VisionLabs
MACHINES CAN SEE

LUNA CARS.API

Инструкция по установке

ООО «ВижнЛабс»
123458, г. Москва, ул. Твардовского д. 8, стр. 1
+7 (499) 399 3361
info@visionlabs.ru
www.visionlabs.ru

Оглавление

Глоссарий

Термин	Определение
Bbox (Bounding box)	Прямоугольник, ограничивающий пространство изображения с обнаруженным объектом (ТС, номерным знаком ТС).
EXIF	Стандарт, позволяющий добавлять к изображениям дополнительную информацию (метаданные), комментирующую этот файл, описывающий условия и способы его получения, авторство и т. п.
ГРЗ	Государственный регистрационный знак транспортного средства.
ТС	Транспортное средство.

Введение

Настоящий документ представляет собой руководство администратора сервиса CARS.API.

Руководство определяет порядок установки, настройки и администрирования сервиса.

Перед установкой и эксплуатацией сервиса рекомендуется внимательно ознакомиться с настоящим руководством.

Общие сведения

VisionLabs LUNA CARS – система, предназначенная для определения атрибутов транспортных средств и распознавания автомобильных номеров. Система состоит из трёх сервисов: CARS.Analytics, CARS.API и CARS.Stream.

VisionLabs CARS.API – это сервис для распознавания транспортных средств, который позволяет в режиме реального времени:

- определять марку и модель ТС;
- распознавать ГРЗ;
- определять принадлежность ТС к маршрутному транспорту, такси и спецтранспорту;
- определять категорию ТС;
- оценивать качество поступающих изображений;
- определять условия оформления штрафа для транспорта, нарушившего скоростной режим;
- определять условия оформления штрафа для транспорта, пересекшего сплошную линию;
- определять тип ТС;
- определять цвет ТС;
- извлекать дескриптор ТС;
- определять страну принадлежности ГРЗ.

Системные требования

Для работы сервиса CARS.API необходимо учитывать ряд требований и условий, указанных ниже. Список системных требований представлены в Таблице 1.

Таблица 1. Системные требования

Необходимый ресурс	**Рекомендовано**
Процессор (CPU)	Intel, не менее 4 физических ядер с тактовой частотой не менее 2,0 ГГц
Оперативная память (RAM)	Не менее 8 Гб DDR4
Объем свободного дискового пространства (HDD/SSD)	Не менее 10 Гб
Операционная система (OS)	CentOS 7.4 x86_64*
Поддержка инструкций	AVX 2
Программное обеспечение	Python 3.7; Nginx; Ansible-playbook;
Поддерживаемые версии браузеров	Microsoft Edge (версия 44.0 и выше); Mozilla Firefox (версия 60.3.0 и выше); Google Chrome (версия 50.0 и выше).

* В качестве операционной системы (при развертывание в Docker) может выступать любая другая подобная ОС с поддержкой Python 3.7 (Ubuntu, Debian и т. д.).

При использовании ресурсов видеокарты поддерживаются графические ускорители NVIDIA с версией CUDA 10.1 и объемом видеопамяти не менее 2 Гб. Поддерживаются архитектуры Pascal, Volta, Turning.

Приведенная выше конфигурация обеспечит работу программного комплекса в демонстрационных целях и не предназначена для коммерческих систем. Конфигурация продуктивного контура рассчитывается на основании предполагаемой нагрузки на систему. Требования к производительности серверной части зависят от количества запросов в единицу времени при нормированном времени ответа.

Требования к изображениям

Требования к поступающим изображениям ТС и ГРЗ:

- Изображения должны быть трехканальными (RGB) или черно-белыми;
- Формат изображения: JPEG, закодированный в стандарте Base64;
- Изображение не должно содержать EXIF-тегов.
- Ракурс съемки ТС и ГРЗ может быть любым кроме «отвесных», при котором камера находится над объектом;
- ТС и ГРЗ должны быть целиком видны на кадре;

- Поддерживаемый размер изображения от 320x240 до 1920x1080 px.

Требования к видео

Требования к видео с ТС и ГРЗ:

- Рекомендуемое разрешение - 1920x1080 px;
- Скорость потока должна быть постоянной;
- Поддерживаемый битрейт – 4096 Кб/сек;
- Скорость затвора (выдержка) – не ниже 1/200 *;
- Используемые протоколы передачи данных – TCP, RTCP.

> Данный набор параметров является минимально рекомендуемым при котором система работает эффективно. При понижении значений количество детекций может также снизиться, а при повышении – излишне нагружать систему.

* Скорость затвора подбирается исходя из скоростного режима потока автомобилей. Чем выше скорость потока, тем быстрее должен работать затвор.

