Smilart API Reference

Table of Contents

Overview

Smilart API is the set of several API services, which provide necessary functions to use Smilart recognition algorithms when you create your own app.

List of services, which are included into Smilart API:

1. Person service — person management service, which implements basic operations with persons, such as adding, getting, removing and updating person in Platform database.
2. Camera service — camera management service, which implements basic operations with added camera in Platform.
3. Video Content Analytics service — the service provides access to some events which are generated in Platform, such as receiving frames from camera, face detection results, identification result.
4. Photo Booth service — the service selects the best photos from the camera stream, in order to add them to Platform database to get the best identification results.
5. Verification service — the service implements verification case.
6. Instant Photo Analytics service — the service for instant photo analysis.
7. Adaptive Verification service — the service for adaptive verification.

Platform gets images from connected cameras. Each camera must send stream in one of the supported protocol. Camera can be either physical device or software, which can send data in supported format.

All services share common base of persons that used for face recognition.

Common protocol description

Smilart API uses messages exchange to interact with its clients.

Specification of the transport layer for message exchange is not a part of that specification and SHOULD be defined by concrete implementation of the Smilart API.

Each message is message described in *protocol buffer language (protobuf)* of version 2.6.

Protobuf Documentation

Adaptive Verification API

Provides opportunities to manage Adaptive Verification (AV) service.

The service provides the way to improve the user experience in verification process by **populating person base** by sampled photos during their successful verification from cameras and **adjustment of verification thresholds**.

Side effects on other services

Impact on Person Management service

Being activated, AV service can modify list of person photos (add and delete photos), but it **can delete only those photos which were added by this service (sampled photos). This service will not delete any person's photos, added by another service (e.g. Person Management service)**.

All sampled photos will be accessible in Person Management service with special flag indicating whether the photo was added (sampled) by AV service or not.

Being deactivated this service does not delete sampled photos. Client can get rid of sampled photos via explicit remove requests at any moment.

Impact on Verification service

Being activated AV service will change thresholds for verification requests and provide additional person photos for verification.

Impact on other services

Other services will not take into account sampled photos and will not change their behavior because of the activity of AV service.

Protocol

Adaptive Verification service uses Protobuf messages as format for interchange (look for `AdaptiveVerification.proto` file).

Service name: `api-av-service`.

Version: 1.0.

Content type of messages: `application/vnd.com.smilart/api.av.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
get_config	GetConfig.Request	optional	Request to get the service configuration.
set_config	SetConfig.Request	optional	Request to install the service configuration.
remove_all_photos	RemoveAllPhotos.Request	optional	Request to remove all sampled photos.
remove_photos_by_person	RemovePhotosByPerson.Request	optional	Request to remove all sampled photos of the person.
remove_photos_by_camera	RemovePhotosByCamera.Request	optional	Request to remove all sampled photos from the camera for every person.

ResponseEnvelope

Defines envelope response messages from service.

Field	Type	Label	Description
malformed_protobuf	MalformedProtobuf	optional	Protobuf in request was considered malformed.
get_config	GetConfig.Response	optional	Response to `GetConfig` message.
set_config	SetConfig.Response	optional	Response to `SetConfig` message.
remove_all_photos	RemoveAllPhotos.Response	optional	Response to `RemoveAllPhotos` message.
remove_photos_by_person	RemovePhotosByPerson.Response	optional	Response to `RemovePhotosByPerson` message.
remove_photos_by_camera	RemovePhotosByCamera.Response	optional	Response to `RemovePhotosByCamera` message.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description

message	string	required	Readable details of that decision.

GetConfig

Defines messages of the retrieving service configuration.

Field	Type	Label	Description

GetConfig.Request

Field	Type	Label	Description

GetConfig.Response

Response to the request to get the service configuration.

Field	Type	Label	Description
config_provided	GetConfig.Response.ConfigProvided	optional	Service configuration was provided.

GetConfig.Response.ConfigProvided

Service configuration was provided.

Field	Type	Label	Description
config	Config	required	Service configuration.

SetConfig

Defines messages for installation a new service configuration.

Field	Type	Label	Description

SetConfig.Request

Field	Type	Label	Description
config	Config	required	

SetConfig.Response

Response to request to install a new service configuration.

Field	Type	Label	Description
config_installed	SetConfig.Response.ConfigInstalled	optional	Config was installed.

SetConfig.Response.ConfigInstalled

New service configuration was installed.

Field	Type	Label	Description
config	Config	required	Actual configuration of the service.

RemoveAllPhotos

Defines messages for remove all sampled photos.

Field	Type	Label	Description

RemoveAllPhotos.Request

Field	Type	Label	Description

RemoveAllPhotos.Response

Response to request to remove all sampled photos.

Field	Type	Label	Description
photos_removed	RemoveAllPhotos.Response.PhotosRemoved	optional	Sampled photos were removed or did not exist.

RemoveAllPhotos.Response.PhotosRemoved

Sampled photos were removed or did not exist.

Field	Type	Label	Description

RemovePhotosByPerson

Defines messages for remove all sampled photos.

Field	Type	Label	Description

RemovePhotosByPerson.Request

Field	Type	Label	Description
person_id	string	required	Identifier of the person which sampled photos should be removed.

RemovePhotosByPerson.Response

Response to request to remove all sampled photos of the specified person.

Field	Type	Label	Description
photos_removed	RemovePhotosByPerson.Response.PhotosRemoved	optional	Sampled photos were removed or did not exist.

RemovePhotosByPerson.Response.PhotosRemoved

Sampled photos of the person were removed or did not exist.

Field	Type	Label	Description

RemovePhotosByCamera

Defines messages for remove all sampled photos from the specified camera for every person.

Field	Type	Label	Description

RemovePhotosByCamera.Request

Field	Type	Label	Description
camera_pid	string	required	

RemovePhotosByCamera.Response

Response to request to remove all sampled photos from the specified camera for every person.

Field	Type	Label	Description
photos_removed	RemovePhotosByCamera.Response.PhotosRemoved	optional	Sampled photos were removed or did not exist.

RemovePhotosByCamera.Response.PhotosRemoved

Sampled photos from camera for every person were removed or did not exist.

Field	Type	Label	Description

Config

Service configuration description.

Field	Type	Label	Description

active	bool	required	Desired service activity status.True if service should sample person photos and adapt verification thresholds, otherwise false.

Camera Management API

Provides opportunities to work with cameras.

Protocol

Camera Management service uses Protobuf messages as format for interchange (look for `CameraManagement.proto` file).

Service name: `api-cameras-service`.

Version: 1.1.

Content type of messages: `application/vnd.com.smilart/api.camera.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
list_cameras	ListCameras.Request	optional	Request to list cameras.
get_camera_mjpeg_stream_url	GetCameraMjpegStreamUrl.Request	optional	Request to get camera MJPEG stream url.

ResponseEnvelope

Defines envelope response messages from service.

Field	Type	Label	Description
list_cameras	ListCameras.Response	optional	Response to `ListCameras` message.
malformed_protobuf	MalformedProtobuf	optional	Protobuf in request was considered malformed.
get_camera_mjpeg_stream_url	GetCameraMjpegStreamUrl.Response	optional	Response to `GetCameraMjpegStreamUrl` message.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description
message	string	required	Readable details of that decision.

Camera

Represents camera's instance in the system.

Field	Type	Label	Description
pid	string	required	PID (persistent identifier) of the camera.
type	string	optional	Information about camera's type.
status	Camera.Status	optional	Extended information about camera.

Camera.Status

Status of the specified camera.

Field	Type	Label	Description
active	bool	optional	Was camera activated (started), typically intentionally by system administrator, in the system.
running	bool	optional	True if camera frames are available for processing, otherwise false.

ListCameras

Defines messages of the retrieving available cameras protocol in the system.

Field	Type	Label	Description

ListCameras.Request

Field	Type	Label	Description

ListCameras.Response

Response to list of available cameras request.

Field	Type	Label	Description
cameras_provided	ListCameras.Response.CamerasProvided	optional	Cameras was provided.

ListCameras.Response.CamerasProvided

Information about cameras was provided.

Field	Type	Label	Description
cameras	Camera	repeated	List of available cameras.

GetCameraMjpegStreamUrl

Defines messages of the camera MJPEG stream URL retrieving protocol.

Field	Type	Label	Description

GetCameraMjpegStreamUrl.Request

Request for get camera stream URL.

Field	Type	Label	Description
pid	string	required	Identifier of a camera.
max_fps	uint32	optional	Limit FPS stream from camera (0 means no limit).
resolution	GetCameraMjpegStreamUrl.Resolution	optional	Size of frames from camera.Picture from the camera will be scaled to fit built-in dimensions.It converts so that the proportions do not change and do not become larger.

GetCameraMjpegStreamUrl.Response

Response to get camera stream URL request.

Field	Type	Label	Description
stream_url	string	optional	Camera stream URL received.

camera_not_found	GetCameraMjpegStreamUrl.Response.CameraNotFound	optional	Camera was not found.
camera_not_active	GetCameraMjpegStreamUrl.Response.CameraNotActive	optional	Camera was not active.

GetCameraMjpegStreamUrl.Response.CameraNotFound

URL could not be provided due to unavailability of the specified camera.

Field	Type	Label	Description

GetCameraMjpegStreamUrl.Response.CameraNotActive

URL could not be provided due to inactivity of the specified camera.

Field	Type	Label	Description

GetCameraMjpegStreamUrl.Resolution

Defines resolutions of the frames.

Name	Number	Description
ORIGINAL	1	Original frame size (provided by the camera).
SMALL	2	Small frame size (Provided by the server, default is 320×240).
MEDIUM	3	Medium frame size (Provided by the server, default is 800×600).
LARGE	4	Large frame size (Provided by the server, default is 1400×1050).

Instant Photo Analytics API

Provides capability to analyze an image and generate analysis report.

Protocol

Instant Photo Analytics (IPA) service uses Protobuf messages as format for interchange (look for ipa.proto file).

Service name: `api-ipa-service`.

Version: 1.0.

Content type of messages: `application/vnd.com.smilart/api.ipa.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
analyze_image	AnalyzeImage.Request	optional	Request to `AnalyzeImage`.

ResponseEnvelope

Defines envelope response messages from the service.

Field	Type	Label	Description
analyze_image	AnalyzeImage.Response	optional	Response to `AnalyzeImage` message.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description
message	string	required	Readable details of that decision.

DetectOptions

Options of face detects.

