SAFEGUARD
SAFETY REDEFINED

HRSense™
System specifications

www.TURNSTILES.us | patrick.mcallister@TURNSTILES.us | 303-670-1099

CONTENTS

1. INTRODUCTION

1.1. SYSTEM PURPOSE

HRSense™ by SafeGuard is a superior facerecognition and video-analytics product.

1.2. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

API	Application programming interface
DB	Database
DNN	Deep Neural Network
OS	Operating system
MP	Megapixel (million pixels)
PX	Pixel
SaaS	Software as a service
TLS	Transport layer security
UUID	Universally unique identifier

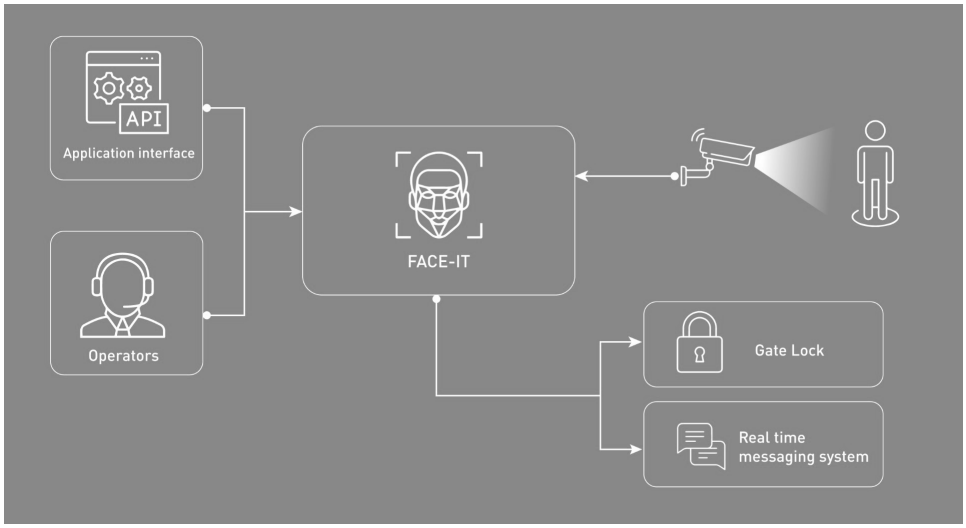
2. GENERAL SYSTEM DESCRIPTION

2.1. SYSTEM CONTEXT

HRSense™ by SafeGuard is a superior face recognition and video-analytics product.

2.1.1. Schematic system view

The schematic system



2.1.2. System interfaces

The section below lists the system's interfaces:

- REST API Iapplication programming interface (API) that conforms to the constraints of REST architectural style and allows HRSense™ interaction with REST web services
- Web interface for HRSense™ operators enables:

  Setup and configure the system

  View detections and identifications

  View system statistics
- Gates Controllers: applicative interface for HRSense™ to control the gates/doors locks
- Real-time messaging e.g. telegram
- Camera(Note: generally, legacy digital surveillance cameras can be used; there is no need to purchase dedicated cameras.)

2.2. SYSTEM CAPABILITIES

2.2.1. Face(s) recognition

Accuracy	all ethnicities	A single face suffices as model	Among10 best algorithms as NIST benchmark
Working modes	Real-time recognition	Video recordings analysis	Volunteering and in-the-wild recognitions
Adds-on	Instant learning and deployment	Real-time alerts to mobile devices	Anti-fraud (differentiate real person vs. digital picture)

2.2.2. Faces grouping

Accuracy	all ethnicities	Volunteering and in-the-wild recognitions
Working modes	No prior face learning required	Video recordings analysis
Use-cases	“Last seen” feature	Reconstruction of suspect pathway for forensic retroactive analysis

2.2.3. Granular rules

Organize target population	White and black and custom lists	Identification threshold / list	Real-time alerts / list
Organize detections	Identification threshold / camera	Geo-restrictions / camera / list	Anti-fraud / camera

2.2.4. Video analytics

Accuracy	Understanding video content	AI/ML expertise	
On-demand, E.g.	Raise alert when system detects people calling for help or immobile people	Identify people w/o helmets	Report when a pedestrian is noticed in forbidden areas

