



Система автоматического распознавания лиц IVA CV



Версия 1.8

Руководство администратора

АННОТАЦИЯ

В данном программном документе приведено руководство администратора программного обеспечения IVA Cognitive Video (IVA CV) (далее – Программа), предназначенного для обнаружения лиц людей в видеопотоках с онлайн-камер или в загружаемых видеофайлах, получения биометрических данных на основе обнаружений/детекций лиц, сравнения полученных биометрических данных с имеющимися в Программе и выдачи сигнала о выявлении совпадений.

В документе приводится необходимая информация по установке и настройке Программы, а также сведения по администрированию Программы.

СОДЕРЖАНИЕ

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ	4
1. ОБЩИЕ СВЕДЕНИЯ	5
1.1. Назначение программного обеспечения	5
1.2. Функции программного обеспечения	5
1.3. Состав Программы	6
1.4. Требования к уровню подготовки администраторов Программы	6
1.5. Требования к используемому оборудованию	6
1.6. Средства, необходимые для функционирования Программы	7
2. ОПИСАНИЕ РАБОТЫ И СТРУКТУРЫ ПРОГРАММЫ	8
2.1. Структура программы с описанием функций составных частей	8
3. УСТАНОВКА ПРОГРАММЫ	15
3.1. Начальные этапы	15
3.2. Установка компонентов Программы	15
3.2.1. Подготовка к установке	15
3.2.2. Установка Программы	15
4. КОНФИГУРИРОВАНИЕ СИСТЕМЫ	17
5. НАСТРОЙКИ	36
5.1. Настройки доступа	36

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Администратор	Технический специалист, занимающийся администрированием и обеспечением рабочего состояния программных продуктов.
Эталонное изображение	Изображения, координаты объектов на которых известны и зафиксированы в какой-либо системе координат.
АРМ	Автоматизированное рабочее место.
Ubuntu	Операционная система, основанная на операционной системе Linux с открытым исходным кодом.
FFmpeg	Свободная библиотека с открытым исходным кодом, позволяющая записывать, конвертировать и передавать цифровые аудио- и видеозаписи в различных форматах
Интерфейс	Комплекс средств, предназначенных для взаимодействия двух систем друг с другом
Битрейт	Количество бит, используемых для передачи/обработки данных в единицу времени
Биометрические данные	Уникальные биологические и физиологические характеристики, которые позволяют установить личность человека.
Аффинные преобразования	Отображение плоскости или пространства в себя, при котором параллельные прямые переходят в параллельные прямые.

1. ОБЩИЕ СВЕДЕНИЯ

1.1. Назначение программного обеспечения

Программа IVA Cognitive Video (IVA CV) – это программа автоматического распознавания лиц, предназначенная для обнаружения лиц людей в видеопотоках с онлайн-камер или в загружаемых видеофайлах, получения биометрических данных на основе обнаружений/детекций лиц, сравнения полученных биометрических данных с имеющимися в Программе и выдачи сигнала о выявлении совпадений.

1.2. Функции программного обеспечения

Программа распознавания лиц в фото и видеопотоках позволяет реализовать проекты по биометрической видеоаналитике, предоставляя следующий функционал:

- детекция лиц (в реальном времени с сохранением всех лиц, попавших в поле зрения камер);
- идентификация лиц;
- детектор наличия медицинской маски на лице;
- бесконтактное измерение температуры тела человека с идентификацией объекта измерения;
- запись и хранение видеоархива с разметкой по событиям;
- работа с результатами распознавания в режиме реального времени и исторического поиска;
- система уведомлений и сигналов тревоги при (не)обнаружении определенных лиц;
- инструмент оценки качества фотоизображений;
- инструмент поиска дубликатов в базах изображений лиц;
- поиск в накопленной истории по фотографии.

1.3. Состав Программы

Программа состоит из набора модулей (сервисов), устанавливаемых на один или несколько серверов и веб-интерфейса пользователя для доступа к функциям Программы.

1.4. Требования к уровню подготовки администраторов Программы

Администратор Программы должен обладать следующими навыками:

- установки и администрирования Ubuntu;
- подключения и настройки камер видеонаблюдения.

1.5. Требования к используемому оборудованию

Программа устанавливается на ПК/сервер, характеристики которого подбираются индивидуально в зависимости от количества подключенных камер и прогнозируемого количества лиц в обрабатываемом секторе.

Для инсталляции Программы используется вычислительное оборудование с характеристиками (минимальными требованиями), вычисляемыми исходя из:

- работы с видеопотоками или фотографиями;
- количества камер;
- разрешения видеопотока;
- количества анализируемых кадров в секунду (fps);
- интенсивности (RPS) – количества лиц в поле зрения каждой камеры одновременно;
- и других метрик.

При работе с Программой используется АРМ оператора (ПК). К АРМ предъявляются минимальные технические требования – возможность запустить браузер Google Chrome (не ниже версия 88.0.4324.150) и оперативная память 1 GB.

1.6. Средства, необходимые для функционирования Программы

Системные программные средства, используемые программой, должны быть представлены операционной системой Linux Ubuntu (не ниже версии 18.04) с разрядностью 64bit.

2. ОПИСАНИЕ РАБОТЫ И СТРУКТУРЫ ПРОГРАММЫ

2.1. Структура программы с описанием функций составных частей

Структура программы отображена на рисунке 1. Программные компоненты Программы с описанием их функций приведены в таблице 2.

Сторонние утилиты для работы и запуска Программы, не влияющие на выполнение функциональных требований, входящие в комплект поставки приведены в таблице 3.

