

VisionLabs FaceStream

Примечания к выпуску

Содержание

FaceStream v.5.1.13	3
FaceStream v.5.1.12	3
FaceStream v.5.1.10	4
FaceStream v.5.1.9	5
FaceStream v.5.1.8	6
FaceStream v.5.1.7	7
FaceStream v.5.1.6	8
FaceStream v.5.1.5	8
FaceStream v.5.1.4	9
FaceStream v.5.1.3	10
FaceStream v.5.1.2	11
FaceStream v.5.1.1	11
FaceStream v.5.1.0	12
FaceStream v.5.0.7	15
FaceStream v.5.0.6	15

FaceStream v.5.1.13

- Добавлен механизм миграции настроек LUNA Streams, задаваемых в сервисе LUNA Configurator.

Обновление LUNA Streams

Ранее настройки LUNA Streams автоматически мигрировались при миграции сервиса LUNA Configurator. Теперь настройки LUNA Streams мигрируются с помощью отдельного механизма миграций. См. руководство по обновлению для более подробной информации.

Запуск LUNA Streams с нуля

Теперь настройки LUNA Streams загружаются в сервис Configurator с помощью механизма миграций. Настройки FaceStream по-прежнему загружаются в LUNA Configurator с помощью скрипта «load_dump.py». См. руководство по установке для более подробной информации.

Исправленные ошибки

- Исправлено падение FaceStream, возникавшее при задании некорректного адреса в параметре «lunastreams» > «origin» настроек FaceStream.

Теперь FaceStream корректно завершается, а в логи пишется соответствующая ошибка.

- Исправлена ошибка, из-за которой не проверялся допустимый диапазон (от 0 до 1024 пикс.) для параметра «size_source_frame» настроек FaceStream, что приводило к возможности задания любых значений для этого параметра.
- Исправлена ошибка, при которой FaceStream выдавал трассировку стека вместо ошибки если был занят порт на котором он запускался.

Теперь в логах FaceStream выдается ошибка `Error occurred on initializing application. Description: bind: Address already in use.`

- Исправлена ошибка, из-за которой изменение параметра «enable_gpu_processing» настроек FaceStream не применялось должным образом и приводило к падению при перезагрузке конфигурации.

FaceStream v.5.1.12

- FSDK обновлен до версии 5.11.0.

Исправленные ошибки

- Исправлена ошибка, из-за которой счетчик ошибок потока обрабатывался неправильно.

Если за время «period» количество ошибок не достигало значения «max_error_count», то счетчик ошибок не сбрасывался, а продолжал увеличиваться.

- Исправлена ошибка, из-за которой в событие LUNA PLATFORM записывались строки нулевой длины и значения «null» из параметров группы «location» настроек управления потоками.

FaceStream v.5.1.10

- Добавлена поддержка новой системы авторизации LUNA PLATFORM 5.

В [настройках создания потока](#) LUNA Streams доступно новое необязательное поле авторизации `event_handler > authorization`, в котором необходимо указать `token` или `account_id`. Параметр `event_handler > authorization > account_id` должен совпадать с параметром `account_id`, задаваемым при создании потока. Если новое поле авторизации не заполнено, то будет использован идентификатор `account_id`, задаваемый при создании потока.

При обновлении на новую версию LUNA Streams, автоматически выполнится миграция базы данных для поддержки нового поля.

Начиная с этой версии сборки FaceStream требуется использовать версию LUNA PLATFORM не менее v.5.32.0.

- Контейнеры сервисов LUNA Streams и FaceStream теперь именуются одинаково при ручном запуске и при запуске с помощью Docker Compose.
- В контейнерах сервисов LUNA Streams и LUNA Configurator обновлена версия Python до 3.10.

В документации обновлены все команды, связанные с использованием Python внутри контейнеров, а именно, команды вида «python3.9» заменены на «python3».

- В комплект поставки добавлены новые документы «FS_Docker_Upgrade_Manual_Rus.pdf» и «FS_Docker_Upgrade_Manual_Rus.html», описывающие процесс обновления с предыдущей сборки сервисов LUNA Streams и FaceStream, а также сервисов LUNA PLATFORM, запущенных в соответствии с документацией FaceStream.

