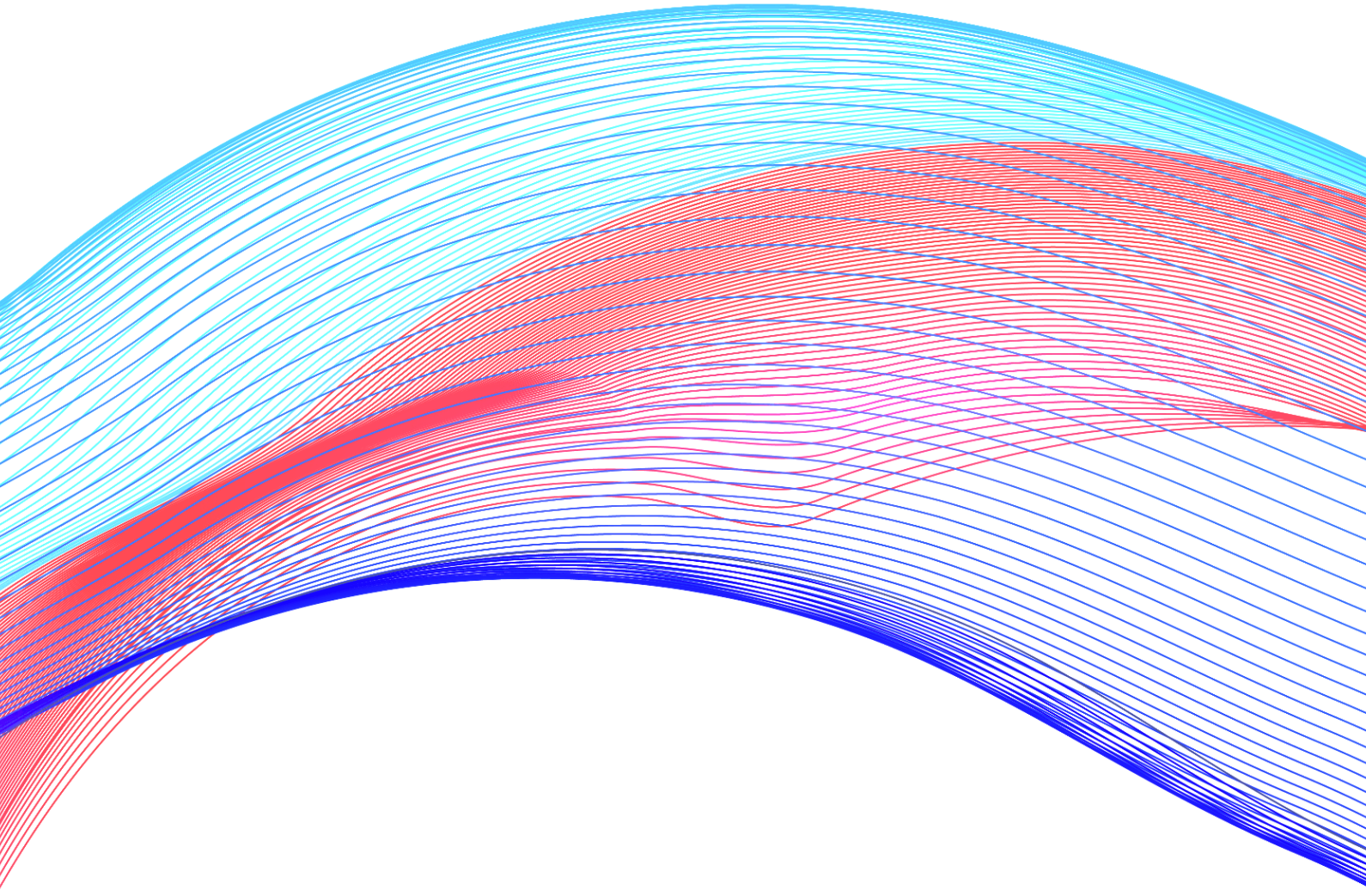
ООО “ТРИДИВИ”

Image API 1.0.1

Руководство администратора



2023 г.

Содержание

Форма поставки

Форма поставки Image API – дистрибутив. В дистрибутиве лежат файлы необходимые для развертывания Image API (директория `setup`) и документация в формате pdf. Ссылка на скачивание дистрибутива и прикрепленный файл лицензии передаются клиенту по электронной почте.