Программа функционирует следующим способом:

- 1) Запускается Программа, интерфейс Программы;
- 2) Через интерфейс Программы подключается онлайн-камера (или несколько камер) или загружается видеофайл;
- 3) Изображение с онлайн-камеры или видеофайл декодируется Компонентом декодирования (Reader) при анализе видеофайла – файл сохраняется в архиве;
- 4) Декодированные кадры обрабатываются Компонентом захвата (первая нейронная сеть – Detector), на данном этапе на кадре обнаруживаются области, где находятся лица, вычисляются 5 контрольных точек на лицах (глаза, нос, уголки рта). Результат работы компонента захвата отображаются в интерфейсе с помощью Плеера. Detector работает на базе свёрточной нейронной сети;
- 5) Обработанные на предыдущих этапах кадры выравниваются и подгоняются по нужному размеру (112x112 пикселей) Компонентом предварительной обработки (Aligner) с помощью аффинных преобразований (без изменения естественных пропорций лица);
- 6) Выравненные кадры лиц с отмеченными контрольными точками отдаются в Компонент извлечения биометрического шаблона (вектора лица), далее эти предобработанные данные и идут в трекер. Компонент извлечения биометрического шаблона, это вторая нейронная сеть – Embedder;

Компоненты декодирования, захвата, предварительной обработки, извлечения биометрического шаблона и плеер представляют собой отдельный функциональный модуль Pipeline (конвейер обработки), который запускается компонентом управления камерами (Менеджер камер – Camera-manager) отдельно для каждой камеры. Начиная с версии 1.8 компонент извлечения биометрического шаблона может не входить в состав Camera-manager, а функционировать как отдельный сервис.

7) Выравненные кадры масок с отмеченными контрольными точками отдаются в Компонент извлечения биометрического шаблона (вектора лица), далее эти предобработанные данные и идут в трекер.

8) Изображения jpeg и биометрические данные (вектора) идут в Компонент трекинга (Tracker). В котором, на основе векторов, разработанными алгоритмами трекинга формируются наборы кадров, относящихся к одному и тому же лицу (треки). Кадры треков в jpeg записываются на жесткий диск для дальнейшего хранения и использования. Количество кадров для хранения указывается в настройках Программы (см. Руководство оператора и Руководство администратора);

9) В момент формирования трека, вектора лиц отправляются по http в Компонент поиска (Search) для сравнения с базой изображений и описывающих их векторов (списки людей) для идентификации;

10) Компонент сравнения (Search) отправляет обратно в Компонент трекинга (Tracker) факты идентификации и suspect id (уникальный номер человека);

11) Событийный компонент Notifier берет результаты идентификации из Трекера и транслирует их в WebSocket, а также записывает данные (url, вектора, suspect id) в базу данных Программы;

12) Фронтенд через Notifier узнает, что произошло событие (факт обнаружения, распознавания и др.) и при необходимости обращается в базу данных.