Field	Type	Label	Description
detect_report_options	DetectOptions.DetectReportOptions	optional	If isn't set, face detection report will be excluded from analysis report.
max_faces	uint32	optional	Maximal number of faces to search for (default is 1000000).

DetectOptions.DetectReportOptions

Face detect report options.

Field	Type	Label	Description

add_face_cut	bool	optional	Being set true adds face cutting into detect report.

CorrelationOptions

Face correlation options with person base.

Correlations are primarily **debugging information** that can be completely correct interpreted only by the vendor's specialists and reflects the features of the **currently used** face recognition algorithm that **may change** in the future. Therefore, you **SHOULD NOT** make any conclusions based on the received coefficients, except for getting the top of the most similar persons in the database according to the **current** face recognition algorithm.

Field	Type	Label	Description
correlation_report_options	CorrelationOptions.CorrelationReportOptions	optional	If isn't set, correlation results will be excluded from analysis report.

CorrelationOptions.CorrelationReportOptions

Correlation report options.

Field	Type	Label	Description
max_persons	uint32	optional	Maximal number of persons to be added to correlation report (default is 1).

IdentificationOptions

Person identification options.

Field	Type	Label	Description
identification_report_options	IdentificationOptions.IdentificationReportOptions	optional	If isn't set, identification report will be excluded from analysis report.

IdentificationOptions.IdentificationReportOptions

Identification report options.

Field	Type	Label	Description

AnalyzeImage

Defines messages of the image analysis protocol.

Field	Type	Label	Description

AnalyzeImage.Request

Request for image analysis: find faces on image, correlate found faces with base, list identified persons on image.

Field	Type	Label	Description
image	Image	required	Image for analysis.
detect_options	DetectOptions	optional	Defines face detection options.If isn't set, face detection will use default values from `DetectOptions`.
correlation_options	CorrelationOptions	optional	Defines correlation options.If isn't set, will use default values from `CorrelationOptions` or can be completely omitted if `identification_options` was not set.
identification_options	IdentificationOptions	optional	Defines identification options.If isn't set, identification step will be omitted.

AnalyzeImage.Response

Response to `AnalyzePhoto` request.

Field	Type	Label	Description
image_analyzed	AnalyzeImage.Response.ImageAnalyzed	optional	Image was analyzed.
unsupported_image_type	AnalyzeImage.Response.UnsupportedImageType	optional	Image was not analyzed due to the presence of images with unsupported type.
payload_is_too_large	AnalyzeImage.Response.PayloadIsTooLarge	optional	Image was not analyzed due to excess of payload limits.

unsupported_input_source	AnalyzeImage.Response.UnsupportedInputSource	optional	Image was not analyzed due to the presence of images with unsupported source.
request_timeout	AnalyzeImage.Response.RequestTimeout	optional	Image was not analyzed due to exceeding execution time limit.
too_many_requests	AnalyzeImage.Response.TooManyRequests	optional	Image was not analyzed by the service due to overload by incoming requests.

AnalyzeImage.Response.ImageAnalyzed

Image was analyzed.

Field	Type	Label	Description
faces	DetectedFace	repeated	List of detected faces sorted by face size in descending order.

AnalyzeImage.Response.UnsupportedImageType

Service implementation can't process images due to the presence of images with unsupported by the service implementation format of images.

Field	Type	Label	Description

AnalyzeImage.Response.UnsupportedInputSource

Service implementation can't process images due to the presence of images with unsupported by the service implementation type of source.

Field	Type	Label	Description
type	Source.Type	repeated	Type of source that can't be processed.

AnalyzeImage.Response.PayloadIsTooLarge

Payload of images is too large for processing.

Field	Type	Label	Description
actual_size	int32	required	Received payload in bytes.

maximum_size	int32	required	Maximum acceptable size of payload in bytes by the service implementation.

AnalyzeImage.Response.RequestTimeout

Request processing exceeds execution time limit.

Field	Type	Label	Description

AnalyzeImage.Response.TooManyRequests

Service received too many requests for processing.

Field	Type	Label	Description

DetectedFace

Detected face on image.

Field	Type	Label	Description
detect	Detect	optional	Face detection report.
correlation	Correlation	optional	Face correlation report.
identification	Identification	optional	Person identification report.

Source

Source of some payload.

Field	Type	Label	Description
binary	bytes	optional	Payload was already provided in its binary representation.
url	string	optional	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

Image

Represents abstract image which sent by client to the service.

Field	Type	Label	Description
source	Source	required	Source of the image.

Detect

Face detect report.

Field	Type	Label	Description
face	Face	required	Face details.

Correlation

Face has been correlated to the persons' photos from base.

Field	Type	Label	Description
correlations	PersonCorrelation	repeated	List of face correlations with most similar persons sorted in descending order of similarity.

Identification

Person was identified on image.

Field	Type	Label	Description
identified_correlation	PersonCorrelation	optional	Info about correlation between face from image and photo of the identified person.May be missing if person was not identified.
threshold	double	required	Current threshold of identification.

PersonCorrelation

Correlation of certain original face with person’s photo from base.

Correlations are primarily **debugging information** that can be completely correct interpreted only by the vendor’s specialists and reflects the features of the **currently used** face recognition algorithm that **may change** in the future. Therefore, you **SHOULD NOT** make any conclusions based on the received coefficients, except for getting the top of the most similar persons in the database according to the **current** face recognition algorithm.

Field	Type	Label	Description

person_id	string	required	Identifier of person which face was correlated with original photo.
photo_id	string	required	Identifier of person's photo that fits best.
correlation	double	required	Measure of similarity between original face and person's photo from base.

Face

Face. Specifies original image as bounded context for inner structures. Specifies coordinate system in original image with origin that is placed in left top corner of original image. Specifies pixel as measuring unit in original image.

Field	Type	Label	Description
face_coords	FaceCoords	required	Face coordinates in original image.
cut	FaceCut	optional	Face cut from original image.

FaceCut

Face cut in some bounded context. Specifies cutted image as bounded context for inner structures. Specifies coordinate system inside cutted image with origin that is placed in left top corner of original image. Specifies pixel as measuring unit in cutted image.

Field	Type	Label	Description
image	Source	required	Image of face cut.**Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.
face_coords	FaceCoords	required	Face coordinates on face cut.

FaceCoords

Coordinates of face elements in some bounded context.

Field	Type	Label	Description

quad	Quad	required	Bounded quad where face is located.
eyes	EyeOuterCorners	optional	Eye corner coordinates.

EyeOuterCorners

Coordinates of outer (relative to the center of subject's face) corners of subject's eyes in some bounded context.

Field	Type	Label	Description
left	Point	required	Outer corner of left subject's eye.
right	Point	required	Outer corner of right subject's eye.

Quad

Coordinates of quad in some bounded context. Specifies quad as new bounded context for inner structures. Specifies coordinate system inside quad with origin that is placed in its left top corner, x-axis oriented to the top right point, y-axis oriented to the bottom left point. Specifies pixel as measuring unit in quad.

Field	Type	Label	Description
top_left	Point	required	Top left point of quad in some bounded context.
top_right	Point	required	Top right point of quad in some bounded context.
bottom_right	Point	required	Bottom right point of quad in some bounded context.
bottom_left	Point	required	Bottom left point of quad in some bounded context.

Point

2D geometry point. Coordinate system and measuring unit often provided by some bounded context.

Field	Type	Label	Description
x	double	required	X-coordinate of point in measuring units.
y	double	required	Y-coordinate of point in measuring units.

Source.Type

Defined types of payload.

Name	Number	Description
BINARY	1	Payload can be obtained in its binary representation.
URL	2	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

License Management API

Provides opportunities to manage License (LM) service.

The service provides the following features to automate the process of license management of the system:

1. Get the server fingerprint, which must be transferred to the vendor to obtain a license.
2. Try to install an obtained license and store it on the server if the DRM (Digital Rights Management) subsystem allows the system to function under this license: has a valid signature, not expired, suitable for the current installed product, etc..
3. Request information about the stored license or get its raw version (for example, to backup it).

Stored license can be retrieved from the server, but may be neither readable no suitable by the implementation in case of some internal changes on the server (hardware changes, different product installation etc.).

A server can store no more than one license at a time. Successful license installation overwrites the stored version.

Protocol

License Management service uses Protobuf messages as format for interchange (look for `LicenseManagement.proto` file).

Service name: `api-lm-service`.

Version: 1.0.

Content type of messages: `application/vnd.com.smilart/api.lm.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
get_stored_license	GetStoredLicense.Request	optional	Request to get the raw content of the stored license.
get_license_info	GetLicenseInfo.Request	optional	Request to get the installed product license information.
get_server_fingerprint	GetServerFingerprint.Request	optional	Request to get a server fingerprint.
set_license	SetLicense.Request	optional	Request to install a new license.

ResponseEnvelope

Defines envelope response messages from service.

Field	Type	Label	Description
malformed_protobuf	MalformedProtobuf	optional	Protobuf in request was considered malformed.
get_stored_license	GetStoredLicense.Response	optional	Response to `GetStoredLicense` message.
get_license_info	GetLicenseInfo.Response	optional	Response to `GetLicenseInfo` message.
get_server_fingerprint	GetServerFingerprint.Response	optional	Response to `GetServerFingerprint` message.
set_license	SetLicense.Response	optional	Response to `SetLicense` message.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description
message	string	required	Readable details of that decision.

GetStoredLicense

Defines messages of the retrieving raw content of the current license.

Field	Type	Label	Description

GetStoredLicense.Request

Field	Type	Label	Description

GetStoredLicense.Response

Response to the request to get the raw content of the current license.

Field	Type	Label	Description
license_content_provided	GetStoredLicense.Response.RawLicenseContentProvided	optional	Stored license was provided.
raw_license_is_missing	GetStoredLicense.Response.RawLicenseIsMissing	optional	Stored license was not provided due to its absence on the server.

GetStoredLicense.Response.RawLicenseContentProvided

Raw license content was provided.

Field	Type	Label	Description
raw_license	RawLicense	required	Raw content of the license.

GetStoredLicense.Response.RawLicenseIsMissing

There is no stored license on the server.

Field	Type	Label	Description

GetLicenseInfo

Defines messages of the retrieving installed product license information.

Field	Type	Label	Description

GetLicenseInfo.Request

Field	Type	Label	Description

GetLicenseInfo.Response

Response to the request to get the installed product license information.