1. Требования

1.1. Минимальные системные требования

Аппаратные средства:

ЦП:

- 6 ядер (12 потоков)
- частота - 2,8 ГГц
- расширения набора команд: AVX/AVX2

ОЗУ:

- объем 6 ГБ (при запуске одного экземпляра каждого сервиса)

SSD / HDD:

- свободное пространство от 30 ГБ
- не менее 20% свободного места в файловой системе

Программное обеспечение:

- ОС Ubuntu 22.04
- Docker версии - 20.10.17
- Kubernetes версии - 1.23.8
- Helm версии - 3.10.2

Примечание: Docker, Kubernetes, Helm можно установить командой из пункта 2.1.3 (требуется подключение к интернету).

2. Инструкция по развертыванию

2.1. Подготовка к развертыванию

2.1.1 Подготовка

Переместите файл лицензии `face_sdk.lic` в папку `setup` распакованного дистрибутива.

Дальнейшие команды следует выполнять относительно директории `setup`.

2.1.2 Конфигурация

Описание конфигурационных файлов дистрибутива:

- `./deployment.cfg` - файл конфигурации развертывания экземпляра Image API;
- `./dependencies.cfg` - файл конфигурации зависимостей для установки кластера kubernetes;
- `./cluster.cfg` - файл конфигурации Kubernetes кластера.

Откройте файл `./cluster.cfg`, используя текстовый редактор, и установите значения следующих переменных:

- `apiserver_advertise_address` - внутренний IP-адрес машины, на которой выполняется развертывание.
- `external_ip_address` - публичный IP-адрес машины, на который будет настраиваться DNS.

Примечание: при отсутствии публичного IP-адреса используйте внутренний IP-адрес.

Откройте файл `./deployment.cfg`, используя текстовый редактор, и установите значения следующих переменных:

- `domain` - доменное имя.

IP-адрес для доменного имени `<domain>` должен быть сконфигурирован на DNS-сервере (См. подробнее в пункте 2.2.4 данного руководства).

2.1.3 Установка зависимостей

Если в системе отсутствует Docker, Kubernetes, Helm выполните следующую команду:

```
$ ./cli/dependencies/install.sh
```

2.1.4 Загрузка образов

Загрузите в локальный `registry` docker-образы:

```
$ ./cli/load-images.sh
```

Загрузка образов может длиться около пяти минут.

2.1.5 Установка и настройка кластера

Запустите команду для создания и настройки кластера:

```
$ ./cli/cluster/install.sh
```

Эта команда выполняет следующие действия:

1. Инициализация узла для развертывания кластера
2. Настройка сети
3. Установка ingress-controller
4. Создание “секретов” Kubernetes

2.1.6 Проверка работоспособности кластера

После инициализации главного узла убедитесь, что он готов к работе. Для проверки выполните следующую команду:

```
$ kubectl get nodes
```

В результате в терминале будет отображен следующий вывод:

```
NAME          STATUS      ROLES                   AGE     VERSION
master-node   Ready       control-plane,master    1d     v1.23.8
```

2.2. Развертывание

2.2.1 Настройка файла конфигурации (опционально)

При необходимости перед установкой Image API в кластер можно указать значения для следующих полей в файле конфигурации `deployment.cfg`:

- **`image_pull_policy`:** регулирует политику загрузки из `registry` образов контейнера для развертывания экземпляра деплоймента Image API. Доступны следующие значения поля:
  - `Always` - kubelet (агент узлов в Kubernetes) выполняет загрузку образа из `registry`;
  - `Never` - загрузка образа из `registry` не предусмотрена. Если образ уже присутствует локально, kubelet попытается выполнить запуск контейнера, в противном случае, произойдет сбой загрузки;
  - `IfNotPresent` - образ загружается только в случае, если отсутствует локально.

- **`capturer_config`:** в значении поля указывается название файла конфигурации выбранного детектора лиц.

  В Image API используются детекторы лиц из 3DiVi Face SDK - набора библиотек для разработки решений по распознаванию лиц.

  В текущей версии Image API доступны на выбор два детектора: ULD и REFA. Детектор ULD позволяет выполнять детекцию лиц разного размера, в том числе лиц в масках (скорость детекции выше чем у REFA). Детектор REFA обеспечивает детекцию лиц с наибольшим покрытием углов поворота и наклона головы и максимальным качеством, включая детекцию лиц в масках (качество детекции выше чем у ULD).