Исправленные ошибки

- Исправлена ошибка, при которой некорректно работал параметр «jpeg_quality_level» на GPU.

При значении параметра `request_type = jpeg` сжимался только первый исходный кадр. Остальные исходные кадры не сжимались.

При значении параметра `request_type = json` исходные кадры не сжимались, но сжимались все биометрические образцы, кроме первого.

FaceStream v.5.1.9

Изменения FaceStream

- Изменен механизм расчета DROI.

Раньше DROI представлял координаты относительно ROI с учетом поворота изображения. Теперь DROI представляет координаты относительно исходного кадра.

- Отчёт FaceStream об обработке потока, регулярно отправляемый в LUNA Streams, расширен новыми полями «last_frame» и «live», содержащими URL-адреса изображения последнего кадра и потока, запущенного в режиме реального времени.

Эти URL-адреса можно получить с помощью определенных запросов к сервису LUNA Streams (см. ниже).

- Ускорена работа FaceStream и снижено потребление памяти на GPU за счет ограничения выполнения оценки пяти ключевых точек лица в тех случаях, когда такая оценка не требуется.
- В логи FaceStream добавлено отображение IP-адреса сервера, на котором запущен FaceStream.

Изменения сервиса LUNA Streams

- В LUNA Streams добавлена миграция, при которой все существующие потоки будут обновлены в соответствии с новым механизмом расчета DROI (см. изменения FaceStream выше).

При обновлении FaceStream необходимо обязательно обновить LUNA Streams.

- В запрос [«get health»](#) добавлен параметр «include_luna_services».

С помощью параметра «include_luna_services» можно включать и отключать проверку healthcheck для сервисов LUNA PLATFORM, от которых зависит сервис LUNA Streams. Если опция включена, то отправляются дополнительные запросы на ресурсы «/healthcheck» этих сервисов.

- Добавлено два новых ресурса - [«get last frame preview»](#) и [«get live preview»](#), позволяющих получить значения полей «last_frame» и «live» из отчета FaceStream об обработке потока (см. изменения FaceStream выше).

В запрос [«get streams logs»](#) добавлены новые параметры «preview» > «last_frame» и «preview» > «live».

- Автоматический перезапуск потока теперь имеет свои статусы (параметр «status»).

Ниже перечислены статусы автоматического перезапуска:

- «disabled» - автоматический перезапуск отключен пользователем (отключен параметр «restart»)

- «enabled» - автоматический перезапуск включен, но в данный момент не активен поскольку поток не находится в статусе «failure»
- «in_progress» - автоматический перезапуск в процессе
- «failed» - превышено допустимое количество попыток автоматического перезапуска и ни одна попытка не увенчалась успехом
- «denied» - автоматический перезапуск разрешен пользователем, но невозможен из-за критической ошибки, полученной в отчете FaceStream. На данный момент критическая ошибка имеет следующее содержание: Failed to authorize in Luna Platform. В будущих версиях список критических ошибок может пополняться.

Статус автоматического перезапуска можно получить с помощью запроса [«get stream»](#).

FaceStream v.5.1.8

- Скрипт для загрузки дампов FaceStream и LUNA Streams в сервис LUNA Configurator теперь можно запустить на версиях Python 2.x и Python 3.x.

Для запуска скрипта на версии Python 2.x необходимо использовать команду `python`. Для запуска скрипта на версии Python 3.x необходимо использовать команду `python3`.

Также скрипт для загрузки дампов добавлен в Docker Compose. Теперь для использования скрипта `start_facestream.sh` требуется только наличие запущенных контейнеров LUNA PLATFORM и активированной лицензии.

- Добавлена поддержка Liveness на GPU.

Исправленные ошибки FaceStream

- Исправлена ошибка, при которой в счетчике Streams total (общее количество запущенных потоков) в логах FaceStream учитывались потоки со статусом «failure».
- Исправлено падение FaceStream, возникавшее при задании несуществующего тега в поле `logging > tags` в настройках FaceStream.
- Исправлена ошибка, приводившая к падениям FaceStream на многопоточной среде.
- Исправлена ошибка, при которой поворот изображения, регулируемый настройкой «rotation», выполнялся по часовой стрелке на CPU и против часовой на GPU. Теперь на обоих типах процессоров поворот выполняется по часовой стрелке.
- Исправлена ошибка, при которой FaceStream брал в обработку большее количество потоков, чем указано в лицензионном ключе.
- Исправлено падение FaceStream при работе на GPU с использованием параметра «size_source_frame» из настроек FaceStream не равным нулю.