Field	Type	Label	Description
license_info_provided	GetLicenseInfo.Response.LicenseInfoProvided	optional	Stored license was successfully parsed and license information can be provided.
license_is_missing	GetLicenseInfo.Response.LicenseIsMissing	optional	Product license information cannot be provided due to absence of a stored license on the server.
unknown_license_format	GetLicenseInfo.Response.UnknownLicenseFormat	optional	Product license information cannot be provided due to the inability to correctly parse the saved license.

GetLicenseInfo.Response.LicenseInfoProvided

Product license information was provided.

Field	Type	Label	Description
license_info	LicenseInfo	required	Product license information.

GetLicenseInfo.Response.LicenseIsMissing

There is no stored license on the server.

Field	Type	Label	Description

GetLicenseInfo.Response.UnknownLicenseFormat

License information cannot be correctly parsed.

Field	Type	Label	Description

GetServerFingerprint

Defines messages of the retrieving a server fingerprint required to obtain a license.

Field	Type	Label	Description

GetServerFingerprint.Request

Field	Type	Label	Description

GetServerFingerprint.Response

Response to the request to get the server fingerprint.

Field	Type	Label	Description
server_fingerprint_provided	GetServerFingerprint.Response.ServerFingerprintProvided	optional	Server fingerprint was provided.

GetServerFingerprint.Response.ServerFingerprintProvided

Server fingerprint was provided.

Field	Type	Label	Description
server_fingerprint	bytes	required	Opaque byte array as server fingerprint.

SetLicense

Defines messages for installation a new product license. By default, license will be accepted only if the **license allows the current installed product to operates at the moment**:

License is in its activity time interval. License is suitable for this server.

These assertions could be bypassed with optional request parameters.

Field	Type	Label	Description

SetLicense.DryRun

A request option which tests a license but in the case of passing all the supposed checks license would not be installed on the server. Response will be the same as without this option.

Field	Type	Label	Description

SetLicense.AllowActivationInFuture

A request option which bypasses a license activity time interval check if the license activity period has not begun and has not expired.

Field	Type	Label	Description

SetLicense.Request

Field	Type	Label	Description
raw_license	RawLicense	required	Raw version of a new license.
dry_run	SetLicense.DryRun	optional	If present, license would NOT really installed on the server. Use it to check a license applicability to the server.
allow_activation_in_future	SetLicense.AllowActivationInFuture	optional	If present, bypass license activity time interval check.

SetLicense.Response

Response to request to install a new product license.

Field	Type	Label	Description
license_installed	SetLicense.Response.LicenseInstalled	optional	License was installed and will be stored on the server.
bad_license	SetLicense.Response.BadLicense	optional	The license was not installed due to the fact that some not bypassed checks have not passed.
unknown_license_format	SetLicense.Response.UnknownLicenseFormat	optional	License was not installed due to the inability to correctly parse the license file.

SetLicense.Response.LicenseInstalled

New product license was installed.

Field	Type	Label	Description
license_info	LicenseInfo	required	Product license information.

SetLicense.Response.BadLicense

The license was not installed due to the fact that some not bypassed checks have not passed.

Field	Type	Label	Description
license_info	LicenseInfo	required	Product license information.

SetLicense.Response.UnknownLicenseFormat

The license information cannot be correctly parsed.

Field	Type	Label	Description

RawLicense

Raw license content.

Field	Type	Label	Description
content	bytes	required	Opaque byte array as license content.

LicenseInfo

Information about product license.

Field	Type	Label	Description
activation_date	int64	optional	License activation date in ms. If absent, then the license is not limited in time.
license_period_days	int64	optional	The number of days that the license will work from the activation date (include activation date). If absent, then the license is timeless.
license_product	string	required	License product name.
serial_number	string	required	License Serial Number.
checks	LicenseInfo.Checks	required	License checks.

LicenseInfo.Checks

License checks. The presence of some not passed checks denotes the system in a non operability state.

Field	Type	Label	Description
passed	bool	required	It is an aggregate value for all checks. True if all checks are passed.

product_check	LicenseInfo.Checks.ProductCheck	required	Information about the correspondence of the product name from the license and the installed product name.
activity_time_interval_check	LicenseInfo.Checks.ActivityTimeIntervalCheck	optional	Information about correspondence of the current server time and license activity time interval. If present, the license may be inactive at some moment according to licensing scheme.
fingerprint_check	LicenseInfo.Checks.FingerprintCheck	optional	Information about the correspondence of the server fingerprint to the server fingerprint in the license.

LicenseInfo.Checks.ProductCheck

Information about the correspondence of the product name from the license and the installed product name.

Field	Type	Label	Description
passed	bool	required	True, if the product name from the license and the installed product name match. Otherwise false.
server_product	string	required	Name of the installed product.

LicenseInfo.Checks.ActivityTimeIntervalCheck

Information about correspondence of the current server time and license activity time interval.

Field	Type	Label	Description
passed	bool	required	True, if at the time of the request the server time is in license activity period. Otherwise false.
server_date	int64	required	Date on the server in ms.

LicenseInfo.Checks.FingerprintCheck

Information about the correspondence of the server fingerprint to the server fingerprint in the license.

Field	Type	Label	Description
passed	bool	required	True, if server fingerprints match. Otherwise false.

Person Management API

The service provides functionality for manipulating base of persons.

ACID properties of storage intentionally are not part of that specification. Their description SHOULD be provided by the vendor implementation.

Protocol

Person Management service uses Protobuf messages as format for interchange (look for `PersonManagement.proto` file).

Service name: `api-persons-service`.

Version: 1.4.

Content type of messages: `application/vnd.com.smilart/api.person.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
add_person	AddPerson.Request	optional	Request to add person.
get_person	GetPerson.Request	optional	Request to get person.
remove_persons	RemovePersons.Request	optional	Request to remove persons.
keep_persons	KeepPersons.Request	optional	Request to keep persons.
update_person	UpdatePerson.Request	optional	Request to update person.
list_person_ids	ListPersonIds.Request	optional	Request to list identifiers of persons.

ResponseEnvelope

Defines envelope response messages from the service.

Field	Type	Label	Description
add_person	AddPerson.Response	optional	Response to `AddPerson` message.
get_person	GetPerson.Response	optional	Response to `GetPerson` message.
remove_persons	RemovePersons.Response	optional	Response to `RemovePersons` message.
keep_persons	KeepPersons.Response	optional	Response to `KeepPersons` message.
update_person	UpdatePerson.Response	optional	Response to `UpdatePerson` message.
malformed_protobuf	MalformedProtobuf	optional	Protobuf in request was considered malformed.
list_person_ids	ListPersonIds.Response	optional	Response to `ListPersonIds` message.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description
message	string	required	Readable details of that decision.

AddPerson

Defines messages of the person adding protocol.

Field	Type	Label	Description

AddPerson.DryRun

Option of request which force to return response the same as without that option but NOT ADD person to database

Field	Type	Label	Description

AddPerson.Request

Request for add person to the base.

Field	Type	Label	Description

images	Image	repeated	List of images desirable associated with a person.
on_multiple_faces	AddPerson.OnMultipleFaces	optional	Strategy for handling multiple faces on images (default is ON_MULTIPLE_FACES_TAKE_LARGEST).
on_no_faces	AddPerson.OnNoFaces	optional	Strategy for handling absence of faces on images (default is ON_NO_FACES_SKIP).
id	string	optional	Identifier of person that would be created.Provides capability to set identifier of the person on client side. If not set — it will be generated on server side.Non empty string. Max length is 50. ASCII symbols with codes [32, 126]. Should be unique.**ATTENTION!**It SHOULDN'T be a place to store any security critical info or personal, private information of the person.It SHOULD be a place to store "foreign key" (GUID, for example) or other relational information to bind person entity from recognition base with client base.
dry_run	AddPerson.DryRun	optional	If set: person would NOT really added to database.If this option set: user provided 'Identifier of person' would not be checked on duplicate.Except mentioned facts, response is independent of this option.

AddPerson.Response

Response to `AddPerson` request.

Field	Type	Label	Description
person_added	AddPerson.Response.PersonAdded	optional	Person was added.
unsupported_image_type	AddPerson.Response.UnsupportedImageType	optional	Person was not added due to the presence of images with unsupported type.
aborted	AddPerson.Response.Aborted	optional	Person was not added due to abort triggered by selected strategies of processing.
payload_is_too_large	AddPerson.Response.PayloadIsTooLarge	optional	Person was not added due to excess of payload limits.
unsupported_input_source	AddPerson.Response.UnsupportedInputSource	optional	Person was not added due to the presence of images with unsupported source.
duplicate_id	AddPerson.Response.DuplicatePersonId	optional	Person was not added due to presence of the person with the same id.
person_id_mismatch	AddPerson.Response.PersonIdMismatch	optional	Person was not added due to non-compliance with the requirements for the person's identifier format.
photo_id_uniqueness_violation	AddPerson.Response.PhotoIdUniquenessViolation	optional	Person was not added due to presence of the no unique identifiers of photos in request.
photo_id_format_violation	AddPerson.Response.PhotoIdFormatViolation	optional	Person was not added due to non-compliance by one of `Image` format of photo identifier.

AddPerson.Response.PersonAdded

The person was successfully added.

Field	Type	Label	Description
person	Person	required	Added person.

AddPerson.Response.UnsupportedImageType

Service implementation can't process images due to the presence of images with unsupported by the service implementation format of images.

Field	Type	Label	Description
image_ids	string	repeated	Ids of uploaded images of unsupported type.

AddPerson.Response.UnsupportedInputSource

Service implementation can't process images due to the presence of images with unsupported by the service implementation type of source.

Field	Type	Label	Description
type	Source.Type	repeated	Type of source that can't be processed.

AddPerson.Response.Aborted

Process was aborted by strategies of processing images.

Field	Type	Label	Description
multiple_faces_image_ids	string	repeated	Ids of uploaded images with multiple faces.
no_faces_image_ids	string	repeated	Ids of uploaded images with no faces.

AddPerson.Response.PayloadIsTooLarge

Payload of images is too large for processing.

Field	Type	Label	Description
actual_size	int32	required	Received payload in bytes.
maximum_size	int32	required	Maximum acceptable size of payload in bytes by the service implementation.

AddPerson.Response.DuplicatePersonId

Person with specified id already exists.

Field	Type	Label	Description

AddPerson.Response.PersonIdMismatch

Provided identifier of the person doesn't meet the requirements of the format.

Field	Type	Label	Description
message	string	required	Human-readable description of mismatch.