  В качестве значений для поля `capturer_config` можно указать следующие файлы конфигурации детекторов:

  - `common_capturer_uld_fda.xml` (Файл конфигурации детектора ULD)
  - `common_capturer_refa_fda_a.xml` (Файл конфигурации детектора REFA)

  Более подробно о файлах конфигурации детекторов лиц смотрите в документации 3DiVi Face SDK.

- **`recognizer_config:`** в значении поля указывается название файла конфигурации рекогнайзера лиц. Image API использует набор рекогнайзеров из 3DiVi Face SDK. В каждом рекогнайзере применяется определенный метод идентификации. Всего в 3DiVi Face SDK предусмотрено несколько методов идентификации, отличающихся по характеристикам качества распознавания и времени работы.

  Для Image API можно указать следующие файлы конфигурации рекогнайзеров:

  - `method11v1000_recognizer.xml`
  - `method12v30_recognizer.xml`
  - `method12v50_recognizer.xml`
  - `method12v100_recognizer.xml`
  - `method12v1000_recognizer.xml`

  Более подробно о рекогнайзерах, методах идентификации и файлах конфигурации рекогнайзеров смотрите в документации 3DiVi Face SDK.

- **`enable_use_avx2:`** использование набора команд AVX2 позволяет ускорить работу рекогнайзеров. Для включения/выключения AVX2 укажите следующие значения для поля: “1” (включено) или “0” (выключено).
- **`downscale_rawsamples:`** принимаемые значения поля: “1” (включено) или “0” выключено). По умолчанию указано значение “0”. При включенном поле все сэмплы уменьшаются до предпочтительного размера в целях уменьшения потребления памяти, при этом происходит снижение производительности. Рекомендуется оставить поле отключенным.

2.2.2 Установка

Запустите скрипт для установки Image API в кластер:

```
$ ./cli/deployment/install.sh
```

Для получения статуса работы развертывания выполните команду:

```
$ kubectl get pods
```

В терминал будет выведен список *под* (экземпляров деплоймента), их статус, количество перезапусков и время с момента создания.

Пример вывода:

```
NAME                                                READY     STATUS      RESTARTS AGE
image-api-v1-0-0-age-estimator-dep-5bf7889dcd-jm46h      1/1     Running        0      1h
```

```
image-api-v1-0-0-body-detector-dep-6778495696-4zkpm          1/1     Running        0       1h
image-api-v1-0-0-emotion-estimator-dep-7c8cdd7c6b-z4vlv      1/1     Running        0       1h
image-api-v1-0-0-face-detector-face-fitter-dep-5fb48dcb5…    1/1     Running        0       1h
image-api-v1-0-0-gateway-dep-65d4846c59-g2nfj                1/1     Running        0       1h
image-api-v1-0-0-gender-estimator-dep-86b9457f9b-2464q       1/1     Running        0       1h
image-api-v1-0-0-face-detector-liveness-estimator-dep-7bd…   1/1     Running        0       1h
image-api-v1-0-0-mask-estimator-dep-76b958b84b-w7pss         1/1     Running        0       1h
image-api-v1-0-0-quality-assessment-estimator-dep-fbf8bc4…   1/1     Running        0       1h
image-api-v1-0-0-face-detector-template-extractor-dep-5758… 1/1     Running        0       1h
image-api-v1-0-0-verify-matcher-dep-6d66dc4948-lx8j7         1/1     Running        0       1h
```

Примечание*:* Наличие у всех *под* статуса `Running` означает, что все контейнеры успешно запущены.

Ниже приведено краткое описание деплойментов:

- `image-api-v1-0-0-gateway-dep`: обратный прокси-сервер, отвечает за единый доступ к сервисам обработки;
- `image-api-v1-0-0-face-detector-face-fitter-dep`: детекция лица и вычисление антропометрических точек лица и углов наклона/поворота головы;
- `image-api-v1-0-0-body-detector-dep`: детекции тела;
- `image-api-v1-0-0-emotion-estimator-dep`: оценка эмоций;
- `image-api-v1-0-0-age-estimator-dep`: оценка возраста;
- `image-api-v1-0-0-gender-estimator-dep`: оценка пола;
- `image-api-v1-0-0-face-detector-liveness-estimator-dep`: оценка принадлежности лица на изображении реальному человеку (liveness);
- `image-api-v1-0-0-mask-estimator-dep`: оценка наличия/отсутствия медицинской маски на лице человека;
- `image-api-v1-0-0-quality-assessment-estimator-dep`: оценка качества изображения;
- `image-api-v1-0-0-verify-matcher-dep`: сравнение двух биометрических шаблонов;
- `image-api-v1-0-0-template-extractor-dep`: извлечение биометрического шаблона лица с изображения.