Изменения сервиса LUNA Streams

- Значительно ускорено выполнение запросов к ресурсам [«get stream»](#) и [«get streams»](#) при большом количестве записей в таблицах логов потоков базы данных LUNA Streams.
- Добавлен новый ресурс [«delete streams logs»](#), позволяющий очистить логи потоков старше указанной в запросе даты (параметр "log_time__lt"), исключая последнюю запись в логе.

FaceStream v.5.1.7

- Теперь URL для отправки исходных кадров лиц и тел указывается в параметре «frame_store». Ранее для отправки исходных кадров тел использовался URL из параметра «origin», а URL для отправки исходных кадров лиц задавался в параметре «frame_store».
- Теперь при запросах к сервисам LUNA PLATFORM, выполненных с ошибкой, в логах FaceStream будет отображаться значение «Luna-Request-Id», сгенерированное FaceStream.
По сгенерированному значению «Luna-Request-Id» можно найти ошибку в логах сервисов LUNA PLATFORM 5.
- В директорию «example-docker/luna_configurator/dumps/» комплекта поставки добавлен скрипт «load_dump.py», позволяющий загружать настройки FaceStream и LUNA Streams в LUNA Configurator.

Пример команды загрузки настроек для сервиса LUNA Streams: `python -m load_dump --dump-file=streams_dump.json --luna-config=http://127.0.0.1:5070/1`

Данный способ принят за основной в руководстве по установке FaceStream (см. главу «Загрузка настроек в LUNA Configurator»).

Исправленные ошибки FaceStream

- Исправлено падение FaceStream, возникавшее из-за неправильной обработки ошибок сервера при отправке исходного кадра.
- Исправлена ошибка, приводившая к значительному увеличению потребления оперативной памяти во время работы FaceStream.

Изменения сервиса LUNA Streams

- В поле «video_info» ответа на запрос на получение информации о потоке (запрос [«get stream»](#)) добавлены новые параметры «duration» и «progress».

Параметр «duration» отображает общую продолжительность видеофайла в сек, а параметр «progress» отображает текущий прогресс видеофайла от 0 до 1, где 0 - начало видеофайла, 1 - конец видеофайла.

FaceStream v.5.1.6

- Удален параметр «send_detection_path» из настроек FaceStream. Теперь возможность отправки детекций с координатами тела человека регулируется численным значением параметра «minimal_body_track_length_to_send». Если значение параметра равно «0», то отправка детекций выполняться не будет.
- Теперь при отправке исходного кадра в хранилище LUNA Image Store с помощью параметра «frame_store», также будет отправлен идентификатор аккаунта. В папке с исходным кадром в бакете будет сохранен файл <source_frame_id>.meta.json, содержащий параметр «account_id».

Исправленные ошибки сервиса LUNA Streams

- Исправлена ошибка, при которой после перевода потока в статус «pause», в логах потока (см. запрос «get streams logs») отображался статус «in_progress».
- Исправлена ошибка, которая позволяла устанавливать очень большие значения для параметров в запросах «create stream» и «Post feedback on streams».

FaceStream v.5.1.5

- Добавлена возможность получать время детекции лица или тела относительно начала видеофайла (в секундах).

Данное время сохраняется в поле «detect_ts» события LUNA PLATFORM.

Получение времени детекции лица или тела доступно только для типа источника сигнала «videofile».

- Обновлено версия LUNA Streams.

Исправленные ошибки FaceStream

- Исправлена ошибка, при которой поток не переходил в статус «failure» при использовании некорректных параметров.
- Исправлено получение метрики «detections_count» по запросу GET «/metrics».
- Из настроек FaceStream удалён неподдерживаемый с версии 5.0.6 параметр «show_window».
- Исправлена ошибка, возникающая на некоторых видеокартах при просмотре потока с помощью ресурса «/streams/preview/{id}» во время использования FaceStream с GPU.