AddPerson.Response.PhotoIdUniquenessViolation

Violation of uniqueness of the photo identifier of the person.

Field	Type	Label	Description
ids	string	repeated	List of non unique identifiers of photos.

AddPerson.Response.PhotoIdFormatViolation

Provided identifier of the photo doesn't meet the requirements of the format.

Field	Type	Label	Description
message	string	required	Human-readable description of format violation.

GetPerson

Defines messages of the person retrieving protocol.

Field	Type	Label	Description

GetPerson.Request

Request for retrieving person from the service.

Field	Type	Label	Description
person_id	string	required	Identifier of the requested person.

GetPerson.Response

Response to `GetPerson` request.

Field	Type	Label	Description
person_found	GetPerson.Response.PersonFound	optional	Person was found.

person_not_found	GetPerson.Response.PersonNotFound	optional	Person was not found.

GetPerson.Response.PersonFound

Person was found and provided.

Field	Type	Label	Description
person	Person	required	Person state from base.

GetPerson.Response.PersonNotFound

Person was not found in base.

Field	Type	Label	Description
person_id	string	required	Identifier of the requested person.

RemovePersons

Defines messages of the persons removal protocol.

Field	Type	Label	Description

RemovePersons.Request

Request for removing persons from base.

- If specified person doesn't exist, then it won't influence on set of removing persons.

Field	Type	Label	Description
person_ids	string	repeated	Identifiers of the persons, which client wants to remove.

RemovePersons.Response

Response to `RemovePersons` request.

Field	Type	Label	Description
persons_removed	RemovePersons.Response.PersonsRemoved	optional	Persons were removed.

RemovePersons.Response.PersonsRemoved

Persons were removed.

Field	Type	Label	Description
failed_to_remove_ids	string	repeated	Identifiers of persons, which was not removed and still present in base after request was processed.

KeepPersons

Defines messages of the persons keep protocol.

Field	Type	Label	Description

KeepPersons.Request

Request for removal of all persons, except specified from base.

- If empty list is specified, then all persons in base will be removed.
- All non-existent persons in base won't influence on set of removing persons.

Field	Type	Label	Description
person_ids	string	repeated	Identifiers of persons which client want to keep.

KeepPersons.Response

Response to `KeepPersons` request.

Field	Type	Label	Description
persons_kept	KeepPersons.Response.PersonsKept	optional	Persons were kept.

KeepPersons.Response.PersonsKept

Persons were kept in base.

Field	Type	Label	Description
failed_to_keep_person_ids	string	repeated	List of person identifiers absent in base.
removed_count	int32	required	Amount of persons which was deleted in base.

failed_to_remove_count	int32	required	Amount of persons which identifiers are absent in request but still present in base after request was processed.

UpdatePerson

Defines messages of the person update protocol.

Field	Type	Label	Description

UpdatePerson.Request

Request for update list of photos of the specified person in base. At first it keeps only photos from `keep_photo_ids` and then adds new photos from `add_photos`. Maximum total size of images per request is limited by the server.

Field	Type	Label	Description
person_id	string	required	Identifier of the person to update.
add_photos	Image	repeated	Images which will be added to the person.
keep_photo_ids	string	repeated	Identifier of the photos to keep.If empty list is specified, then all photos of the person will be removed.All non-existent photos in base won't influence on set of removing photos.
on_multiple_faces	UpdatePerson.OnMultipleFaces	optional	Strategy for handling multiple faces on images (default is ON_MULTIPLE_FACES_TAKE_LARGEST).
on_no_faces	UpdatePerson.OnNoFaces	optional	Strategy for handling absence of faces on images (default is ON_NO_FACES_SKIP).

UpdatePerson.Response

Response to `UpdatePerson` request.

Field	Type	Label	Description

person_updated	UpdatePerson.Response.PersonUpdated	optional	Person was updated.
person_not_found	UpdatePerson.Response.PersonNotFound	optional	Person was not found.
unsupported_image_type	UpdatePerson.Response.UnsupportedImageType	optional	Person was not updated due to the presence of images with unsupported type.
aborted	UpdatePerson.Response.Aborted	optional	Person was not updated due to abort triggered by selected strategies of processing.
payload_is_too_large	UpdatePerson.Response.PayloadIsTooLarge	optional	Person was not updated due to excess of payload limits.
unsupported_input_source	UpdatePerson.Response.UnsupportedInputSource	optional	Person was not updated due to the presence of images with unsupported source.
photo_id_uniqueness_violation	UpdatePerson.Response.PhotoIdUniquenessViolation	optional	Person was not updated due to non-compliance of uniqueness of photo identifiers for the person.
photo_id_format_violation	UpdatePerson.Response.PhotoIdFormatViolation	optional	Person was not added due to non-compliance by one of `Image` format of photo identifier.

UpdatePerson.Response.PersonUpdated

Person was successfully updated.

Field	Type	Label	Description
person	Person	required	Person state after update.

UpdatePerson.Response.PersonNotFound

Requested person to update wasn't found.

Field	Type	Label	Description
person_id	string	required	Identifier of the person which wasn't found.

UpdatePerson.Response.UnsupportedImageType

Service implementation can't process images due to the presence of images with unsupported by the service implementation format of images.

Field	Type	Label	Description
image_ids	string	repeated	Identifiers of uploaded images of unsupported type.

UpdatePerson.Response.UnsupportedInputSource

Service implementation can't process images due to the presence of images with unsupported by the service implementation type of source.

Field	Type	Label	Description
type	Source.Type	repeated	Source which isn't supported.

UpdatePerson.Response.Aborted

Process was aborted by strategies of processing images.

Field	Type	Label	Description
multiple_faces_image_ids	string	repeated	Identifiers of uploaded image with multiple faces.
no_faces_image_ids	string	repeated	Identifiers of uploaded images with no faces.

UpdatePerson.Response.PayloadIsTooLarge

Payload of images is too large for processing.

Field	Type	Label	Description
actual_size	int32	required	Received payload in bytes.
maximum_size	int32	required	Maximum acceptable size of payload in bytes by the service implementation.

UpdatePerson.Response.PhotoIdUniquenessViolation

Violation of uniqueness of the photo identifier of the person.

Field	Type	Label	Description

ids	string	repeated	List of non unique identifiers of photos.

UpdatePerson.Response.PhotoIdFormatViolation

Provided identifier of the photo doesn't meet the requirements of the format.

Field	Type	Label	Description
message	string	required	Human-readable description of format violation.

ListPersonIds

Defines messages of the list identifiers of persons protocol.

Field	Type	Label	Description

ListPersonIds.Request

Request for list identifiers of persons in base.

Field	Type	Label	Description
offset	int32	optional	Option to offset over the specified number of identifiers.If offset is absent or zero then person identifiers list starts from the beginning.Otherwise it starts from offset.
limit	int32	optional	Option to limit the number of identifiers.if limit is a positive number then person identifiers list contains at most limit of identifiers.If limit is absent or zero then returns full list of person identifiers.
filter_by_person_id_substring	string	optional	Option for filtering the list of persons by substring in the person identifier.Case insensitive. If option is empty or is an empty string - this filter will not be applied.

ListPersonIds.Response

Response to `ListPersonIds` request.

Field	Type	Label	Description
person_ids_listed	ListPersonIds.Response.PersonIdsListed	optional	Identifiers of persons were listed.
negative_limit	ListPersonIds.Response.NegativeLimit	optional	Identifiers of persons were not listed due to negative limit was provided.
negative_offset	ListPersonIds.Response.NegativeOffset	optional	Identifiers of persons were not listed due to negative offset was provided.

ListPersonIds.Response.PersonIdsListed

Identifiers of persons were listed.

Field	Type	Label	Description
person_ids	string	repeated	Identifiers of persons from base.
number_of_persons	int32	optional	Number of persons in base.

ListPersonIds.Response.NegativeLimit

Limit cannot be a negative number.

Field	Type	Label	Description

ListPersonIds.Response.NegativeOffset

Offset cannot be a negative number.

Field	Type	Label	Description

Source

Source of some payload.

Field	Type	Label	Description
binary	bytes	optional	Payload was already provided in its binary representation.

url	string	optional	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

Image

Represents abstract image which sent by client to the service.

Field	Type	Label	Description
id	string	required	Identifier of image.
source	Source	required	Source of the image.
photo_id	string	optional	Identifier of photo that would be created.Provides capability to set identifier of the photo created by this image on client side. If not set — it will be generated on server side.Non empty string. Max length is 50. ASCII symbols with codes [32, 126]. Should be unique for the person.

Photo

Represents person's photo which sent by the service to the client. **Person service can dynamically select source type for photos at its discretion, and a client should be ready to retrieve photos of all types. Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.

Field	Type	Label	Description
id	string	required	Identifier of photo. Unique for the person.
source	Source	required	Source of photo.
creation_time	int64	optional	Photo creation time in ms.
auto_sampled	bool	optional	True if this photo was sampled by Adaptive Verification service during self-learning process, other false.

Person

Person entity.

Field	Type	Label	Description
id	string	required	Identifier of the person in base.
photos	Photo	repeated	Photos of the person.
creation_time	int64	optional	Person creation time in ms.
modification_time	int64	optional	Last person modification time in ms. It is updated on every modification of any person field, include initial person creation moment.

AddPerson.OnMultipleFaces

Defines strategy how to process images with more than one face found.

Name	Number	Description
ON_MULTIPLE_FACES_TAKE_LARGEST	1	The largest face from the image will be added to a person.
ON_MULTIPLE_FACES_SKIP	2	No faces from the image will be added to a person.
ON_MULTIPLE_FACES_ABORT	3	Abort processing when image with more than one faces found.

AddPerson.OnNoFaces

Defines strategy how to process images without faces.

Name	Number	Description
ON_NO_FACES_SKIP	1	Skip image without faces.
ON_NO_FACES_ABORT	2	Abort processing if image without faces found.

UpdatePerson.OnMultipleFaces

Defines strategy how to process images with more than one face found.

Name	Number	Description

ON_MULTIPLE_FACES_TAKE_LARGEST	1	The largest face from the image will be added to a person.
ON_MULTIPLE_FACES_SKIP	2	No faces from the image will be added to a person.
ON_MULTIPLE_FACES_ABORT	3	Abort processing when image with more than one faces found.

UpdatePerson.OnNoFaces

Defines strategy how to process images without faces.

Name	Number	Description
ON_NO_FACES_SKIP	1	Skip image without faces.
ON_NO_FACES_ABORT	2	Abort processing if image without faces found.