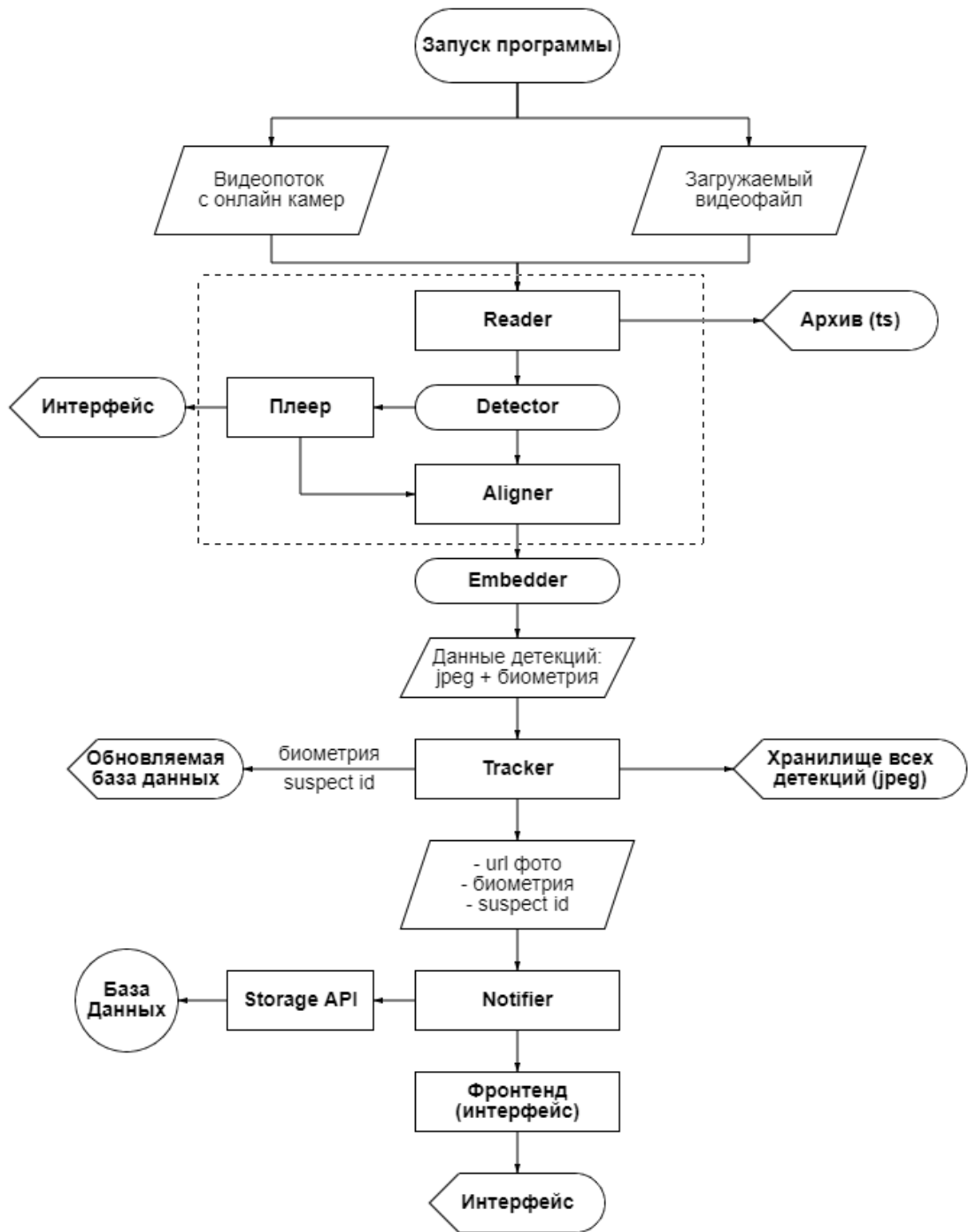


Рисунок 1 – Базовый алгоритм работы Программы

Таблица 2 – Функциональные компоненты Программы

	Компонент	Функции
Pipeline		
1	Компонент декодирования (Reader)	Декодирует видеопоток или видеофайл для дальнейшей обработки Компонентом захвата.
2	Компонент захвата (Detector)	Первая нейронная сеть. Распознает, обнаруживает лица на всех кадрах. Detector работает на сети собственной разработки ivaface.
3	Плеер	Проигрывает (в реальном времени) видео с онлайн-камер после обработки детектером.
4	Компонент предварительной обработки (Aligner)	Определяет контрольные точки на вырезанных изображениях лиц, выравнивает.
5	Компонент извлечения биометрического шаблона (Embedder)	Вторая нейронная сеть. Извлекает биометрический шаблон (вектор) лица. Работает на сетях ResNet50, ResNet100.
Отдельно		
6	Компонент трекинга (Tracker)	Формирует на базе поступивших биометрических шаблонов (векторов), разработанными алгоритмами трекинга наборы кадров, относящихся к одному и тому же лицу (треки).
7	Компонент поиска (Search)	Сравнивает собранные треки с базой изображений и описывающих их векторов (списки подозреваемых) для идентификации.

	Компонент	Функции
8	Нотифаер (Notifier)	«Событийный» компонент Программы. При возникновении событий (появление человека в поле зрения камеры, идентификация человека из списка и др.) отправляются Нотифаером в WebSocket.
9	Менеджер камер (Camera-manager)	Осуществляет управление онлайн-камерами, постоянный мониторинг настроек камер и данных от pipeline об обработке видеопотока.
10	Фронтенд	Клиентская сторона пользовательского интерфейса
11	База данных	Долгосрочное хранение данных
12	Транспортный протокол	Осуществляет связь и передачу данных между компонентами программы
13	Видеоменеджер (Video-manager)	Осуществляет управление видеофайлами, постоянный мониторинг настроек и данных от pipeline об обработке видеофайлов.
14	Сервер авторизации	Отвечает за контроль доступа к модулям ПО (выделяет токены, проверяет их на валидность). В дальнейшем будет отвечать за распределение прав доступа

Таблица 3 – Сторонние утилиты

№	Наименование модуля	Тип лицензии	Исполняемый файл, в который входит утилита при установке. Функции.
1	nvidia sdk	GPL v3	ivacv_os_cuda10.0_rc1_4_0.tar.gz

			Набор средств разработки и библиотек для работы видеопроцессоров.
2	Redis server	BSD 3-Clause	ivacv_redis_rc1_2_0.tar.gz Нереляционная высокопроизводительная СУБД.
3	MySQL server	GPL	ivacv_mysql_rc1.2.5.tar.gz Свободная реляционная система управления базами данных.
4	PostgreSQL		Свободная объектно-реляционная система управления базами данных.
5	numpy	BSD 3-Clause	ivacv_os_cuda10.0_rc1_4_0.tar.gz Библиотека для языка программирования Python с возможностями поддержки многомерных массивов, высокоуровневых математических функций.
6	OpenCV	BSD 3-Clause	ivacv_os_cuda10.0_rc1_4_0.tar.gz Библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов
7	Apache MXNet	Apache License 2.0	ivacv_os_cuda10.0_rc1_4_0.tar.gz Быстро масштабируемая система обучения и получения логических выводов.
8	Tensorflow	Apache License 2.0	ivacv_os_cuda10.0_rc1_4_0.tar.gz Программная библиотека для машинного обучения для решения задач построения и тренировки нейронной сети с целью автоматического нахождения и классификации образов.
9	tvm	Apache License 2.0	ivacv_os_cuda10.0_rc1_4_0.tar.gz Стек компиляторов глубокого обучения для процессоров и графических процессоров. Для компиляции Tensorflow, Darknet.

10	cupy	MIT	<code>ivacv_os_cuda10.0_rc1_4_0.tar.gz</code> Расширение языка Python, добавляющее поддержку многомерных массивов и матриц, вместе с библиотекой высокоуровневых математических функций для операций с этими массивами.
11	Nginx	BSD 3-Clause	<code>ivacv_nginx_rc1_2_0.tar.gz</code> . Веб-сервер

3. УСТАНОВКА ПРОГРАММЫ

3.1. Начальные этапы

Установка Программы включает в себя:

1. Установку ОС Ubuntu или ее настройку;
2. Создание и загрузка на сервер необходимых данных;
3. Запуск скрипта, устанавливающего драйверы (см. пункт 3.2.2.пп2);
4. Запуск скрипта, устанавливающего ПО;
5. Конфигурирование Программы.

3.2. Установка компонентов Программы

3.2.1. Подготовка к установке

После установки ОС Ubuntu, скопируйте и распакуйте установочный архив в директорию.

3.2.2. Установка Программы

После распаковки установочного архива перейдите в папку, куда он был распакован (в папку «IVACV» как пример).