Изменения сервиса LUNA Streams

- В запросы «create stream» и «put stream» сервиса LUNA Streams добавлен новый параметр «endless», позволяющий управлять перезагрузкой потока при получении ошибки сети (ошибка определяется системой как маркер конца файла eof).

Параметр доступен только для типов источника сигнала «udp» и «tcp».

Если параметр `endless` принимает значение `true`, то в случае получения `eof` и успешного переподключения, обработка потока будет продолжена. Если все попытки переподключения не удались (см. группу параметров «healthcheck»), то поток примет статус «failure». Если же параметр принимает значение `false`, то обработка потока не будет продолжена и поток примет статус «done».

При трансляции видеофайла предполагается использование значения `false`. Это позволяет избежать повторной обработки уже обработанного фрагмента видеофайла при получении `eof`. Если же при трансляции видеофайла значение параметра `endless` будет `true`, то после окончания обработки видеофайл начнет обрабатываться с начала.

Исправленные ошибки сервиса LUNA Streams

- Исправлен порядок получения логов потоков в запросе «get streams logs». Теперь новые логи отображаются первыми.

FaceStream v.5.1.4

- Добавлены два новых ключа запуска для FaceStream - «CONFIG_RELOAD» и «PULLING_TIME».

Ключ запуска «CONFIG_RELOAD» включает проверку наличия изменений в секции «FACE_STREAM_CONFIG» сервиса LUNA Configurator и принимает следующие значения:

- «1» - отслеживание изменений включено, при наличии изменений в конфигурации будут автоматически перезапущены все контейнеры FaceStream;
- «0» - отслеживание изменений отключено (значение по умолчанию).

Ключ запуска «PULLING_TIME» задает период получения новых параметров из секции «FACE_STREAM_CONFIG» сервиса LUNA Configurator в диапазоне [1...3600] сек. Используется совместно с тегом «CONFIG-RELOAD». По умолчанию значение равно «10».

Ключи запуска также доступны в режиме работы FaceStream с конфигурационными файлами.

- Добавлен новый параметр «min_body_size_threshold» в группу параметров «filtering» в настройке управления потоками.

Параметр задает размер детекции тела, меньше которого она не будет отправлена на обработку. Размер высчитывается как квадратный корень из произведения ширины (в пикселях) на высоту (в пикселях) детекции.

Пример: `min_body_size_threshold = sqrt (64*128)= 90.5`

Если значение равно «0», то фильтрация детекции тела по размеру выполняться не будет.

Исправленные ошибки

- Адрес исходного изображения теперь записывается в поле «detections» > «image_origin» события LUNA PLATFORM. Ранее этот адрес записывался в поле «face» > «user_data».
- Исправлена ошибка, при которой в запросе получения потоков из очереди (ресурс «/1/streams/processing/queue») не обрабатывался параметр «limit» (максимальное количество возвращаемых потоков) при фильтрации по группе.

FaceStream v.5.1.3

- SDK обновлен до версии 5.6.0.
- Добавлена возможность группировки потоков. Группировка призвана объединять потоки с множеством камер в логические группы. Например, можно группировать потоки по территориальному признаку.

Любой поток может принадлежать к нескольким группам, а также ни к одной.

Группа создается с помощью запроса «[create group](#)». Для создания группы нужно указать обязательные параметры «account_id» и «group_name». При необходимости можно указать описание группы.

Поток может быть привязан к группе двумя способами:

- с помощью параметров «group_name» или «group_id» во время создания потока (запрос «[create stream](#)»).
- с помощью запроса «[linker](#)». В запросе необходимо указать идентификаторы потоков и группу, к которой их необходимо привязать. С помощью данного запроса можно также отвязать потоки от группы.

Если поток был привязан к группе, то при запросе «[get stream](#)» или «[get streams](#)» группа будет отражена в поле «groups».

- Добавлена возможность передачи временных меток локального трека в создаваемые LUNA PLATFORM 5 события.
 - Для события, созданного по статическому обработчику, теперь записываются параметры «create_time» (время создания трека) и «end_time» (время окончания трека).
 - Для события, созданного по динамическому обработчику, теперь записывается параметр «end_time» (время окончания трека).
 - Для лучших снимков (биометрических образцов) теперь записывается параметр «detect_time» (время обнаружения).
- Теперь при включённом параметре «send_detection_path» вместе с лучшим снимком по умолчанию будет отправляться соответствующая детекция с координатами тела. Ранее такая детекция могла не отправляться.