Source.Type

Defined types of payload.

Name	Number	Description
BINARY	1	Payload can be obtained in its binary representation.
URL	2	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

Photo Booth API

The service provides the way to get optimal set of faces of person who stands front of the camera to enroll the person using these photos.

There are several steps to get the set of faces:

1. The person who should be enrolled stands front of the camera.
2. The process of building the set of faces starts with message `Request.Start`.
3. The person moving his or her head in different positions provides to the service different images of his or her face.
4. The service takes the images depending on sampler scheme.
5. When all necessary images have collected, the service stops the sampling process and creates the optimal set of faces, which will return from the service.

Protocol

Photo Booth service uses Protobuf messages as format for interchange (look for `PhotoBooth.proto` file).

Service name: `api-photo-booth-service`.

Version: 3.0.

Content type of messages: `application/vnd.com.smilart/api.photo-booth.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

Each event is sent as `EventEnvelope` message with `event` field set according to type of event.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
start	Start.Request	optional	Request to start sampling process.
terminate	Terminate.Request	optional	Request to terminate sampling process.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description
message	string	required	Readable details of that decision.

ResponseEnvelope

Defines envelope response messages from service.

Field	Type	Label	Description
start	Start.Response	optional	Response to `Start` message.
terminate	Terminate.Response	optional	Response to `Terminate` message.
malformed_protobuf	MalformedProtobuf	optional	Protobuf in request was considered malformed.

EventEnvelope

Defines envelope event messages.

Field	Type	Label	Description
frame	Frame	optional	The frame from camera has been processed.
detect	Detect	optional	The face was detected on frame.
progress	Progress	optional	Sampling process has been progress.
failure	Failure	optional	Failure occurred during sampling process.
completed	Completed	optional	Sampling process was successfully finished.
partial_completed	PartialCompleted	optional	Sampling process was finished with partial success.

Start

Defines messages of the faces sampling start protocol from the camera.

Field	Type	Label	Description

Start.Request

Request for the start sampling process from specified camera.

The request returns parameters, which will use to collect images from the camera defined by `camera_pid` and starts streams of required events.

The returned stream contains the events of particular kind if options field for event of this kind is set in request.

Each process have a unique identifier `process_descriptor`. All events relate to this process have the same identifier.

The process of collecting images from the camera to create a set of optimal faces may have time restriction. It depends on service configuration.

Sampling scheme is set by the service and returned to the client in the reply, if process runs successfully.

Receiving events

The events are sent by the server according to specified destination.

Serialization guarantees

1. All events about new frame arrival are sent exactly in the same order as frames arrive.
2. All events about further processing of a frame are sent in the order processing happens.

Field	Type	Label	Description
camera_pid	string	required	Camera identifier to receive events from.
frame_options	FrameOptions	optional	Defines the content of event sent when new frame is about to be processed.If isn't set, no such events will be sent.
detect_options	DetectOptions	optional	Defines the content of event sent when face is found on a frame.If isn't set, no such events will be sent.
progress_options	ProgressOptions	optional	Defines the content of event sent when appearance of that face led to improvement of quality of selected faces.If isn't set, no such events will be sent.
destination	string	required	Defines the destination, where events SHOULD be sent by the service.Destination definition protocol SHOULD be provided by the vendor implementation.

Start.Response

Response to start sampling request.

Field	Type	Label	Description
started	Start.Response.Started	optional	Sampling process was started.
no_camera	Start.Response.NoCamera	optional	Camera was not found.

already_started	Start.Response.AlreadyStarted	optional	New sampling process from specified camera can't be started due to presence of already running another samplingprocess from that camera.
unsupported_destination	Start.Response.UnsupportedDestination	optional	Requested destination is not supported by the vendor implementation.

Start.Response.Started

Successful start of the process of face selection. Sampling process is finished, when area of interest is full. Area of interest is considered full, if each named group included in it is full. A group of head pose is considered full, if it's collected enough detects where face relates to the group. **Implementation of the service reserves the right to define strategy of group of head pose progress estimation.**

Field	Type	Label	Description
time_limit_seconds	int32	optional	Time restriction to catch faces from the camera. Absence or zero means no time restriction.
process_descriptor	string	required	Descriptor of the running process.
scheme	HeadPoseSamplingScheme	required	Sampler scheme of running process.

Start.Response.NoCamera

Camera defined as `camera_pid` cannot be used.

Field	Type	Label	Description

Start.Response.AlreadyStarted

Process collecting faces from `camera_pid` have been already running.

Field	Type	Label	Description

Start.Response.UnsupportedDestination

Service doesn't support the specified destination. Look the vendor implementation documentation for details.

Field	Type	Label	Description
description	string	required	A helpful description of the reason for rejection.

Terminate

Terminate process identified by descriptor.

Field	Type	Label	Description

Terminate.Request

Request to terminate the process of collecting faces. The message is idempotent. If the process is running or the process doesn't exist, the result will be the same. And repeated request doesn't influence the system.

Field	Type	Label	Description
process_descriptor	string	required	Descriptor of the process, which should be terminate.

Terminate.Response

Response to terminate request.

Field	Type	Label	Description
terminated	Terminate.Response.Terminated	optional	Specified sampling process has been terminated or it was not among active processes.

Terminate.Response.Terminated

Process was terminated.

Field	Type	Label	Description
process_descriptor	string	required	Process descriptor from the process.

Frame

A new frame has been received from the camera identified as `camera_pid`.

Field	Type	Label	Description

id	string	required	Identifier of `Frame` event.
timestamp_ms	uint64	optional	The time when frame was *received from camera* in milliseconds.It can differ from the time frame was *acquired by the camera*.Corresponds to the server time and can be compared to other server time only.
camera_pid	string	required	Identifier of the camera acquired the frame.
image	Source	required	Image of the frame.**PhotoBooth service can dynamically select source type for images at its discretion, and a client should be ready to retrieve images of all types.Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.
meta	MetaEntry	repeated	Additional information about the frame. Is not specified at the time.
process_descriptor	string	required	Descriptor of process, where the event was generated.

Detect

Face has been found on frame identified as `frame_id`.

Field	Type	Label	Description
id	string	required	Identifier of `Detect` event.

timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.
frame_id	string	required	Identifier of the frame detect was made on.
face	Face	optional	Face details.
head_pose	HeadPose	optional	Head pose on detect.
process_descriptor	string	required	Descriptor of process, where the event was generated.

Progress

The set of collected head pose was changed.

Field	Type	Label	Description
id	string	required	Identifier of `Progress` event.
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.
progress_percentage	double	required	Percent of progress in collecting of faces. Can be changed from 0 to 1.
statistics	HeadPosesCollectedStatistics	optional	Statistics of head poses, which was collected.
process_descriptor	string	required	Descriptor of process where the event was generated.

Completed

The message informs that the process completed and returns the optimal set of faces, which was gotten.

Field	Type	Label	Description
photos	SelectedPhotos	required	The set of optimal images.

process_descriptor	string	required	Descriptor of process, where the even was generated.
timestamp_ms	uint64	optional	The time when the process was completed in milliseconds.Corresponds to the server time and can be compared to other server time only.

PartialCompleted

Process was interrupted and returns the best result from the collected faces. If these photos are used, then the result of identification service might be worse, than if the photos would be gotten from `Completed` message. It's recommended to use this photo just if you don't have another way.

Field	Type	Label	Description
progress_percentage	double	required	Percent of progress, when the process was stopped.
photos	SelectedPhotos	required	The set of optimal photos.
process_descriptor	string	required	Descriptor of process, where the event was generated.
time_limit_exceeded	PartialCompleted.TimeLimitExceeded	optional	The cause of partially completed was stop due to exceeding time limit.
terminated	PartialCompleted.Terminated	optional	The cause of partially completed was stop due to terminate request.
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.

PartialCompleted.TimeLimitExceeded

Process exceeds time limit.

Field	Type	Label	Description

PartialCompleted.Terminated

Process was terminated.

Field	Type	Label	Description

Failure

The message informs about error, because the process was stopped and it's impossible to get a result.

Field	Type	Label	Description
message	string	required	human-readable error message.
process_descriptor	string	required	Descriptor of process where the event was generated.
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.

FrameOptions

Options of frames

Field	Type	Label	Description

DetectOptions

Options of face detects.

Field	Type	Label	Description
add_head_pose	bool	optional	Being set true adds information about head poses into detect event.
add_face	bool	optional	Being set true adds information about face into detect event.

ProgressOptions

Options of process notifications.

Field	Type	Label	Description

add_head_pose_statistics	bool	optional	Being set true adds statistics about collected head poses into progress event.

HeadPoseSamplingScheme

The service can be configured to use different sampler schemes of head pose.The scheme, which is used by the service, is returned in a reply to the request `Request.Start`.

Field	Type	Label	Description
grid3x3	HeadPoseSamplingScheme.Grid3x3	optional	Grid 3x3 will be used for sampling.

HeadPoseSamplingScheme.Grid3x3

Some set of the named groups might form area of interest. The service can be configured to work with following area of interest.

Field	Type	Label	Description
area_of_interest	HeadPoseSamplingScheme.Grid3x3.AreaOfInterest	required	Area of interest, which is used by the service, is returned in reply on the request `Request.Start`.

SelectedPhotos

Selected photos by the service.

Field	Type	Label	Description
photos	bytes	repeated	List of photos with its binary content.

HeadPose

Information about head pose.

Field	Type	Label	Description
position	HeadPoseNamedPosition	optional	Named position of head from the used scheme.

Point

2D geometry point. Coordinate system and measuring unit often provided by some bounded context.

Field	Type	Label	Description
x	double	required	X-coordinate of point in measuring units.
y	double	required	Y-coordinate of point in measuring units.

Quad

Coordinates of quad in some bounded context. Specifies quad as new bounded context for inner structures. Specifies coordinate system inside quad with origin that is placed in its left top corner, x-axis oriented to the top right point, y-axis oriented to the bottom left point. Specifies pixel as measuring unit in quad.

Field	Type	Label	Description
top_left	Point	required	Top left point of quad in some bounded context.
top_right	Point	required	Top right point of quad in some bounded context.
bottom_right	Point	required	Bottom right point of quad in some bounded context.
bottom_left	Point	required	Bottom left point of quad in some bounded context.

EyeOuterCorners

Coordinates of outer (relative to the center of subject's face) corners of subject's eyes in some bounded context.

Field	Type	Label	Description
left	Point	required	Outer corner of left subject's eye.
right	Point	required	Outer corner of right subject's eye.