Для установки необходимого окружения Программы запустите скрипт «setup» в папке «gen_bash» следующим образом:

1. Перейдите в директорию со скриптом (где «home [PATH]» — путь до папки с образцами):

```
cd ~/IVACV/gen_bash
```

2. Установите окружение:

```
sudo ./setup -home ~/IVACV -mini
```

3. Создайте необходимые конфигурационные файлы:

```
sudo ./setup -home ~/IVACV -env
```

```
sudo ./setup -home ~/IVACV -dc
```

4. Загрузите образы ПО:

```
sudo ./setup --home ~/IVACV --push
```

5. Далее вернитесь в каталог IVACV (`cd ~/IVACV`) и проведите конфигурирование программы и отредактируйте файл:

```
docker-compose -comment.yml  
файл.env
```

Далее вернитесь в каталог IVACV (`cd ~/IVACV`) и выполните команду:

```
docker-compose up -d
```

Для запуска и работы с Программой в браузере Chrome введите в адресную строку:

«IP сервера»: 18080

После запуска Программы, в интерфейсе откройте Настройки (см. пункт 4.1 данного Руководства) и в настройках во всех полях замените localhost на IP сервера.

4. КОНФИГУРИРОВАНИЕ СИСТЕМЫ

Конфигурирование Программы – это установка необходимых значений параметров, перечисленных в таблицах 4-16.

Таблица 4 – Manager options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
api-url API_URL	API URL / URL-адрес		
chain-template CHAIN_TEMPLATE	Path to the json chain template file / Путь к файлу шаблона		
chain-template CHAIN_TMPDIR	Path to directory to store temporary chains in / Путь к каталогу хранения временных цепочек		
state-tmpdir STATE_TMPDIR	Path to directory to store temporary state files in / Путь к каталогу хранения временных файлов		
update-delay UPDATE_DELAY	Handler update delay / Задержка обновления функции (обработчика)	[float]	
max-start-together MAX_START_TOGETHER	Maximum number of handlers to start / Максимальное число запускаемых обработчиков	[int]	no limit

Таблица 5 – General handler arguments

Параметр	Описание	Ограничение по значению	Значение по умолчанию
disable-gpu	Do not use GPU devices at all (overrides handler settings) / Не использовать графические устройства	[flag]	
enable-fast-resize	Enable fast two-stage frame resize / Включение двухэтапного изменения размера фрейма	[flag]	
handler-names HANDLER	Handlers to run / Количество обработчиков для запуска		if several are specified, they are merged into one handler pipe

Таблица 6 – Service options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
service-name name SERVICE_NAME	Name of service which will be used in transport protocol / Название службы используемой в транспортном протоколе		

Таблица 7 – Transport protocol options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
socketfile-uuid [SOCKETFILE_UUID]	Use UUID for socket names / Использование UUID для файлов сокета	[flag]	
image-encoding IMAGE_ENCODING	How CV Images are encoded in outcoming payloads / Как закодированы изображения в полезной нагрузке	[raw/jpeg [@<quality>]]	
max-batch-wait-msecs MAX_BATCH_WAIT_MSECS	Time which binary will wait for payloads, will send final_blow payload to handler when timer fired (used for standalone embedder) / Время ожидания полезной нагрузки для отправления final_blow обработчику	[int]	

Таблица 8 – Backend options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
backend	Transport backend / Бэкенд	{redis}	

rnet	Redis network type / Тип сетевого хранилища	{unix,tcp}	
raddr RADDR	Redis server addr or filename for unix / Адрес сервера или имя файла	[host:port]	
rdb RDB	Database to be selected after connecting to the redis / База данных выбираемая после подключения		
rpwd RPWD	Redis password / Пароль		
rannounce	Announce type for redis backend / Тип команды для бэкенда	{pubsub,blpop}	
rcreds RCREDS	Path to redis credentials file / Путь к файлу учетных данных		
withglobal [WITHGLOBAL]	Flag to get Global Data / Отметка для получения данных		
iql IQL	Input Queue Limit: will cut INPUT Queue, by deleting items, if size bigger than iql (0 – unlimited queue size) / Сокращение очереди ввода путём удаления элементов если размер превышает предел очереди	[int]	
oql OQL	Output Queue Limit: will cut OUTPUT Queue, by deleting	[int]	

	<p>items, if size bigger than oql (0 – unlimited queue size) /</p> <p>Сокращение очереди вывода путём удаления элементов если размер превышает предел очереди</p>		
--	---	--	--

Таблица 9 – Stream reader options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
reader-backend	Reader/decoder backend / Декодер бэкенда	{opencv}	
decoder-device DECODER_ARGS	stream-decoder / Стрим-декодер DECODER_DEVICE	[cpu:<id>/gpu:<id >] Decoder device	
decoder-args DECODER_ARGS	Additional args for ffmpeg decoder / Дополнительные аргументы для ffmpeg декодера		
input-args INPUT_ARGS ffmpeg-args INPUT_ARGS	Additional input args for ffmpeg / Дополнительные вводимые аргументы для ffmpeg		
start-second START_SECOND	Second to start reading at (video only) / Секунда с которой начинается считывание	[float]	
enable-archive [ENABLE_ARCHIVE]	Save stream data in filesystem / Сохранение потоковых данных в системе	[flag]	

archive-dir ARCHIVE_DIR	Base video archive directory / Основной каталог видеоархива		
archive-format ARCHIVE_FORMAT	Video archive video format / Формат видеоархива		
archive-segment-time ARCHIVE_SEGMENT_TIME	Segment duration in video archive (s) / Длительность части в видеоархиве	[int]	
archive-ffmpeg-args ARCHIVE_FFmpeg_ARGS	Additional output ffmpeg args for video archive / Дополнительные выводимые аргументы ffmpeg видеоархива		
probe-timeout PROBE_TIMEOUT	Timeout for stream properties probing (s) / Тайм-аут для исследования свойств потока	[int]	
read-timeout READ_TIMEOUT	Timeout for stream read operation (s), 0 = no timeout / Тайм-аут для чтения потока	[float]	
archive-ext ARCHIVE_EXT	Extension of video files in archive (if enabled) / Расширение видеофайлов в архиве		

Таблица 10 – CV tracking options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
enable-cvtracking [ENABLE_CVTRACKING]	Enable CV tracking of faces in addition to normal detection / Включение отслеживания лиц в	[flag]	