1. Установка

1.1. Подготовка к установке

Дистрибутив CARS.API представляет собой архив «luna-cars_v.*.zip».

Архив содержит компоненты, необходимые для установки и эксплуатации сервиса. Архив не включает некоторые зависимости, которые входят в стандартную поставку репозитория CentOS и могут быть загружены из открытых источников в процессе установки.

> Дальнейшие действия необходимо выполнять под учетной записью суперпользователя (с root-правами).

Перед установкой поместите файлы дистрибутива и скрипты установки в отдельную папку на сервере, с которого будет производиться установка. В данной директории не должно быть других файлов дистрибутива и лицензии кроме целевых, используемых для установки конечного продукта.

Установка через Ansible и Docker происходит совместно с дистрибутивом CARS.Analytics.

1.2. Включение поддержки AVX или GPU

CARS.API может использовать для вычислений видеокарту или процессор. Для каждой из этих архитектур в комплекте поставки доступны модели с соответствующим постфиксом (_gpu.plan, _cpu.plan).

Проверить поддержку AVX2 можно выполнив команду:

```
lscpu | grep -o avx2
```

Для выбора архитектуры и задания её настроек используются параметры в конфигурационном файле /data/modelRunner.cfg. Описание параметров файла представлено в Таблице 2.

Таблица 2. Параметры конфигурационного файла «modelRunner.cfg»

#	Параметр	Описание
1	numThreads	Количество потоков, на которых запускается модель при использовании процессора.
2	planType	Тип архитектуры. Возможные значения: • cpu – использование центрального процессора; • gpu – вычисление с помощью процессора графического ускорителя.
3	gpuNumber	Порядковый номер используемой видеокарты.
4	numComputeStreams	Количество аллоцируемых потоков CUDA.

Результаты распознавания классификаторов «marka_taxi_mt» и «grz_ai_recognition_v2» могут быть представлены с помощью латинских символов или с помощью кириллицы. Параметр «useLatinCharacters» отвечает за выбор символов. Параметр располагается в файле /data/modelRunner.cfg. Возможные значения параметры «useLatinCharacters»:

- True – использование латинских символов;
- False – использование кириллицы.

1.3. Установка через Ansible

Необходимо установить пакет ansible, выполнив команды:

```
# обновление менеджера пакетов
yum update
# установка дополнительных репозиториев
yum install epel-release
# установка ansible
yum install ansible
```

1.3.1. Настройка SSH

Необходимо сгенерировать SSH-ключ и добавить его на целевой сервер. Для начала необходимо проверить и настроить SSH сервис:

```
# проверка работоспособности ssh сервиса
systemctl status sshd
# если сервис неактивен необходимо его запустить
systemctl start sshd
# если сервис не установлен необхоимо его установить
yum install -y openssh-server
```

После этого необходимо сгенерировать открытый SSH-ключ, выполнив команду:

```
# генерирование открытого SSH-ключа
ssh-keygen
```

При необходимости можно задать ключевую фразу или оставить ее пустой. Для копирования ключа на целевой сервер необходимо ввести команду:

```
# копирование открытого SSH-ключа на целевой сервер
ssh-copy-id username@hostname
```

где «username» - имя авторизованного пользователя, а «hostname» - IP-адрес целевого сервера.

> Данный способ не является единственно возможным. Вы можете использовать любой другой удобный способ для обеспечения SSH-доступа на целевой сервер.

1.3.2. Настройка конфигурационного файла «hosts»

В комплекте поставки находится конфигурационный файл «hosts». Этот файл расположен в директории /ansible. В разделе CARS.API необходимо задать внешний IP-адрес сервера куда будет устанавливаться приложение и где будет расположен балансировщик «nginx»:

```
#CARS.API
#Multiple hosts allowed
[api]
<IP_адрес>

#Only 1 host
[nginx]
<IP_адрес>
```

1.3.3. Настройка конфигурационного файла «all.yml»

Необходимо произвести настройку сервиса в файле «all.yml», расположенного в директории /ansible/group_vars. Описание параметров находится в Таблице 3.

Таблица 3. Параметры конфигурационного файла «all.yml»

#	Параметр	Описание
1	luna_cars_vers	Указывает имя архива. Например, luna-cars_v.0.0.12.
2	luna_cars_zip_location	В этом параметре задается путь до архива CARS.Stream. Например, /distr/api/<luna-cars_v.0.0.12.zip>.

1.3.4. Настройка маршрутизации и масштабирования

Перед установкой также необходимо произвести настройку портов сервиса, которые будут использоваться при работе. Сделать это можно в блоке, посвященном портам.

> Список всех портов по умолчанию и их назначение представлен в приложении 1.