FaceCoords

Coordinates of face elements in some bounded context.

Field	Type	Label	Description
quad	Quad	required	Bounded quad where face is located.
eyes	EyeOuterCorners	optional	Eye corner coordinates.

FaceCut

Face cut in some bounded context. Specifies cutted image as bounded context for inner structures. Specifies coordinate system inside cutted image with origin that is placed in left top corner of original image. Specifies pixel as measuring unit in cutted image.

Field	Type	Label	Description
image	Source	required	Image of face cut.**PhotoBooth service can dynamically select source type for images at its discretion, and a client should be ready to retrieve images of all types.Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.
face_coords	FaceCoords	required	Face coordinates on face cut.

Face

Face. Specifies original image as bounded context for inner structures. Specifies coordinate system in original image with origin that is placed in left top corner of original image. Specifies pixel as measuring unit in original image.

Field	Type	Label	Description
face_coords	FaceCoords	required	Face coordinates in original image.
cut	FaceCut	optional	Face cut from original image.

HeadPosesCollectedStatistics

Statistics head poses collected from start of process.

Field	Type	Label	Description

per_pose_statistics	HeadPosesCollectedStatistics.HeadPoseStatistics	repeated	list of element of head pose statistic.Must contain information just about no empty named group of head pose.

HeadPosesCollectedStatistics.HeadPoseStatistics

Statistics head poses grouped by named positions.

Field	Type	Label	Description
named_position	HeadPoseNamedPosition	required	Named group position of the head.
collected_percentage	float	required	Percent of collected detects for this named group.

MetaEntry

Storage of single key-value pair.

Field	Type	Label	Description
name	string	required	Key of entry.
value	string	required	Value of entry.

Source

Source of some payload.

Field	Type	Label	Description
binary	bytes	optional	Payload was already provided in its binary representation.
url	string	optional	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

HeadPoseSamplingScheme.Grid3x3.AreaOfInterest

Name	Number	Description

CROSS	1	Contains following named poses: `LEFTWARD_TURN`, `FORWARD_TURN`, `FORWARD_UPWARD_TURN`, `FORWARD_DOWNWARD_TURN`, `RIGHTWARD_TURN`.
HORIZONTAL	2	Contains following named poses: `LEFTWARD_TURN`, `FORWARD_TURN`, `RIGHTWARD_TURN`.
FORWARD_TURN	3	Contains following named poses: `FORWARD_TURN`.
ALL_INNER_POSES	4	Contains following named poses: `LEFTWARD_TURN`, `LEFTWARD_UPWARD_TURN`, `LEFTWARD_DOWNWARD_TURN`,`FORWARD_TURN`, `FORWARD_UPWARD_TURN`, `FORWARD_DOWNWARD_TURN`,`RIGHTWARD_TURN`, `RIGHTWARD_UPWARD_TURN`, `RIGHTWARD_DOWNWARD_TURN`.

Source.Type

Defined types of payload.

Name	Number	Description
BINARY	1	Payload can be obtained in its binary representation.
URL	2	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

HeadPoseNamedPosition

Defines recognizing poses of the head.

Name	Number	Description
LEFTWARD_TURN	1	Left direct pose.
LEFTWARD_UPWARD_TURN	2	Left top pose.
LEFTWARD_DOWNWARD_TURN	3	Left bottom pose.
FORWARD_TURN	4	Direct pose.
FORWARD_UPWARD_TURN	5	Top direct pose.
FORWARD_DOWNWARD_TURN	6	bottom direct pose.
RIGHTWARD_TURN	7	Right direct pose.
RIGHTWARD_UPWARD_TURN	8	Right top pose.

RIGHTWARD_DOWNWARD_TURN	9	Right bottom pose.
OUTSIDE_LEFTWARD_TURN	10	Outside of left border.
OUTSIDE_RIGHTWARD_TURN	11	Outside of right border.
OUTSIDE_UPWARD_TURN	12	Outside of top border.
OUTSIDE_DOWNWARD_TURN	13	Outside of bottom border.

Video Content Analytics API

Provides access to events generated during processing frames from camera.

Protocol

Video Content Analytics (VCA) service uses Protobuf messages as format for interchange (look for `VCA.proto` file).

Service name: `api-vca-service`.

Version: 1.1.

Content type of messages: `application/vnd.com.smilart/api.vca.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

Each event is sent as `EventEnvelope` message with `event` field set according to type of event.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
subscribe	Subscribe.Request	optional	Request to subscribe to VCA events.

ResponseEnvelope

Defines envelope response messages from service.

Field	Type	Label	Description
subscribe	Subscribe.Response	optional	Response to subscribe message.
malformed_protobuf	MalformedProtobuf	optional	Protobuf in request was considered malformed.

EventEnvelope

Defines envelope event messages.

Field	Type	Label	Description
eos	Eos	optional	Eos occurred.
frame	Frame	optional	The frame from camera has been processed.
detect	Detect	optional	The face was detected on frame.
correlation	Correlation	optional	The face was correlated with person's faces from base.
identification	Identification	optional	Person was identified.

Subscribe

Defines messages of the VCA event subscription protocol from the camera.

Field	Type	Label	Description

Subscribe.Request

Request for the subscription to VCA event stream from specified camera. The returned stream contains the events of particular kind if options field for event of this kind is set in request.

Response is `Subscribe.Response.Subscribed` message if subscription is successful, or other nested message of `Subscribe.Response` if not.

Subscription expires after `subscription_term_seconds` returned with response. If there are no subscriptions to stream, the stream can be shut down (no events will be sent to stream destination until new subscription to the stream). Thus it requires repeat subscription request before subscription expires to receive events longer than `subscription_term_seconds`.

Each new subscription request resets term of subscription.

Serialization guarantees

All events about new frame arrival are sent exactly in the same order as frames arrive. All events about further processing of a frame are sent in the order processing happens.

Subscription cost

Doesn't necessarily creates new event stream. The existing one may be used if exists.

Field	Type	Label	Description

camera_pid	string	required	Camera identified by pid frames from where will be used in processing.
frame_options	FrameOptions	optional	Defines the content of event sent when new frame is about to be processed. If isn't set, no such events will be sent
detect_options	DetectOptions	optional	Defines the content of event sent when face is found on a frame. If isn't set, no such events will be sent.
correlation_options	CorrelationOptions	optional	Defines the content of event sent when correlation of face to base is finished. If isn't set, no such events will be sent.
identification_options	IdentificationOptions	optional	Defines the content of event sent when person is identified on frame. If isn't set, no such events will be sent.

Subscribe.Response

Response to subscription request.

Field	Type	Label	Description
subscribed	Subscribe.Response.Subscribed	optional	Subscription succeeded.
no_camera	Subscribe.Response.NoCamera	optional	Camera was not found.

Subscribe.Response.Subscribed

Means successful subscription.

Field	Type	Label	Description

destination	string	required	Defines the destination, where events will be sent by the service.Destination definition protocol SHOULD be provided by the vendor implementation.
subscription_term_seconds	uint32	required	A period of time in seconds from the time request is processed during which events will be sent.

Subscribe.Response.NoCamera

Subscription couldn't be processed due to unavailability of the specified camera.

Field	Type	Label	Description
camera_pid	string	required	Camera identifier from request.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description
message	string	required	Readable details of that decision.

DetectOptions

Options of face detects.

Field	Type	Label	Description

FrameOptions

Options of frames.

Field	Type	Label	Description

CorrelationOptions

Options of correlations.

Correlations are primarily **debugging information** that can be completely correct interpreted only by the vendor's specialists and reflects the features of the **currently used** face recognition algorithm that **may change** in the future. Therefore, you **SHOULD NOT** make any conclusions based on the received coefficients, except for getting the top of the most similar persons in the database according to the **current** face recognition algorithm.

Field	Type	Label	Description
max_persons	uint32	optional	Maximal number of persons to be added to correlation event (default is 1).

IdentificationOptions

Options of identifications.

Field	Type	Label	Description

Frame

A new frame has been received from the camera identified as `camera_pid`.

Field	Type	Label	Description
id	string	required	Identifier of `Frame` event.
timestamp_ms	uint64	optional	The time when frame was *received from camera* in milliseconds.It can differ from the time frame was *acquired by the camera*.Corresponds to the server time and can be compared to other server time only.
camera_pid	string	required	Identifier of the camera acquired the frame.
image	Source	required	Image of the frame.**Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.

meta	MetaEntry	repeated	Additional information about the frame. Is not specified at the time.
original_frame_meta	MetaEntry	repeated	Stored metadata of original frame.

Detect

Face has been found on frame from camera.

Field	Type	Label	Description
id	string	required	Identifier of the `Detect` event.
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.
passage_id	string	required	Identifier of a passage. The passage is a sequence of frames containing face of the same person.
frame_id	string	required	Identifier of the frame detect was made on.
face	Face	required	Face details.

Correlation

Face has been correlated to the persons' photos from base.

Field	Type	Label	Description
id	string	required	Identifier of `Correlation` event.
timestamp_ms	uint64	optional	The time when correlation result was received in milliseconds.Corresponds to the server time and can be compared to other server time only.
correlations	PersonCorrelation	repeated	Correlation results to most similar persons.

frame_id	string	required	Identifier of the `Frame` event sent when frame was received from camera.
detect_id	string	required	Identifier of the `Detect` event sent when face was found.

Identification

Person was identified on frame from camera.

Field	Type	Label	Description
id	string	required	Identifier of the `Identification` event.
timestamp_ms	uint64	optional	The time when person was identified in milliseconds.Corresponds to the server time and can be compared to other server time only.
correlation	PersonCorrelation	required	Info about correlation between face from frame and photo of the identified person.
threshold	double	required	Current threshold of identification.
frame_id	string	required	Identifier of the `Frame` event which contains image where person was found.
detect_id	string	required	Identifier of the `Detect` event which contains face of identified person.
correlation_id	string	required	Identifier of the `Correlation` event which contains information about correlations of face on frame.

Eos

Field	Type	Label	Description

timestamp_ms	uint64	optional	The time when the end of stream was detected in milliseconds.Corresponds to the server time and can be compared to other server time only.

PersonCorrelation

Correlation of certain original face with person's photo from base.

Correlations are primarily **debugging information** that can be completely correct interpreted only by the vendor's specialists and reflects the features of the **currently used** face recognition algorithm that **may change** in the future. Therefore, you **SHOULD NOT** make any conclusions based on the received coefficients, except for getting the top of the most similar persons in the database according to the **current** face recognition algorithm.