FaceStream v.5.1.2

- Добавлен новый параметр «preferred_program_stream_frame_width» в настройки управления потоками, предназначенный для работы с протоколами, в которых подразумевается наличие нескольких каналов с разными битрейтами и разрешением (например, HLS).

Если у потока существует несколько таких каналов, то данный параметр позволит выбрать из всех каналов цельного потока тот канал, ширина кадра которого ближе к значению, указанному в данном параметре.

Например, имеется 4 канала, ширины кадров которых равны 100, 500, 1000 и 1400. Если параметр «preferred_program_stream_frame_width» равен «800», то будет выбран канал, ширина кадра которого равняется 1000.

Если у потока всего один канал, то данный параметр будет игнорироваться.

Данный параметр используется только для источников типа «tsr» или «udr».

- В секцию «lunastreams» настроек FaceStream добавлены следующие новые параметры:
 - request_stream_period - параметр задаёт временной промежуток между запросами на получение новых потоков из LUNA Streams в диапазоне от 0.1 до 3600 секунд.
 - send_feedback_period - параметр задаёт временной промежуток между отправкой отчётов об обработанных потоках в LUNA Streams в диапазоне от 1.0 до 3600 секунд. Значение данного параметра не должно превышать значение параметра «STREAM_STATUS_OBSOLETING_PERIOD», задаваемого в настройках сервиса LUNA Streams.
 - max_feedback_delay - параметр задаёт максимальную задержку отправки отчёта в диапазоне от 1.0 до 3600 секунд. Если отчёт не был отправлен за данное время, то FaceStream остановит обработку текущих потоков. Значение данного параметра не должно быть меньше, чем значение параметра «send_feedback_period» и не должно превышать значение параметра «STREAM_STATUS_OBSOLETING_PERIOD», задаваемого в настройках сервиса LUNA Streams.

FaceStream v.5.1.1

- Исправлена ошибка в файле docker-compose.yml, приводившая к ошибке «host network_mode is incompatible with port_bindings» во время запуска скрипта Docker Compose.

FaceStream v.5.1.0

- Изменён принцип создания потоков. Ранее настройки для потоков задавались в конфигурационном файле «input.json» или в секции Configurator «FS_CONFIG». Теперь настройки для потоков задаются в теле формата JSON и отправляются с помощью HTTP запроса к сервису LUNA Streams (см. запрос [«create stream»](#) в документе «StreamsReferenceManual.html» комплекта поставки).

После отправки HTTP запроса с заданными параметрами к сервису LUNA Streams, параметры добавляются в базу данных LUNA Streams под уникальным идентификатором «stream_id». В зависимости от количества доступных потоков в лицензии (данная информация поступает из сервиса LUNA Licenses), параметры попадают во внутреннюю очередь, где находятся в статусе «pending» пока FaceStream не заберёт их из очереди для последующей обработки (см. подробное описание сервиса в разделе «Взаимодействие FaceStream с LUNA Streams» в руководстве администратора).

Изменения лицензирования:

Сервис может обрабатывать определённое количество потоков одновременно. Количество таких потоков задаётся в параметре лицензионного ключа LUNA PLATFORM 5, определяющим количество потоков для работы LUNA Streams. Таким образом, начиная с версии 5.1.0, FaceStream лицензируется с помощью ключа LUNA PLATFORM 5.

Для запуска LUNA Streams и FaceStream необходимо как минимум три дополнительных компонента LUNA PLATFORM:

- Сервис LUNA Licenses. С помощью данного сервиса выполняется лицензирование сервиса LUNA Streams.
- Сервис LUNA Configurator. Настройки LUNA Streams хранятся в данном сервисе аналогично настройкам «FACE_STREAM_CONFIG» и «TRACK_ENGINE_CONFIG» для FaceStream. При отсутствии необходимости использования LUNA Configurator, можно запустить LUNA Streams с использованием конфигурационных файлов, однако данные действия не описаны в руководстве по установке.
- Контейнер PostgreSQL. В данном контейнере создаётся база данных LUNA Streams, в которую сохраняются параметры потоков, заданные ранее в теле JSON (см. описание базы данных в разделе «Описание базы данных LUNA Streams» руководства администратора). При необходимости можно использовать базу данных Oracle. Её использование не описано в документации.