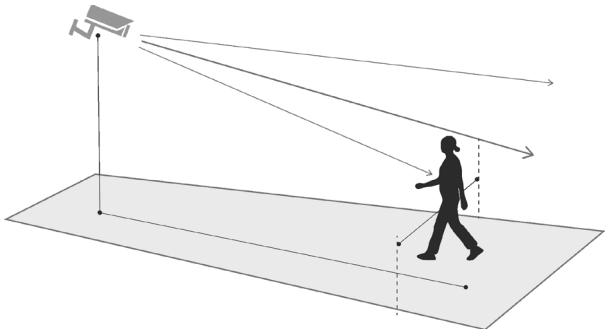
2.3. MAJOR SYSTEM CONSTRAINTS

2.3.1. Single face recognition model

- Minimum target face bounding box: 150*150 px
- Straight
- Uncovered e.g. without hat, sunglasses or face mask
- Distinct and bright

2.3.2. Optimum face recognition

- Minimum face bounding box for recognition: 80*80 px
- Well-lighted environment.
- Relatively close (1-15 m distance to the camera)
- Relatively soft angle (~2 m height to the camera)
- Clean lens, avoid obscureness (avoid setting camera against the sun)
- Avoid items covering faces, such as sunglasses, face mask



2.3.3. Target faces population volume

- Basic system supports 10K unique faces
- Beyond 10K faces – system support on demand Larger target faces population volume supported on-demand

2.3.4. Operational scenarios

Main use-cases of HRSense™.

2.3.5. Access control

- Volunteering recognition
- Real-time recognition
- Anti-fraud (differentiate real person vs. digital picture)

2.3.6. Suspects detection

- Real-time alerts
- In-the-wild recognitions
- Anti-fraud (differentiate real person vs. digital picture)

2.3.7. Forensic investigation

- Video recordings analysis
- Reconstruction of suspect pathway

2.3.8. In-the-wild recognitionStatistical analysis

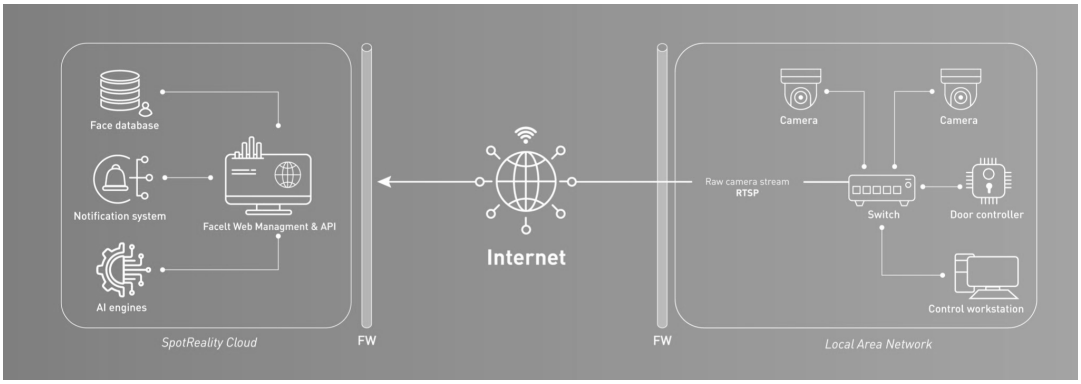
- In-the-wild recognition
- Real-time recognition and/or Video recordings analysis