Field	Type	Label	Description
person_id	string	required	Identifier of person which face was correlated with original photo.
photo_id	string	required	Identifier of person's photo that fits best.
correlation	double	required	Measure of similarity between original face and person's photo from base.

Face

Face. Specifies original image as bounded context for inner structures. Specifies coordinate system in original image with origin that is placed in left top corner of original image. Specifies pixel as measuring unit in original image.

Field	Type	Label	Description
face_coords	FaceCoords	required	Face coordinates in original image.
cut	FaceCut	optional	Face cut from original image.

FaceCut

Face cut in some bounded context. Specifies cutted image as bounded context for inner structures.

Specifies coordinate system inside cutted image with origin that is placed in left top corner of original image. Specifies pixel as measuring unit in cutted image.

Field	Type	Label	Description
image	Source	required	Image of face cut.**Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.
face_coords	FaceCoords	required	Face coordinates on face cut.

FaceCoords

Coordinates of face elements in some bounded context.

Field	Type	Label	Description
quad	Quad	required	Bounded quad where face is located.
eyes	EyeOuterCorners	optional	Eye corner coordinates.

EyeOuterCorners

Coordinates of outer (relative to the center of subject's face) corners of subject's eyes in some bounded context.

Field	Type	Label	Description
left	Point	required	Outer corner of left subject's eye.
right	Point	required	Outer corner of right subject's eye.

Quad

Coordinates of quad in some bounded context. Specifies quad as new bounded context for inner structures. Specifies coordinate system inside quad with origin that is placed in its left top corner, x-axis oriented to the top right point, y-axis oriented to the bottom left point. Specifies pixel as measuring unit in quad.

Field	Type	Label	Description
top_left	Point	required	Top left point of quad in some bounded context.

top_right	Point	required	Top right point of quad in some bounded context.
bottom_right	Point	required	Bottom right point of quad in some bounded context.
bottom_left	Point	required	Bottom left point of quad in some bounded context.

MetaEntry

Storage of single key-value pair.

Field	Type	Label	Description
name	string	required	Key of entry.
value	string	required	Value of entry.

Point

2D geometry point. Coordinate system and measuring unit often provided by some bounded context.

Field	Type	Label	Description
x	double	required	X-coordinate of point in measuring units.
y	double	required	Y-coordinate of point in measuring units.

Source

Source of some payload.

Field	Type	Label	Description
url	string	optional	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

Verification API

Implements verification scenario: is the person who stands in front of the camera a person from base.

Protocol

Verification service uses Protobuf messages as format for interchange (look for `Verification.proto` and `VerificationEvent.proto` files).

Service name: `api-verification-service`.

Version: 1.2.

Content type of messages from `Verification.proto`: `application/vnd.com.smilart/api.verification.service/protobuf`.

Each request must be sent as `RequestEnvelope` message with `request` field set properly.

Each response is sent as `ResponseEnvelope` message with `response` field set according to request.

Content type of messages from `VerificationEvent.proto`: `application/vnd.com.smilart/api.verification.event.service/protobuf`.

Each event is sent as `EventEnvelope` message with `event` field set according to type of event.

RequestEnvelope

Defines envelope request messages from client.

Field	Type	Label	Description
verify	Verify.Request	optional	Request to start of the verification process.
subscribe	Subscribe.Request	optional	Request to subscribe to verification events.

ResponseEnvelope

Defines envelope response messages from service.

Field	Type	Label	Description
verify	Verify.Response	optional	Response to `Verify` message.
malformed_protobuf	MalformedProtobuf	optional	Protobuf in request was considered malformed.
subscribe	Subscribe.Response	optional	Response to `Subscribe` message.

MalformedProtobuf

Send protobuf message was considered malformed.

Field	Type	Label	Description

message	string	required	Readable details of that decision.

Verify

Defines messages of the verification process start protocol of the person from the camera.

Field	Type	Label	Description

Verify.Request

Request for the start of the verification process from specified camera of specified person. Verification process executed during the specified timeout by correlation faces from specified camera with photos of the specified person in base.

Handling concurrent verification requests

1. If there is already running verification process with specified camera, then service sends `Terminated` message to the client that initiate running verification process and starts new verification process for client of incoming request from that camera of a new specified person.
2. It is not recommended to setup environment in which access to the service with same camera parameters is provided for multiple, not synchronized clients.

Estimation of the timeout and prolongation of verification

It is supposed that client should find balance between two following strategies:

1. Setup small enough (up to few seconds) timeout and repeatedly send same requests with sufficiently accurate timings (in according to the timeout) in to service to prolong verification of the same person.
2. Setup big enough timeout (about ten seconds) and send only one request per verification.

First one gives more granularity for every verification attempt, more resistant to handling unexpected malfunctions and reduce utilization of computer resources, but client should be able to send more requests and some of them should be sent in parallel.
Second one is more simple for usage for client, but due to work with longer time intervals less resistant to handling unexpected malfunctions and may consume more computer resources than needed.

Field	Type	Label	Description
camera_pid	string	required	Camera identified by pid frames from where would be used for verification.
person_id	string	required	Person identified by id whose photos will be used for verification.

verification_term_seconds	int32	required	Term of verification process that will be started.
id	string	required	Client generated identifier of verification of up to 128 UTF-8 characters.
threshold_name	string	optional	Name of the predefined verification threshold in the vendor implementation. Case insensitive.If not set — verification process will be started with threshold selected by implementation.List of available names SHOULD be provided by the vendor implementation. **ATTENTION!**You set this field at your own risk. If you do, ask for professional help.
abort_if_no_photos	bool	optional	Option to interrupt the verification process if the person has no photos.

Verify.Response

Response to verification request

Field	Type	Label	Description
internal_request_id	string	required	Server generated identifier of processed request.
verified	Verify.Response.Verified	optional	Person was verified.
not_verified	Verify.Response.NotVerified	optional	Person was not verified.
terminated	Verify.Response.Terminated	optional	Verification process was terminated.
person_not_found	Verify.Response.PersonNotFound	optional	Person was not found.
no_camera	Verify.Response.NoCamera	optional	Camera was not available.

threshold_name_not_found	Verify.Response.ThresholdNameNotFound	optional	Name of threshold was not found.
aborted	Verify.Response.Aborted	optional	Verification process was aborted.

Verify.Response.Verified

Specified person was successfully verified by face during timeout.

Field	Type	Label	Description

Verify.Response.NotVerified

Specified person was not verified by face during timeout.

Field	Type	Label	Description

Verify.Response.Terminated

Verification process of the specified person was started but was abruptly terminated before the expiration of timeout by another start request with the same camera pid.

Field	Type	Label	Description

Verify.Response.PersonNotFound

Verification process couldn’t be started due to absence of the specified person.

Field	Type	Label	Description

Verify.Response.NoCamera

Verification process couldn’t be started due to unavailability of the specified camera.

Field	Type	Label	Description

Verify.Response.ThresholdNameNotFound

Verification process couldn’t be started due to absence of the specified name of threshold.

Field	Type	Label	Description

Verify.Response.Aborted

Verification process has been aborted.

Field	Type	Label	Description

reason	Verify.Response.Aborted.Reason	required	Reason for the process interruption.

Subscribe

Defines messages of the verification event subscription protocol from the camera.

Field	Type	Label	Description

Subscribe.Request

Request for the subscription to verification event stream from specified camera. Provides capability for partial disclosure of internal state and actions of the verification algorithm by means of access to person verification related events such as receiving frames from camera, face detection results, correlation results, and etc. that forms observable behavior of the service. The returned stream contains the events of particular kind if options field for event of this kind is set in request. Subscription expires after `subscription_term_seconds` returned with response. If there are no subscriptions to stream, the stream can be shut down (no events will be sent to stream destination until new subscription to the stream). Thus it requires repeat subscription request before subscription expires to receive events longer than `subscription_term_seconds`.

Each new subscription request resets term of subscription.

Serialization guarantees

1. All events about new frame arrival are sent exactly in the same order as original frames arrive from camera.
2. All events about further processing of a frame are sent in the order processing happens.
3. Verification process creates context within which continuous sequence of frames will be processed. If frame is processed in context of running verification than event about request that initiate that verification will be sent before first frame that was used for that verification.
4. `LoggedRequest.VerifyRequest` event will be sent before `LoggedResponse.VerifyResponse` event of that verification will be sent.
5. After result of verification was sent in the corresponding `LoggedResponse.VerifyResponse` event no related events with request that initiate that verification will be sent.

Consistency and accuracy guarantees

1. Service doesn't ought to be "Single source of truth" for results of verification and other events when been used by someone other than clients that requests verification of persons (e.g when non empty set of event subscribers presents). Finally, the decision of service - a person has been verified or not, can be obtained only by receiving corresponding response to a client verification request that initiate that verification, not by obtaining published event about that response via subscription functionality.
2. Since publishing of events is an additional functionality which based on partial (intentionally or due to circumstances) disclosure of internal state and actions of decision making algorithm of main verification functionality, all information retrieved by subscription should be considered

as complementary and may be insufficient to make strong conclusions about system operation only on it (including for reasons beyond service control).

Field	Type	Label	Description
camera_pid	string	required	Camera identified by pid frames from where will be used for verification.
frame_options	Subscribe.Request.FrameLoggingOptions	optional	Being set adds frame events to stream.
detect_options	Subscribe.Request.FaceDetectLoggingOptions	optional	Being set adds detect events to stream.
correlation_options	Subscribe.Request.CorrelationLoggingOptions	optional	Being set adds correlation events to stream.
request_options	Subscribe.Request.VerifyRequestLoggingOptions	optional	Being set adds events of verify requests to stream.
response_options	Subscribe.Request.VerifyResponseLoggingOptions	optional	Being set adds events of verify response to stream.

Subscribe.Request.FrameLoggingOptions

Logging options of frames.

Field	Type	Label	Description

Subscribe.Request.FaceDetectLoggingOptions

Logging options of face detects.

Field	Type	Label	Description

Subscribe.Request.CorrelationLoggingOptions

Logging options of correlations.

Correlations are primarily **debugging information** that can be completely correct interpreted only by the vendor's specialists and reflects the features of the **currently used** face recognition algorithm that **may change** in the future. Therefore, you **SHOULD NOT** make any conclusions based on the received coefficients, except for getting the top of the most similar photos in the database according to the **current** face recognition algorithm.