Рекомендуется оставлять номер портов по умолчанию. Переменная «CARS_API_PORT_RANGE» позволяет задать количество портов, начиная с порта, заданного в переменной «CARS_API_PORT».

При установке автоматически будет создан конфигурационный файл балансировщика /etc/nginx/conf.d/cars-api.conf, в котором будут указаны все порты и правила обращения к ним. Пример наполнения этого файла представлен ниже:

```
upstream lunaCars {
least_conn;
server<IP_адрес>:8100;
server<IP_адрес>:8101;
keepalive 8;
}
server {
```

```
client_max_body_size 100m;
keepalive_timeout 600;
proxy_connect_timeout 600s;
proxy_read_timeout 600s;
proxy_send_timeout 600s;
listen 81 default_server;
server_name _;
location / {
proxy_pass http://lunaCars; }
}
```

1.3.5. Выбор страны

В файле «all.yml» доступен выбор режима распознавания стран ГРЗ. Настройка происходит в секции сетевых настроек:

```
#Will use Emirates GRZ classifiers (True)
#Will use default GRZ classifiers (False)
EMIRATES: False
```

Для выбора режима необходимо изменить значение параметра «EMIRATES»:

- True – система будет распознавать ГРЗ только ОАЭ;
- False – система будет распознавать ГРЗ остальных стран (РФ, СНГ, ЕС).

1.3.6. Запуск установки через Ansible

> В процессе установки будет отключен встроенный firewall и selinux (в дальнейшем потребуется перезагрузка).
> В некоторых случаях может появиться ошибка о недоступности репозиториев, в этом случае необходимо перезапустить установку.

Для запуска установки необходимо находиться в директории /ansible и выполнить команду:

```
ansible-playbook -i hosts install_api.yml
```

1.3.7. Проверка работоспособности

По окончании установки необходимо проверить состояние сервиса CARS.API командой:

```
systemctl status luna-cars-api
```

При корректной работе система не должна выдать сообщения об ошибке.

1.4. Установка при помощи Docker

1.4.1. Установка docker и docker-compose

При использовании CentOS можно воспользоваться официальной инструкцией, так как эта инструкция является самой актуальной.

```
    #Установка дополнительных зависимостей
    sudo yum install -y yum-utils
    #Добавление репозитория docker
    sudo yum-config-manager \
        --add-repo \
        https://download.docker.com/linux/centos/docker-ce.repo
    #Установка docker
    sudo yum install docker-ce docker-ce-cli containerd.io
    #Проверка корректности установки
    docker -v
    #Скачивание docker-compose
    sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
    #Назначение свойства исполняемого файла
    sudo chmod +x /usr/local/bin/docker-compose
```

1.4.2. Настройка файла окружения «.env»

Параметры файла окружения «.env» представлены в Таблице 4.

Таблица 4. Параметры файла окружения.

#	Параметр	Описание
1	HASP_license_server	Задает путь до сервера, к которому установщик обращается за доступной сетевой лицензией на продукт. Если сетевой лицензии нет, то она задается локально.
2	HASP_wait_time	Задает время ожидания ответа при отправке запроса о доступной лицензии на сервер. Задается в минутах.
3	Emirates	Выбор страны распознавания ГРЗ. Доступные значения: • true – система будет распознавать ГРЗ только ОАЭ; • false – система будет распознавать ГРЗ РФ, СНГ, ЕС.
4	ENG	Задает язык системы. Доступные значения: • true – английский язык системы; • false – русский язык системы.

1.4.3. Запуск установки

Скопируйте архив CARS.API в директорию /distr/api.

В корневой папке (где расположен docker-compose.yml) выполните команду

```
    docker -compose up –d
```

После изменения файла «.env» или иных параметров необходимо запускать систему с ключом пересборки:

```
    docker -compose up -d –build
```

Приложения

Приложение 1.

Таблица 5. Список используемых портов по умолчанию

Порт	Сервис	К порту обращается
34569	CARS.Stream	CARS.Analytics backend, Пользователь (stream preview)
81	Nginx перед CARS.API	CARS.Stream, CARS.Analytics backend
8100+	Начальный порт CARS.API	Nginx
8000	CARS.Analytics backend	CARS.Analytics frontend, CARS.Stream, Пользователь (admin)
8080	CARS.Analytics frontend	Пользователь (Графический интерфейс)
1947	HASP	CARS.Stream
5432	Postgre SQL	CARS.Analytics backend
6379	Redis	CARS.Analytics backend

Приложение 2. История изменений.

Дата	Версия	Описание
05.08.21	1.1	Полная ревизия и обновление документа
29.01.21	1.0	Первичная версия документа