ID

Чтобы использовать функции LUNA ID, например открыть камеру, отправить запрос в LUNA PLATFORM 5 и т. д., импортируйте библиотеки LUNA ID и укажите необходимые функции в файле build.gradle.kts, как показано в примере ниже:

```
import android.app.Application
```

VisionLabs LUNA ID: Инструкция по установке

```
import ru.visionlabs.sdk.lunacore.LunaConfig
import ru.visionlabs.sdk.lunacore.LunaCoreConfig
import ru.visionlabs.sdk.lunacore.LunaID

class DemoApp : Application () {
    override fun onCreate() {
        super.onCreate()

        LunaID.init(
            app = this@App,
            lunaConfig = LunaConfig.create(),
            areDescriptorsEnabled = true
        )

        LunaID.showCamera()

        LunaID.apiHuman

        // specify the URL to LUNA PLATFORM
        val baseUrl = "http://luna-platform.com/api/6/"
    }
}
```

Пример содержит следующие компоненты:

Компонент	Описание
LunaID.init()	Функция. Инициализирует библиотеку LUNA ID.
LunaID.showCamera()	Метод. Открывает камеру мобильного устройства.

VisionLabs LUNA ID: Инструкция по установке

<code>LunaID.apiHuman</code>	Свойство. Предоставляет доступ к API LUNA PLATFORM и позволяет отправлять запросы.
<code>baseUr1</code>	Переменная. Указывает URL-адрес LUNA PLATFORM, который используется функцией <code>LunaID.apiHuman()</code> .

Первоначальная настройка VL-LUNA-ID для iOS

Скачивание дистрибутива

Дистрибутив представляет собой архив **luna-id-sdk_ios_v.1.7.5**, где 1.7.5 – это числовой идентификатор, обозначающий версию VL-LUNA-ID.

Архив включает в себя минимальный набор библиотек и нейронных сетей, необходимый для встраивания VL-LUNA-ID в мобильное приложение.

Скачайте архив с <https://download.visionlabs.ru/>.

Скачивать артефакты с <https://download.visionlabs.ru/> могут только авторизованные пользователи. Учетные данные пользователя приходят из VisionLabs.

Переместите архив в необходимую директорию и разархивируйте его.

Добавление библиотек XCFramework

Переместите следующие библиотеки из дистрибутива в раздел **Frameworks, Libraries, and Embedded Content** в Xcode:

- **flower.xcframework**
Расположение файла: luna-id-sdk_ios_v.X.X.X\build\Release-iphonios\frameworks\flower.framework\
- **fsdk.xcframework**
Расположение файла: luna-id-sdk_ios_v.X.X.X\build\Release-iphonios\frameworks\fsdk.framework\
- **LunaAuth.xcframework**
Расположение файла: luna-id-sdk_ios_v.X.X.X\build\Release-iphonios\frameworks\LunaAuth.framework\
- **LunaCamera.xcframework**
Расположение файла: luna-id-sdk_ios_v.X.X.X\build\Release-iphonios\frameworks\LunaCamera.framework\
- **LunaCore.xcframework**
Расположение файла: luna-id-sdk_ios_v.X.X.X\build\Release-iphonios\frameworks\LunaCore.framework\
- **LunaWeb.xcframework**
Расположение файла: luna-id-sdk_ios_v.X.X.X\build\Release-iphonios\frameworks\LunaWeb.framework\
- **tsdk.xcframework**
Расположение файла: luna-id-sdk_ios_v.X.X.X\build\Release-iphonios\frameworks\tsdk.framework\

VisionLabs LUNA ID: Инструкция по установке

Убедитесь, что все файлы имеют метку **Embed**, чтобы они были включены в окончательное приложение. В противном случае ваше приложение выйдет из строя при запуске.

Включение проверки OneShotLiveness

Укажите параметры `verifyID` и `lunaServerURL` в объекте `LCLunaConfiguration` при запуске приложения, как показано в примере ниже:

Параметр	Описание
<code>verifyID</code>	Идентификатор верификатора, использованного для развертывания LUNA PLATFORM 5.
<code>lunaServerURL</code>	URL-адрес хоста LUNA PLATFORM 5. URL-адрес не должен иметь косую черту в конце.

```
func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {  
    ...  
    let configuration = LCLunaConfiguration.defaultConfig()  
    configuration.identifyHandlerID = "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXXX"  
    configuration.registrationHandlerID = "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXXX"  
    configuration.verifyID = "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXXX"  
    configuration.lunaAccountID = "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXXX"  
    configuration.lunaServerURL = URL(string: "https://LUNA_PLATFORM_HOST/6")  
    ...  
    return true  
}
```

Активация лицензионного ключа

Запросите значения параметров `Server`, `EID` и `ProductID` у VisionLabs и укажите их в файле `vllicense.plist`.

Добавьте файл `vllicense.plist` к вашему конечному приложению.

Лицензионный ключ будет сгенерирован и сохранен в указанной директории. Файл лицензии имеет двоичный формат. При следующем запуске мобильного приложения на том же устройстве лицензия будет прочитана из этого файла.

Создание экрана распознавания лиц

Добавьте метод `viewController()` в необходимую часть вашего приложения.

Укажите объект `LCLunaConfiguration` в качестве входного параметра. Он позволяет устанавливать различные пороговые значения, влияющие на итоговый экран распознавания.

VisionLabs LUNA ID: Инструкция по установке

Установка демонстрационного приложения LUNA ID Demo

Скачайте дистрибутив по ссылке, полученной из VisionLabs и установите демонстрационное приложение на ваше мобильное устройство.