	дополнение к обычному обнаружения		
cvtracking-ttl-limit CVTRACKING_TTL_LIMIT	Number of frames for which CV tracker stores faces not found by detector / Количество рамок для которых трекер сохраняет лица не найденные детектором	[int]	
cvtracking-iou-threshold CVTRACKING_IOU_THRESHOLD	IoU threshold for matching detector bboxes to CV tracker bboxes / IoU порог для сравнения координат детектора с координатами трекера	[float 0..1]	

Таблица 11 – Player options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
player-fps PLAYER_FPS	Preview fps, 0 - as is / Просмотр кадров в секунду	[float]	
player-scale PLAYER_SCALE	Player scale; format is either SWxSH or just a number (in the latter case image is scaled proportionally) / Масштаб плеера; формат либо SWxSH, либо числовой (в этом случае изображение масштабируется пропорционально)		

player-mode	Player mode; available choices are imshow, rtmp and disable / Режим плеера; доступно – imshow, rtmp, отключение	{imshow, rtmp, disable}	
player-encoder-device PLAYER_ENCODER_DEVICE	Encoder device (for rtmp player) / Устройство кодирования (для плеера rtmp)	[cpu:<id>/ gpu:<id>]	
player-encoder-bitrate PLAYER_ENCODER_BITRATE	Encoded video bitrate (for rtmp player), format is the same as in ffmpeg / Битрейт закодированного видео (для плеера rtmp), формат тот же как в ffmpeg		
player-encoder-args PLAYER_ENCODER_ARGS	Additional output args for ffmpeg encoder (for rtmp player) / Дополнительные выходные аргументы для кодировщика ffmpeg (для проигрывателя rtmp)		
player-rtmp-path PLAYER_RTMP_PATH	Path to RTMP server / Путь к серверу RTMP		

Таблица 12 – Aligner options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
aligner-type ALIGNER_TYPE, --aligner ALIGNER_TYPE	Aligner type / Тип компонента предварительной обработки		

<p>aligner-config-path ALIGNER_CONFIG_PATH, --aconfig ALIGNER_CONFIG_PATH</p>	<p>Aligner config file / Конфиг-файл компонента предварительной обработки</p>		
<p>aligner-device ALIGNER_DEVICE</p>	<p>Aligner device / Устройство компонента предварительной обработки</p>	<p>[cpu:<id>/ gpu:<id>]</p>	
<p>aligner-return-type</p>	<p>If `list`, all faces will be wrapped in own payload, `batch` will put all faces and json data in cv_images and json arrays with single payload / Если "list", то все лица будут завернуты в собственные полезные нагрузки, "batch" поместит все лица и данные json в библиотеку cv_images и массивы json с одной полезной нагрузкой</p>	<p>{list,batch}</p>	
<p>aligner-inference-backend</p>	<p>Backend which will run models / Бэкенд, запускающий модели</p>	<p>{native, tvml}</p>	
<p>no-preview-padding [NO_PREVIEW_PADDING]</p>	<p>If true, faces will be returned without padding, in embedding-ready format / Если верно, то лица будут возвращены без заполнения в готовом для встраивания формате</p>	<p>[flag]</p>	
<p>max-wdr-delta MAX_WDR_DELTA</p>	<p>Faces with wd_ratio_delta bigger than this will not be forwarded from aligner (0 - disable check) / Лица с wd_ratio_delta больше,</p>	<p>[float]</p>	

	<p>чем это, не будут перенаправлены из компонента предварительной обработки</p>		
<p>max-pitch MAX_PITCH</p>	<p>Faces with pitch bigger than this will not be forwarded from aligner (0 - disable check) / Лица с наклоном, превышающим этот, не будут отправляться из компонента предварительной обработки</p>	<p>[float]</p>	
<p>max-yaw MAX_YAW</p>	<p>Faces with yaw bigger than this will not be forwarded from aligner (0 - disable check) / Лица с отклонением, превышающим этот, не будут отправляться из компонента предварительной обработки</p>	<p>[float]</p>	
<p>max-nose-x-delta MAX_NOSE_X_DELTA</p>	<p>Faces with nose_x_delta bigger than this will not be forwarded from aligner (0 - disable check) / Лица с размером nose_x_delta больше, чем этот, не будут отправляться из компонента предварительной обработки</p>	<p>[float]</p>	
<p>min-sharpness MIN_SHARPNESS</p>	<p>Faces with sharpness less than this will not be forwarded from aligner (0 - disable check) / Лица с резкостью меньшей, чем эта, не будут отправляться из компонента предварительной обработки</p>	<p>[float 0..1]</p>	

<p>disable-sharpness [DISABLE_SHARPNESS]</p>	<p>Disable advanced blur detection (sharpness will be always set to 1) / Отключение расширенного обнаружения размытия</p>	<p>[flag]</p>	
<p>disable-pose-solving [DISABLE_POSE_SOLVING]</p>	<p>Disable 3D pose solving (pitch and yaw will be always set to 0) / Отключение 3D позиции</p>	<p>[flag]</p>	
<p>forward-aligned-landmarks [FORWARD_ALIGNED_LANDMARKS]</p>	<p>Whether to forward the coordinates of aligned landmarks / Отправлять ли координаты выравненных объектов</p>	<p>[flag]</p>	
<p>forward-full-frame [FORWARD_FULL_FRAME]</p>	<p>Whether to forward the original frame from aligner alongside cropped faces / Отправлять ли исходный кадр из компонента предварительной обработки вместе с обрезанными лицами</p>	<p>[flag]</p>	
<p>forward-frame-encoding FORWARD_FRAME_ENCODING</p>	<p>How the original frame is encoded when forwarded from aligner (if --forward-full-frame is on) / Как кодируется исходный кадр при пересылке с компонента предварительной обработки</p>	<p>[raw/jpeg[@<quality>]]</p>	
<p>forward-frame-max-resolution FORWARD_FRAME_MAX_RESOLUTION</p>	<p>Max resolution of frame to be forwarded from aligner (if --forward-full-frame is on); frames exceeding this will be resized to roughly the same size (preserving original proportions); format is WxH or just N – a total number of</p>		