Также по умолчанию запускается контейнер InfluxDB OSS 2. С помощью БД Influx осуществляется мониторинг сервисов LUNA PLATFORM. При необходимости мониторинг можно отключить.

Если компоненты LUNA PLATFORM не запущены и не активирована лицензия LUNA PLATFORM, то перед запуском FaceStream нужно выполнить следующие действия:

- Активировать лицензию.
- Запустить требуемые компоненты.
- Загрузить настройки LUNA Streams и FaceStream в LUNA Configurator.
- Создать и инициализировать базу данных LUNA Streams.

Если компоненты LUNA PLATFORM запущены и активирована лицензия LUNA PLATFORM, то перед запуском нужно выполнить следующие действия:

- Загрузить настройки LUNA Streams и FaceStream в LUNA Configurator.
- Убедиться, что в лицензии указан параметр, определяющий количество потоков для обработки сервисом LUNA Streams и указать адрес сервера с запущенным сервисом LUNA Licenses в сервисе LUNA Configurator (если сервис LUNA Licenses не находится на сервере с LUNA Streams).
- Создать и инициализировать базу данных LUNA Streams.

Описание выполнения данных действий приведено в руководстве по установке FaceStream.

- Добавлена возможность развернуть FaceStream с помощью скрипта Docker Compose. Скрипт позволяет автоматически запустить и настроить LUNA Streams и FaceStream. Перед запуском скрипта должны быть выполнены следующие действия:
 - Активирована лицензия.
 - Запущены требуемые компоненты.
 - Настройки LUNA Streams и FaceStream загружены в Configurator.
 - Создана и инициализирована база данных LUNA Streams.
- В документацию добавлено описание ошибок, возвращаемых сервисами LUNA Streams и LUNA Licenses, а также общих ошибок, которые могут возникнуть при взаимодействии с LUNA PLATFORM.
- В настройки FaceStream (секция «FACE_STREAM_CONFIG» сервиса LUNA Configurator) добавлена секция «lunastreams», позволяющая указать адрес используемого сервиса LUNA Streams (параметр «origin»), его версию API (параметр «api_version») и верхнюю границу количества потоков FS (параметр «max_number_streams»). Если сервис LUNA Streams запущен на другом сервере, то необходимо перед запуском FaceStream указать актуальный адрес сервиса. См. подробную информацию в руководстве администратора FaceStream.
- Названия и структура настроек управления потоками были переработаны. Все настройки теперь разбиты по логическим блокам. Были переименованы следующие настройки:
 - input, luna_account_id > account_id
 - input, transport > data, type
 - input, url > data, reference
 - output, image_store_url > event_handler, frame_store

Переработан способ передачи данных для генерации события. Ранее для генерации

события LUNA PLATFORM использовался параметр «output» > «url» в котором при работе с лицами необходимо было указать адрес к ресурсу «/events» в формате `http://<luna_address>:<port>/6/handlers/<handler_id>/events`, где `<handler_id>` - идентификатор обработчика. При работе с телами использовались отдельные параметры «luna_human_handler_id» и «luna_dynamic_human_handler_id», а поле «url» заполнялось в формате `http://<luna_address>:<port>/6/`. Теперь идентификатор обработчика задаётся в группе «event_handler» > «bestshot_handler»/«detection_handler» > «handler_id», где «bestshot_handler» - статический обработчик для лица или тела, «detection_handler» динамический «handler_id» LUNA PLATFORM для работы с телами (бывш. «luna_dynamic_human_handler_id»). Адрес сервиса LUNA API и версия API теперь задаются отдельно. Ниже приведен пример передачи данных для генерации события, когда заданы статический и динамический обработчики для тела.

```
"event_handler": {
  "origin": "http://127.0.0.1:5000",
  "api_version": 6,
  "bestshot_handler": {
    "handler_id": "ee4c42b6-23ae-410e-a2aa-a4220e64ba4b"
  },
  "detection_handler": {
    "handler_id": "426542d6-5509-4e5b-8a01-e2abd5c0a8c6"
  }
}
```

Если необходимо обработать лицо, то должен быть задан «handler_id» с параметрами для обработки лица в группе «bestshot_handler», а группа «detection_handler» не обязательна к заполнению.