3. SYSTEM CAPABILITIES, CONDITIONS, AND CONSTRAINTS

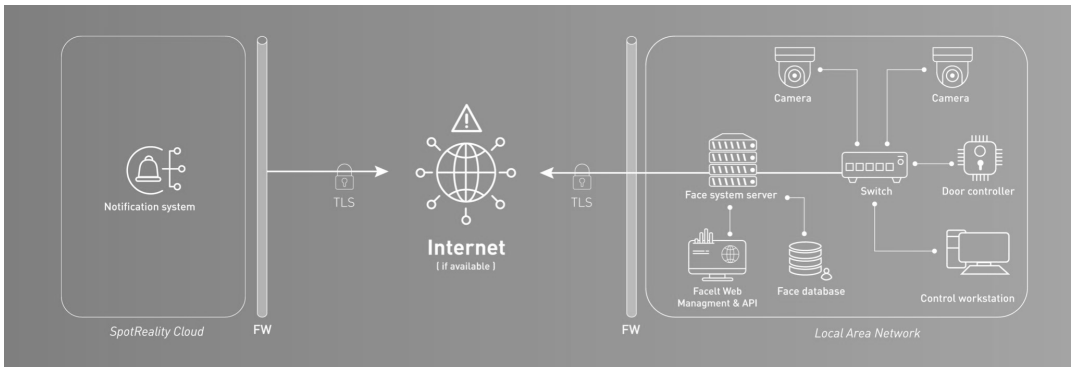
3.1. DEPLOYMENT MODULES

	Data processing	Video upload	Web services / UI available
SaaS (cloudw)	cloud	cloud	Web services/ UI available on all devices remotely
On-Premises	on premises	local servers	Web services/ UI available locally only (or optionally remotely)
Hybrid	Video processing locally, recognitions on cloud	Route videos to local servers	Web services/ UI available on all devices remotely
App	All processing on Android device (mobile, tablet...)	Video processed locally by device	Web services/ UI available on all devices remotely

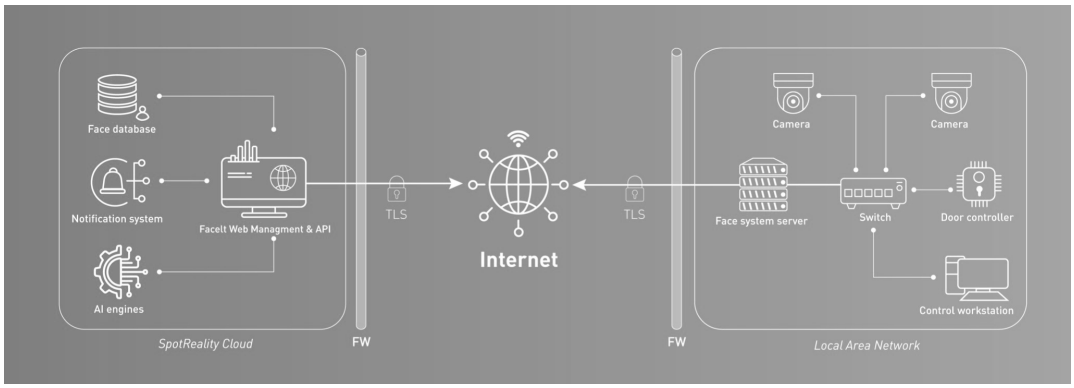
3.1.1. SaaS (cloud)



3.1.2. On-premises



3.1.3. Hybrid



3.2. HARDWARE

3.2.1. CPU based

Cameras	CPU – Intel® Core™ i7 gen 10	RAM	Storage
2	4 cores	10 GB	200 GB

3.2.2. GPU based

Cameras	CPU – Intel® Core™ i7 gen 8 GPU - Nvidia	RAM	Storage
2	2 cores	CPU - 8 GB GPU - 6 GB	200 GB

3.3. SOFTWARE

- OS - Ubuntu latest, official support from 18.04
- For GPU-based: Nvidia driver latest, official support from 460

3.4. SYSTEM PERFORMANCE CHARACTERISTICS

- Face detection: ~10ms
- Face classification: ~17ms
- Frames per second (FPS): 25
- Accuracy: 99.8% on the LFW dataset

3.5. SYSTEM SECURITY

3.5.1. Network

3.5.1.1. Web Interface

HRSense™ UI =is a web interface.

Each web page requires authentication and relevant permissions

To provide the most secure baseline configuration possible, by default HRSense™ is using TLS 1.2 or 1.3, with a secure set of TLS ciphers, using HTTPS.

The web service is installed behind a Nginx Reverse Proxy, providing a protection layer against a DDOS attack.