Field	Type	Label	Description

max_photos	uint32	optional	Maximal number of photos to be added to correlation event (default is 1).

Subscribe.Request.VerifyRequestLoggingOptions

Logging options of verify requests.

Field	Type	Label	Description

Subscribe.Request.VerifyResponseLoggingOptions

Logging options of verify response.

Field	Type	Label	Description

Subscribe.Response

Response to subscription request.

Field	Type	Label	Description
subscribed	Subscribe.Response.Subscribed	optional	Subscription succeeded.
no_camera	Subscribe.Response.NoCamera	optional	Camera was not found.

Subscribe.Response.Subscribed

Subscription succeeded.

Field	Type	Label	Description
destination	string	required	Defines the destination, where events will be sent by the service.Destination definition protocol SHOULD be provided by the vendor implementation.
subscription_term_seconds	uint32	required	A period of time in seconds from the time request is processed during which events will be sent.

Subscribe.Response.NoCamera

Subscription couldn't be processed due to unavailability of the specified camera.

Field	Type	Label	Description
camera_pid	string	required	Camera identifier from request.

Verify.Response.Aborted.Reason

Defines the reason of abort.

Name	Number	Description
NO_PHOTOS	1	The person has no photos.

EventEnvelope

Defines envelope event messages.

Field	Type	Label	Description
eos	Eos	optional	Eos occurred.
frame	Frame	optional	The frame from camera has been processed.
face_detect	FaceDetect	optional	The face was detected on frame.
correlation	Correlation	optional	The face was correlated with person's faces from base.
incoming_service_request	LoggedRequest	optional	The incoming request to the service was received.
outgoing_service_response	LoggedResponse	optional	The outgoing response from the service was send.

Frame

Frame event.

Field	Type	Label	Description

timestamp_ms	uint64	optional	The time when frame was *received from camera* in milliseconds.It can differ from the time frame was *acquired by the camera*.Corresponds to the server time and can be compared to other server time only.
id	string	required	Identifier of this frame event.
internal_request_id	string	optional	Identifier of request in which context frame was processed.
camera_pid	string	required	The camera identified by pid frame from which came.
image	Source	required	Image of the frame.**Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.
meta	MetaEntry	repeated	Metadata of frame.
original_frame_meta	MetaEntry	repeated	Stored metadata of original frame.

FaceDetect

Face detect event.

Field	Type	Label	Description
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.
id	string	required	Identifier of this face detect event.
internal_request_id	string	required	Identifier of request in which context face detect was processed.

frame_id	string	required	Identifier of frame event which contains image where face was found.
face	Face	required	Face details.

Correlation

Correlation event.

Field	Type	Label	Description
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.
id	string	required	Identifier of this correlation event.
internal_request_id	string	required	Identifier of request in which context correlation was processed.
frame_id	string	required	Identifier of frame event which contains image where face that was used for correlation with person’s photos was found.
face_detect_id	string	required	Identifier of face detect event which contains face that was used for correlation with person’s photos.
correlations	PhotoCorrelation	repeated	Info on correlation to most similar person photos.

Eos

"End of stream" event. Special event to indicate that no events will appear in the stream until new subscription request. This event can occur at any time when server decides to close the stream (e.g. if camera events belong to is gone). If client still need the stream, a new Subscribe.Request can be sent.

Field	Type	Label	Description

timestamp_ms	uint64	optional	Server generated timestamp of event.

LoggedRequest

Defines messages about logged incoming request to the service from client.

Field	Type	Label	Description
internal_request_id	string	required	Server generated identifier of incoming request.
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.
verify	LoggedRequest.VerifyRequest	optional	Verify request came in service.

LoggedRequest.VerifyRequest

Request for the start of the verification process from specified camera of specified person.

Field	Type	Label	Description
person_id	string	required	Person identified by id whose photos will be used for verification.
camera_pid	string	required	Camera identified by pid frames from where would be used for verification.
verification_term_seconds	int32	required	Term of verification process that will be started.
client_request_id	string	required	Client generated identifier of verification of up to 128 UTF-8 characters.
threshold_name	string	optional	Verification threshold name if setup by client in request.
threshold	double	optional	Verification threshold if system could define it from the request.

abort_if_no_photos	bool	optional	Interruption settings if the system could define them from the request.

LoggedResponse

Defines messages about logged outcoming response from the service to incoming client request.

Field	Type	Label	Description
internal_request_id	string	required	Server generated identifier of incoming request.
timestamp_ms	uint64	optional	Server generated timestamp of event.Corresponds to the server time and can be compared to other server time only.
verify	LoggedResponse.VerifyResponse	optional	Response to Verify request was send.

LoggedResponse.VerifyResponse

Response to the start of the verification process from specified camera of specified person.

Field	Type	Label	Description
verified	LoggedResponse.VerifyResponse.Verified	optional	Person was verified.
not_verified	LoggedResponse.VerifyResponse.NotVerified	optional	Person was not verified.
terminated	LoggedResponse.VerifyResponse.Terminated	optional	Verification process was terminated.
person_not_found	LoggedResponse.VerifyResponse.PersonNotFound	optional	Person was not found.
failure	LoggedResponse.VerifyResponse.Failure	optional	Verification process was abruptly stopped.
threshold_name_not_found	LoggedResponse.VerifyResponse.ThresholdNameNotFound	optional	Name of threshold was not found.

LoggedResponse.VerifyResponse.Verified

Person was verified.

Field	Type	Label	Description

verified_correlation	PhotoCorrelation	required	Info about correlation between face from frame and photo of the verified person.
threshold	double	required	Current verification threshold.
frame_id	string	required	Identifier of frame event got from camera on which person was verified.
face_detect_id	string	required	Identifier of face detect event face from which was verified.
correlation_id	string	required	Identifier of correlation event which leads to successful verification of person.

LoggedResponse.VerifyResponse.NotVerified

Person was not verified.

Field	Type	Label	Description

LoggedResponse.VerifyResponse.Terminated

Verification process was terminated by another request.

Field	Type	Label	Description

LoggedResponse.VerifyResponse.PersonNotFound

Person was not found in base.

Field	Type	Label	Description

LoggedResponse.VerifyResponse.ThresholdNameNotFound

Name of threshold was not found.

Field	Type	Label	Description

LoggedResponse.VerifyResponse.Failure

Started verification process was abruptly stopped and it’s impossible to get a result.

Field	Type	Label	Description

message	string	required	Human-readable error message.

MetaEntry

Storage of single key-value pair.

Field	Type	Label	Description
name	string	required	Key of entry.
value	string	required	Value of entry.

PhotoCorrelation

Correlation of certain original photo with person's photo from base.

Correlations are primarily **debugging information** that can be completely correct interpreted only by the vendor's specialists and reflects the features of the **currently used** face recognition algorithm that **may change** in the future. Therefore, you **SHOULD NOT** make any conclusions based on the received coefficients, except for getting the top of the most similar photos in the database according to the **current** face recognition algorithm.

Field	Type	Label	Description
photo_id	string	required	Identifier of person's photo from base.
correlation	double	required	Measure of similarity between original face and person's photo from base.

Face

Face. Specifies original image as bounded context for inner structures. Specifies coordinate system in original image with origin that is placed in left top corner of original image. Specifies pixel as measuring unit in original image.

Field	Type	Label	Description
face_coords	FaceCoords	required	Face coordinates in original image.
cut	FaceCut	optional	Face cut from original image.

FaceCut

Face cut in some bounded context. Specifies cutted image as bounded context for inner structures. Specifies coordinate system inside cutted image with origin that is placed in left top corner of

original image. Specifies pixel as measuring unit in cutted image.

Field	Type	Label	Description
image	Source	required	Image of face cut.**Provided graphics file format intentionally is not part of that specification and SHOULD be specified by the vendor implementation**.
face_coords	FaceCoords	required	Face coordinates on face cut.

FaceCoords

Coordinates of face elements in some bounded context.

Field	Type	Label	Description
quad	Quad	required	Bounded quad where face is located.
eyes	EyeOuterCorners	optional	Eye corner coordinates.

EyeOuterCorners

Coordinates of outer (relative to the center of subject's face) corners of subject's eyes in some bounded context.

Field	Type	Label	Description
left	Point	required	Outer corner of left subject's eye.
right	Point	required	Outer corner of right subject's eye.

Quad

Coordinates of quad in some bounded context. Specifies quad as new bounded context for inner structures. Specifies coordinate system inside quad with origin that is placed in its left top corner, x-axis oriented to the top right point, y-axis oriented to the bottom left point. Specifies pixel as measuring unit in quad.

Field	Type	Label	Description
top_left	Point	required	Top left point of quad in some bounded context.

top_right	Point	required	Top right point of quad in some bounded context.
bottom_right	Point	required	Bottom right point of quad in some bounded context.
bottom_left	Point	required	Bottom left point of quad in some bounded context.

Point

2D geometry point. Coordinate system and measuring unit often provided by some bounded context.

Field	Type	Label	Description
x	double	required	X-coordinate of point in measuring units.
y	double	required	Y-coordinate of point in measuring units.

Source

Source of some payload.

Field	Type	Label	Description
url	string	optional	Payload can be obtained by means of protocol-dependent methods of retrieving resource by its URL.

Scalar Value Types

.proto Type	Notes	C++ Type	Java Type	Python Type
double		double	double	float
float		float	float	float
int32	Uses variable-length encoding. Inefficient for encoding negative numbers – if your field is likely to have negative values, use sint32 instead.	int32	int	int

int64	Uses variable-length encoding. Inefficient for encoding negative numbers – if your field is likely to have negative values, use sint64 instead.	int64	long	int/long
uint32	Uses variable-length encoding.	uint32	int	int/long
uint64	Uses variable-length encoding.	uint64	long	int/long
sint32	Uses variable-length encoding. Signed int value. These more efficiently encode negative numbers than regular int32s.	int32	int	int
sint64	Uses variable-length encoding. Signed int value. These more efficiently encode negative numbers than regular int64s.	int64	long	int/long
fixed32	Always four bytes. More efficient than uint32 if values are often greater than 2^28.	uint32	int	int
fixed64	Always eight bytes. More efficient than uint64 if values are often greater than 2^56.	uint64	long	int/long
sfixed32	Always four bytes.	int32	int	int
sfixed64	Always eight bytes.	int64	long	int/long
bool		bool	boolean	boolean
string	A string must always contain UTF-8 encoded or 7-bit ASCII text.	string	String	str/unicode

bytes	May contain any arbitrary sequence of bytes.	string	ByteString	str