См. подробную информацию настройках управления потоками в руководстве администратора FaceStream.

- Прекращена поддержка обычного режима запуска FaceStream. Теперь запустить FaceStream можно только в серверном режиме.
- Информация по запуску FaceStream в серверном режиме с конфигурационными файлами перенесена в руководство администратора (см. раздел «Использование FaceStream с конфигурационными файлами»).
- Изменены параметры логирования по умолчанию в настройках FaceStream, секции «Logging»:
 - Параметр «severity» теперь равен «1». Это означает, что в логи будут выводиться только предупреждения системы.

- Параметр «mode» теперь равен «l2c». Это означает, что логи будут выводиться только в консоль.

FaceStream v.5.0.7

- SDK обновлён до версии 5.5.0.
- Добавлена возможность получить текущую версию FaceStream с помощью запроса GET «/version» (см. подробную информацию в документе «FaceStreamApi.html»).
- Исправлена ошибка, при которой при запуске FS с Configurator не использовался параметр «save-debug-info» из настроек TrackEngine.
- Исправлена ошибка, при которой не отправлялся исходный кадр без детекций с координатами человека (detection_path) с установленным параметром «send_source_frame».
- Исправлена ошибка, при которой не работала отправка «track_id» для лиц, из-за чего не записывалось значение «track_id» в событие LUNA PLATFORM.
- Исправлена ошибка при использовании эстиматора рта. Ошибка могла возникать при использовании двух потоков, когда первый использовал данный эстиматор, а второй нет.

FaceStream v.5.0.6

- FaceStream теперь поставляется только в контейнере Docker. Для запуска FaceStream внутри контейнера используется CentOS 8.
- Обновлён комплект поставки FaceStream. Теперь он включает документацию и конфигурационные файлы, необходимые для запуска FaceStream.
- Добавлен документ «LP_Docker_Installation_Manual_Rus», описывающий запуск FaceStream из контейнера Docker. Данная документация перенесена из руководства администратора. Документ поставляется в форматах PDF и HTML.
- SDK обновлён до версии 5.4.1.
- Добавлена поддержка работы с телами. FaceStream позволяет выполнять поиск и сопровождение тел в потоке, пока они не покинут кадр или не будут перекрыты.

Для использования FaceStream с телами используются дополнительные настройки. Начиная с FaceStream 5.0.6 все настройки разбиты в документации на общие и индивидуальные для лиц и тел.

Для включения отслеживания тел необходимо изменить параметр TrackEngine «use-face-detector» на «use-body-detector», а также задать индивидуальные настройки источников потоков для тел (см. соответствующие главы).

Логика создания событий (отправка лучших снимков и их обработка в LUNA PLATFORM) для тел отличается от логики создания события для лиц (см. руководство администратора FaceStream для получения подробной информации).

- Все названия параметров в конфигурационных файлах приведены к единому виду. Например, ранее использовались названия «frame-processing-mode» и «real_time_mode_fps». Теперь все названия выглядят единым образом: «frame_processing_mode» и «real_time_mode_fps».
- Дополнены сообщения об ошибках. Теперь они содержат больше информации и путь к неправильному параметру.
- Больше не используется синхронное ожидание ответа от LUNA PLATFORM (также влияет на производительность при работе с телами).
- С лучшим снимком теперь будет отправляться время сервера. Ранее отправлялось время устройства. Данное нововведение распространяется на все виды источников (потoki, видеофайлы и наборы изображений).
- Прекращена поддержка 4 и 5 версии API LUNA PLATFORM. Начиная с этой версии FaceStream не может отправлять данные в LUNA PLATFORM 3, LUNA PLATFORM 4, Backport 3 и Backport 4. Отправка изображений возможна только в LUNA PLATFORM 5 (6 версия API).
- Использование параметра «show_window» и связанных с ним параметров более невозможно.