	<p>pixels, pass 0 to disable this check and always forward frame as-is /</p> <p>Максимальное разрешение кадра который будет отправляться из компонента предварительной обработки; кадры, превышающие это значения будут изменены примерно до того же размера (с сохранением исходных пропорций); формат WxH или N – общее количество пикселей</p>		
--	---	--	--

Таблица 13 – Embedder options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
embedder-type EMBEDDER_TYPE	Embedder type / Тип компонента извлечения биометрического шаблона		
embedder-config-path EMBEDDER_CONFIG_PATH, --econfig EMBEDDER_CONFIG_PATH	Embedder config file / Конфигурационный файл компонента извлечения биометрического шаблона		
embedder-model-name EMBEDDER_MODEL_NAME	Embedder model name / Наименование модели компонента извлечения биометрического шаблона		

<p>embedder-batch-size EMBEDDER_BATCH_SIZE</p>	<p>Embedder batch size (applicable only if embedder type is `mxnet`) / Размера пакета компонента извлечения биометрического шаблона</p>	<p>[int]</p>	
<p>embedder-device EMBEDDER_DEVICE</p>	<p>Embedder device (applicable only if embedder type is `mxnet`) / Устройство компонента извлечения биометрического шаблона</p>	<p>[cpu:<id>/ gpu:<id>]</p>	
<p>embedder-batch-ttl EMBEDDER_BATCH_TTL</p>	<p>Batch TTL, batch will be processed after that even if not full (0 - no TTL) / Время жизни пакета, пакет будет обработан после этого, даже если он не заполнен</p>	<p>[int]</p>	
<p>embedder-batch-mode</p>	<p>Batch mode, `wait` will send dead_end response till all slots for batch will be filled, `nowait` will process anyway, empty slots will be filled with zeros / Пакетный режим, "wait" отправит результат dead_end пока все слоты для пакета не будут заполнены, "nowait" обработает, а пустые слоты будут заполнены нулями</p>	<p>{wait, nowait}</p>	
<p>embedder-inference-backend</p>	<p>Backend which will run models / Бэкенд который запускает модели</p>	<p>{native, tvml}</p>	

embedder-enable-flip enable-flip		[flag]	
-------------------------------------	--	--------	--

Таблица 14 – Face analyzer options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
face-analyzer-type FACE_ANALYZER_TYPE	Face analyzer type / Тип анализатора лица		
face-analyzer-config-path FACE_ANALYZER_CONFIG_PATH	Face analyzer config file / Файл конфигурации анализатора лиц		
face-analyzer-model-name FACE_ANALYZER_MODEL_NAME	Face analyzer model name / Наименование модели анализатора лиц		
face-analyzer-batch-size FACE_ANALYZER_BATCH_SIZE	Face analyzer batch size / Размер пакета анализатора лиц	[int]	
face-analyzer-device FACE_ANALYZER_DEVICE	Face analyzer device / Устройство анализатора лиц	[cpu:<id>/gpu:<id>]	
face-analyzer-batch-ttl FACE_ANALYZER_BATCH_TTL	Batch TTL, batch will be processed after that even if not full (0 - no TTL) / Время жизни пакета, пакет будет обработан, даже если он не полный	[int]	

face-analyzer-batch-mode	Batch mode, `wait` will send dead_end response till all slots for batch will be filled, `nowait` will process anyway, empty slots will be filled with zeros / Пакетный режим, "wait" отправит результат dead_end пока все слоты для пакета не будут заполнены, "nowait" обработает, а пустые слоты будут заполнены нулями	{wait, nowait}	
face-analyzer-inference-backend	Backend which will run models / Бэкенд который запускает модели	{native, tvn}	

Таблица 15 – Options to be forwarded to tracker

Параметр	Описание	Ограничение по значению	Значение по умолчанию
threshold THRESHOLD	Threshold / Пороговое значение обнаружения лиц в видеопоток	[float]	
max-track-size MAX_TRACK_SIZE	Maximum length of track (frames) / Максимальная длина трека (кадров)	[int]	
track-reid-threshold TRACK_REID_THRESHOLD	Threshold to determine is the person is the same in tracker, above it will be combined in one track / Порог для определения соответствия человека в трекаре	[float]	

<p>track-sample-threshold TRACK_SAMPLE_THRESH OLD</p>	<p>Threshold for creating new sample of person inside track, all which above will create new sample / Порог срабатывания сравнения людей в треке</p>	<p>[float]</p>	
<p>min-track-size MIN_TRACK_SIZE</p>	<p>Minimal frames for complete track, does not affect tracks with suspects / Минимальное количество кадров для формирования трека</p>	<p>[int]</p>	
<p>track-ttl TRACK_TTL</p>	<p>How many frames tracker will wait for person before track completion / Максимальное количество кадров, по истечению которого при повторном появлении лица в обзоре данной камеры, на это лицо начнет формироваться следующий трек</p>	<p>[int]</p>	
<p>max-images-per-track MAX_IMAGES_PER_TRACK СК</p>	<p>How many images from track will be saved to disk / Сколько изображений из трека будет сохранено на диск</p>	<p>[int]</p>	
<p>preliminary-notifications</p>	<p>Which preliminary notifications will be sent / Какие предварительные уведомления будут отправлены</p>	<p>{suspects,all}</p>	
<p>watcher-mode</p>	<p>Liveness analyzer options / Функции Liveness анализатора</p>	<p>{black,white}</p>	

liveness-batch-size LIVENESS_BATCH_SIZE	Liveness analyzer batch size / Размер пакета Liveness анализатора	[int]	
liveness-device LIVENESS_DEVICE	Liveness analyzer device / Устройство Liveness анализатора	[cpu:<id>/ gpu:<id>]	
liveness-batch-ttl LIVENESS_BATCH_TTL	Batch TTL, batch will be processed after that even if not full (0 - no TTL) / Время жизни пакета, пакет будет обработан, даже если он не полный	[int]	
liveness-batch-mode	Batch mode, `wait` will send dead_end response till all slots for batch will be filled, `nowait` will process anyway, empty slots will be filled with zeros / Пакетный режим, "wait" отправит результат dead_end пока все слоты для пакета не будут заполнены, "nowait" обработает, а пустые слоты будут заполнены нулями	{wait,nowait}	
liveness-inference- backend	Backend which will run models / Бэкенд который запускает модли	{native,jit,tvm}	