2.2.3 Масштабирование

В случае, когда нагрузка возрастает, для стабилизации работы Image API предусмотрено масштабирование описанных в пункте 2.2.2 деплойментов в ручном режиме:

Для масштабирования установленного релиза необходимо выполнить следующую команду:

```
$ kubectl scale deployment <deployment_name> --replicas <count>
```

где `<deployment_name>` - наименование деплоймента (например, `face-detector-liveness-estimator-dep`), а `<count>` - количество реплик деплоймента (под).

Для сохранения состояния масштабирования необходимо установить значение **replicas** под ключем интересующего сервиса, в файле `setup/chart/processing-services.json` и перезапустить развертывание командой:

```
$ ./cli/deployment/install.sh
```

2.2.4 Настройка DNS

Для обеспечения доступа DNS сервер вашей сети должен содержать запись о том, что домен `<domain>` доступен по адресу `<external_ip_address>`. Значения переменных можно получить из файлов `./setup/deployment.cfg` и `./setup/cluster.cfg`, заполненных в пункте 2.1.3. Чтобы выполнить данную конфигурацию, обратитесь к администратору вашей сети.

Для целей тестирования можно указать IP-адрес и домен в файле `/etc/hosts` на Linux или `C:\Windows\System32\drivers\etc\hosts` на Windows. Для этого добавьте в конец данного файла новую строку вида `<external_ip_address> <domain>`, подставив значения соответствующих переменных, и сохраните файл. Обратите внимание, что для редактирования файла `hosts` необходимо обладать правами администратора.

Для использования Image API с той же машины, где выполнено развертывание, можно воспользоваться скриптом, который автоматически добавит необходимую запись в файл `/etc/hosts`.

```
$ ./cli/add-dns.sh
```

2.2.5 Проверка работы

Для проверки работы следует воспользоваться командой:

```
$ ./cli/test.sh
```

Вывод теста должен иметь следующий вид:

```
face-detector-face-fitter ✓
age-estimator ✓
emotion-estimator ✓
gender-estimator ✓
face-detector-liveness-estimator ✓
mask-estimator ✓
quality-assessment-estimator ✓
face-detector-template-extractor ✓
body-detector ✓
face-detector-template-extractor ✓
face-detector-template-extractor ✓
verify-matcher ✓
```

2.2.6 Удаление

Для удаления Image API выполните следующую команду:

```
$ ./cli/deployment/delete.sh
```

2.3 Лицензирование

2.3.1 Установка лицензии

Распакуйте дистрибутив Image API и переместите файл лицензии face_sdk.lic (файл прикреплен к электронному письму) в папку `setup`.

2.3.2 Обновление лицензии

1. Удалите объект типа secret, содержащий файл лицензии, из кластера Kubernetes.

```
$ kubectl delete secret face-sdk
```

2. Установите секрет в кластер, где ./face_sdk.lic - путь до файла лицензии.

```
$ face_sdk_lic_path=./face_sdk.lic ./cli/cluster/install/secrets.sh
```

3. Удалите релиз из кластера.

```
$ ./cli/deployment/delete.sh
```

4. Установите релиз в кластер.

```
$ ./cli/deployment/install.sh
```

3. Доступ к Image API

Для ознакомления с работой сервисов Image API предусмотрен специальный веб-интерфейс, расположенный по адресу *<image_api_url>*, откуда можно получить доступ к Swagger-документации по сервисам.

Адрес *<image_api_url>* имеет вид *http:///<domain>,* где *<domain>* - доменное имя, указанное в пункте 2.1.2.

Для прямого доступа к Swagger-документации конкретного сервиса перейдите по ссылке *<image_api_url>/<service_name>/doc*s, где *image_api_url* - адрес развернутого продукта, а *service_name* - имя сервиса.

Список имен сервисов:

- face-detector-face-fitter
- face-detector-liveness-estimator
- face-detector-template-extractor
- age-estimator
- emotion-estimator
- gender-estimator
- mask-estimator
- quality-assessment-estimator
- body-detector
- verify-matcher

Примечание: Если вы хотите получить помощь по установке, задать вопросы или оставить отзыв, обращайтесь в службу поддержки 3DiVi по адресу support-image-api@3divi.com.