3.5.1.2. Users' management

The customer manages the system users:

- authorization
- access restriction/deletion
- password settings

Users' passwords are stored in a secure way, using the PBKDF2 algorithm with a SHA256 hash, a password stretching mechanism recommended by NIST.

3.5.1.3. RESTful API

HRSense™ uses a “RESTful-API” interface. Each API resource access requires an authorization token and permissions.

The grant of a token is conditional on the successful user authentication, based on customer prior authorization.

As for the Web interface, the APIs uses TLS 1.2 or 1.3, with a secure set of TLS ciphers, using HTTPS.

3.5.1.4. Push Notifications

HRSense™ supports notifications to a client over HTTPS. The customer needs to provide a valid certificate to allow usage of HTTPS.

Note: Currently there is no enforcement on the protocol type (with or without TLS); it is the customer’s choice whether a secure endpoint is used.

3.5.1.5. System monitoring and system upgrades

HRSense™ has a self-monitoring mechanism reporting to SafeGuard about faults or problems faced in the field, such as CPU overheating, disk capacity constraint, non-running services, errors in the logs.

HRSense™ has an upgrade mechanism allowing to check with SafeGuard system whether an upgrade is required. If an upgrade is required, the system retrieves the updated code image from the relevant servers. The upgrade mechanism is using a read-only token to fetch the image from the SafeGuard private container repository.

The communication supporting the mechanisms described above, is over HTTPS protocol (TLS), using a verified, valid, and signed certificate maintained by SafeGuard.

Note: The monitoring and the upgrade mechanisms can be disabled according to customer’s request.

3.5.1.6. Camera Live View

The system enables real-time display of the cameras’ streams with people face detection on the display in real-time.

Access to camera live view requires authentication and authorization to the system as for any other resource. The video is transmitted over the Secure Websocket protocol (WSS).

3.5.1.7. Access to edge computers

SafeGuard does not provide signed certificates on the customer servers (AKA Edges), rather, it uses self-signed certificates, ensuring encrypted communication (https) with the SafeGuard service.

It is possible to use signed certificates on edge devices. It is the customer's responsibility to provide, manage and update the certificates on the customer servers.

Note: SafeGuard is responsible for the security of the HRSense™ system; however, it is the customer's responsibility to secure the edge server's hosts.

3.5.2. Services

All the services of HRSense™ runs inside Docker Containers, with no access to the host system except for the persistent data such as logs, DB, images, and videos.

The containers run within a virtual network that is completely isolated from the host network, except for the Web and API services that open a listening port on the host network (443).

Furthermore, HRSense™ containers run only the strictly required software and libraries, therefore decreasing the attack surface.

3.5.3. Data Storage

The system stores data in a database and in a file system. For on-premises systems, the database and the file system are stored in the on-premises equipment.

3.5.3.1. Database (MongoDB)

The database is stored on the computer disk. Access to the data is solely through the engine running inside a Docker Container and requires authentication and authorization based on username and password.

The database service does not open a port on the external faced network interface, but rather only in the internal Docker network.

Note: By default, the database is not encrypted. Database encryption can be added upon agreement.

3.5.3.2. File System

The images and videos are stored in the file system on a disk in a separate place from the metadata, and accessible with root permission only.

Access to photos and videos does not require identification by default, as their ID's are generated by the UUID-V4 mechanism which has a high entropy (2122 or 5.3×1036 possibilities) that obviates the need for identification.

Note: Authentication while accessing images and videos can be added upon agreement.

3.5.3.3. Images and videos

By default, the system saves images of the faces and videos of events, although it is not necessary for the main operations of the system itself. On customer's demand, it is possible to disable saving the videos and the images. As a result, the system UI will not display images or videos.

3.5.3.4. Faces metadata

The data used by the system to identify a person, is not the image of the person's face but rather, it is a vector of numbers, outputted by a Deep Neural Network (DNN) and cannot in any way, be reverted into an image. The vector is stored in the DB.

3.6. SYSTEM MAINTAINABILITY

For on premises and hybrid deployment models, periodical backups of the database and of the file systems is required.