Таблица 16 – Mask detector options

Параметр	Описание	Ограничение по значению	Значение по умолчанию
----------	----------	----------------------------	--------------------------

<p>mask-detector-batch-size MASK_DETECTOR_BATCH_SIZE</p>	<p>Mask detector batch size / Размер пакета компонента захвата</p>	<p>[int]</p>	
<p>mask-detector-device MASK_DETECTOR_DEVICE</p>	<p>Mask detector device / Устройство компонента захвата</p>	<p>[cpu:<id>/gpu:<id>]</p>	
<p>mask-detector-batch-ttl MASK_DETECTOR_BATCH_TTL</p>	<p>Batch TTL, batch will be processed after that even if not full (0 - no TTL) / Время жизни пакета, пакет будет обработан, даже если он не полный</p>	<p>[int]</p>	
<p>mask-detector-batch-mode</p>	<p>Batch mode, `wait` will send dead_end response till all slots for batch will be filled, `nowait` will process anyway, empty slots will be filled with zeros / Пакетный режим, "wait" отправит результат dead_end пока все слоты для пакета не будут заполнены, "nowait" обработает, а пустые слоты будут заполнены нулями</p>	<p>{wait,nowait}</p>	
<p>mask-detector-inference-backend</p>	<p>Backend which will run models / Бэкенд который запускает модели</p>	<p>{native,tvm}</p>	
<p>mask-threshold MASK_THRESHOLD</p>	<p>Threshold at which faces will be identified as masked / Порог при котором лица будут</p>	<p>[float 0..1]</p>	<p>0.5</p>

	идентифицироваться как замаскированные		
mask-for-overlaps MASK_FOR_OVERLAPS	What mask value to set for overlapping faces / Устанавливаемое значение перекрывающихся лиц	[0, 1]	

5. НАСТРОЙКИ

5.1. Настройки доступа

Доступ осуществляется на базе распределенной клиент-серверной архитектуры, как показано на рисунке 2.

Серверная часть системы включает в себя компьютеры с установленными на них серверными программными компонентами. Количество серверов зависит от требований к производительности системы, т. е. к величине обрабатываемого человекопотока и общему количеству фиксируемых лиц. При невысоких требованиях к производительности системы, при небольшом количестве каналов и малом объеме передаваемых данных, возможна установка серверных программных компонентов на одном компьютере.

Клиентская часть служит для отображения полученных данных и управления системой, это интерфейс оператора.

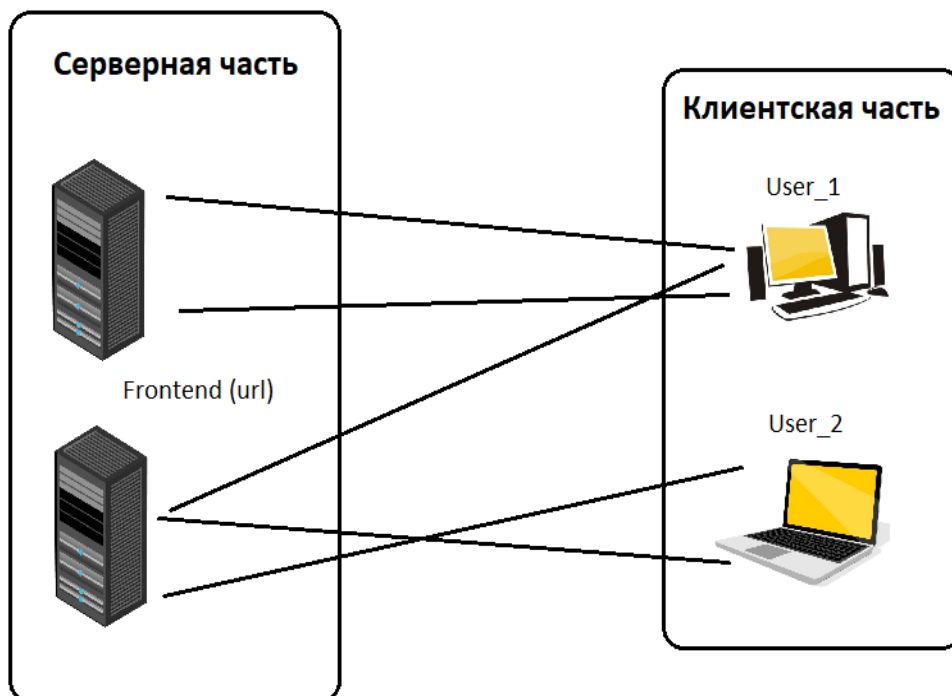


Рисунок 2 – Обобщенная структурная схема доступа односерверной конфигурации

Для того, чтобы настроить доступ с рабочего места пользователя (User_1, User_2 на рисунке 2) к сервисам программы, в браузере Chrome откройте ссылку:

<http://«IP сервера»:18080>

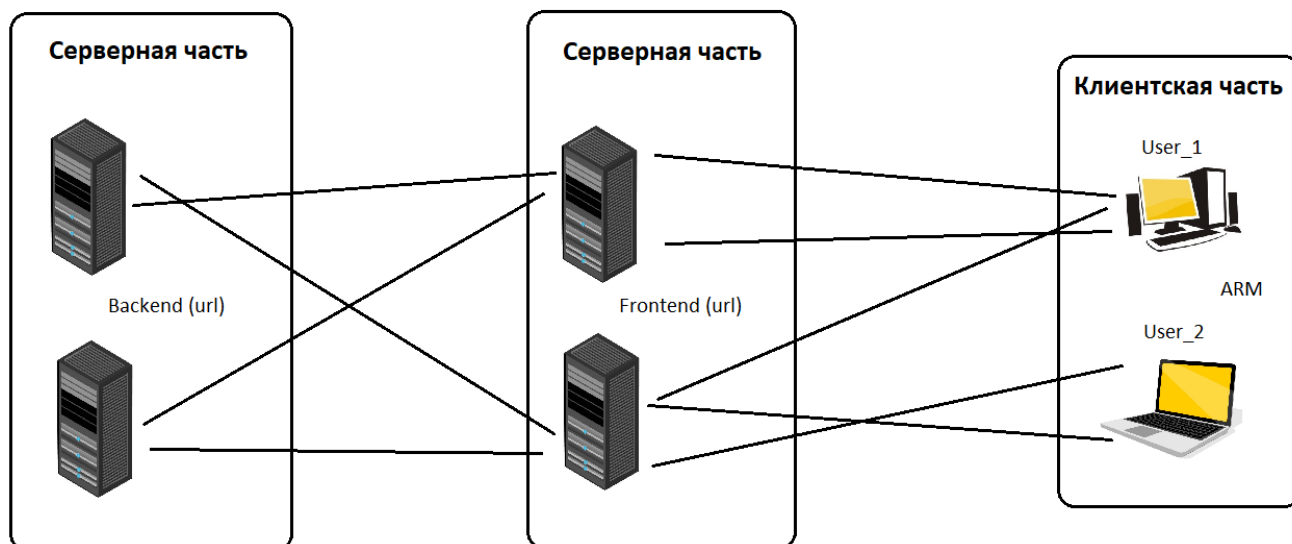


Рисунок 3 – Обобщенная структурная схема доступа многосерверной конфигурации

После этого необходимо ввести логин и пароль. Далее откроется интерфейс программы. Чтобы зайти в Настройки доступа нажмите кнопку с изображением шестеренки, которая расположена в верхнем левом углу основного меню Программы IVA CV. Откроется всплывающее окно Настройки. Вкладка Настройки представлена на рисунках 4, 5.

В данном окне вместо localhost прописываются пути до сервисов программного обеспечения. Если сервер один, то IP адрес будет прописываться один и тот же. В многосерверной конфигурации необходимо указать IP сервера с бэкендом (рисунок 3).

Вместо localhost будет указан IP сервера соответствующего сервиса.

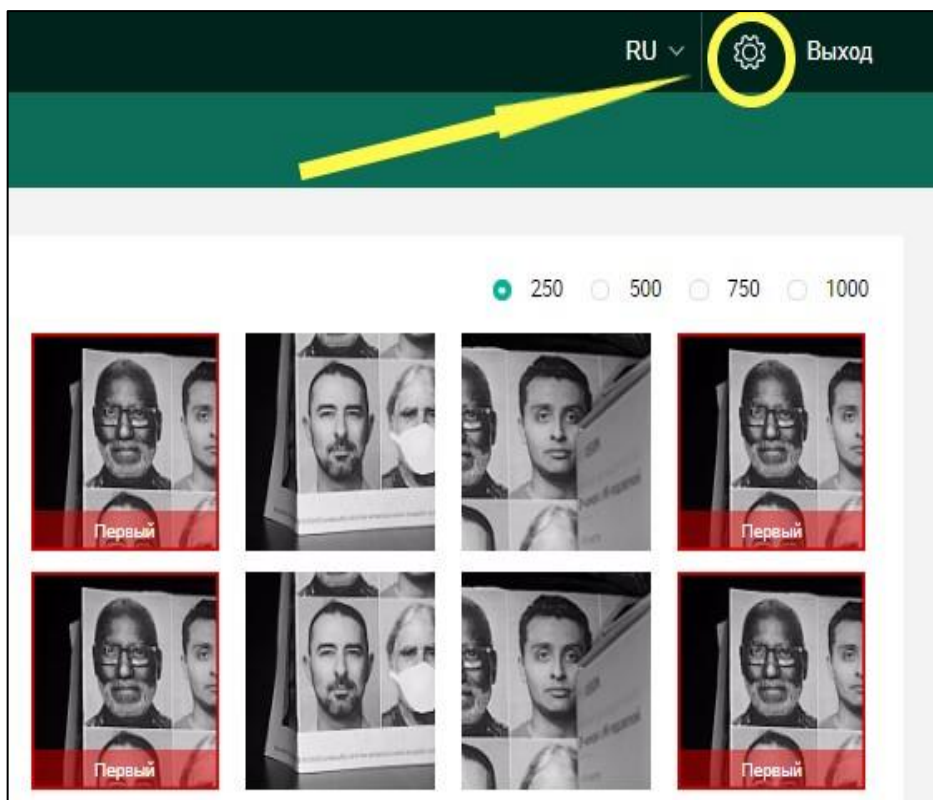


Рисунок 4 – Настройки доступа

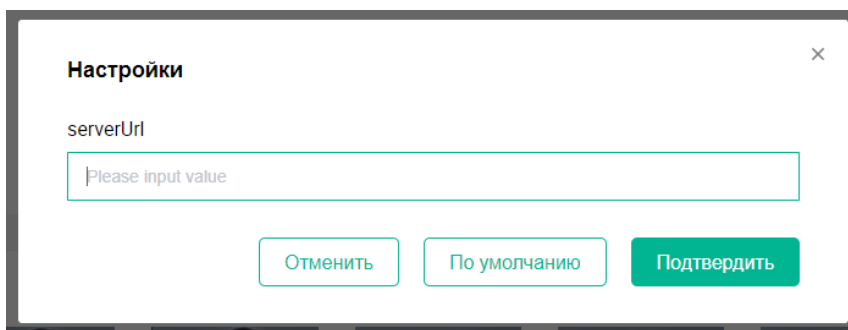


Рисунок 5 – Настройки доступа