ITUPublications

International Telecommunication Union
Standardization Sector

ITU-T Technical Paper

(10/2022)

FSTP-VS-SDCA

Application of software-defined cameras in the surveillance industry

Technical Paper ITU-T FSTP-VS-SDCA

Application of software-defined cameras in the surveillance industry

Summary

This Technical Paper ITU-T FSTP-VS-SDCA introduces several use cases of software-defined cameras in multiple surveillance scenarios, analyses the possible requirements and pain points that customers may put forward, thereby providing guidance for further development of software-defined camera technology in the future. Also specifying the entire software-defined camera system ecosystem mechanism and security implementations. This Technical Paper aims to provide comprehensive guidance for software-defined camera (SDC) technology usage in the surveillance industry.

Keywords

SDC ecosystem, SDC security, software-defined camera, use cases.

Note

This is an informative ITU-T publication. Mandatory provisions, such as those found in ITU-T Recommendations, are outside the scope of this publication. This publication should only be referenced bibliographically in ITU-T Recommendations.

Contacts:

Yalan Zhang Huawei Technologies Co., Ltd China	Tel: E-mail:	+ 86-15952080053 zhangyalan@huawei.com
Longsheng Rao Huawei Technologies Co., Ltd China	Tel: E-mail:	+ 86- 13825274478 raolongsheng@huawei.com
Yuan Zhang China Telecom China	Tel: E-mail:	+86-18918588990 zhangy666@chinatelecom.cn
Yuwei Wang ICT CAS China	Tel: E-mail:	+86-10-62600726 wangyuwei@ict.ac.cn

Table of Contents

Page

Technical Paper ITU-T FSTP-VS-SDCA

Application of software-defined cameras in the surveillance industry

1 Scope

This Technical Paper aims to provide a comprehensive implementation guidance to SDC technology in multiple surveillance industries. It introduces several use cases of SDC in multiple surveillance scenarios, and possible requirements and pain points from customers are also shared. This Technical Paper also specifies the whole SDC system ecosystem including the phase of the algorithms development, online release, transaction and usage. At the end of this Technical Paper, SDC security is specified, which includes three aspects: software-defined camera (SDC) operating system (OS) security, SDC web security and SDC data security.

The scope of this Technical Paper includes:

– Use cases of SDC in multiple surveillance industries;

– SDC ecosystem mechanism;

– SDC security.

2 References

[ITU-T F.735.1] Recommendation ITU-T F.735.1 (2020), *Requirements for software-defined cameras.*

[ITU-T F.735.2] Recommendation ITU-T F.735.2 (2021), *Architecture and protocols for software-defined cameras.*

[ITU-T F.743.1] Recommendation ITU-T F.743.1 (2015), *Requirements for intelligent visual surveillance*.

[ITU-T H.626] Recommendation ITU-T H.626 (V2) (2019), *Architectural requirements for video surveillance system.*

[IETF RFC 9110] IETF RFC 9110 (2022), *HTTP Semantics.*

3 Definitions

3.1 Terms defined elsewhere

This Technical Paper uses the following terms defined elsewhere:

3.1.1 customer unit [ITU-T H.626]: A device located at the customer part of a video surveillance system and used to present multimedia information (such as audio, video, image, alarm signal, etc.) to the end-user.

3.1.2 intelligent premise unit [ITU-T F.743.1]: A kind of a premises unit (PU) that contains a premise intelligent video (PIV) module inside. A PIV can recognize the required information from input videos, and output recognition results. The recognized information includes that which are from event triggers and data acquisition.

3.1.3 premises unit [ITU-T H.626]: A device located at the remote part of a video surveillance system and used to capture multimedia information (such as audio, video, image, alarm signal, etc.) from a surveilled object.

3.1.4 reference architecture [b-ISO/IEC 26550]: Core architecture that captures the high-level design of a software and the systems product line including the architectural structure and texture (e.g., common rules and constraints) that constrains all the member products within a software and the systems product line.

3.1.5 software-defined camera [ITU-T F.735.1]: Software-defined camera is a kind of an intelligent premise unit (IPU) (see [ITU-T F.743.1]), which provides a technical approach to decouple hardware and software and to support algorithms on-demand deployment, online upgrade without services interrupting, continuous self-adaptive learning to adapt to various scenarios.

3.2 Terms defined in this Technical Paper

This Technical Paper defines the following term:

3.2.1 software-defined camera reference architecture (SDCRA): The reference architecture for the whole system of the software-defined camera, which includes the SDC system provider, SDC system user, and SDC system operator, illustrates the roles and activities, user view architecture, and functional view architecture through using the methodology defined in [b-ISO/IEC 26550].

4 Abbreviations and acronyms

This Technical Paper uses the following abbreviations and acronyms:

CU	Customer Unit
CRSF	Cross-Site Request Forgery
DEK	Data Encryption Key
HTTP	Hypertext Transfer Protocol
IPU	Intelligent Premise Unit
ITS	Intelligent Transportation System
KEK	Key Encryption Key
OS	Operating System
OTP	One-Time Programmable
PTZ	Pan/Tilt/Zoom
PU	Premises Unit
SDC	Software-Defined Camera
SDCRA	Software-Defined Camera Reference Architecture
SOI	Service-Oriented Interface
SQL	Structured Query Language
TLS	Transport Layer Security
XSS	Cross Site Scripting

5 Use cases of SDC in multiple surveillance industries

5.1 Purchasing new algorithms in the SDC system

Developer A is dedicated to the water-level measurement algorithm development and has completed the algorithms based on the SDC service-oriented interfaces (SOI) (defined in [ITU-T F.735.1]). A government department wants to implement water level measurements for all the rivers in their jurisdictions. The currently existing deployed cameras in all rivers do not have such intelligent analysis features. If the government department replaces all these cameras with new intelligent cameras having water-level measurement functions inside, this way costs are high. Another efficient and low-cost way is to buy a new algorithm from a professional algorithms developer and deploy this new algorithm in all the deployed cameras. Issues can be figured out from this use case. How can they buy such an algorithm, and how can they deploy the algorithm to all their cameras?

To solve such a problem, SDC may be a potential approach to achieve downloading a new algorithm from the algorithm warehouse and deploying the new algorithm to all the existing cameras in the surveillance network. The whole procedure can be illustrated in Figure 5-1.

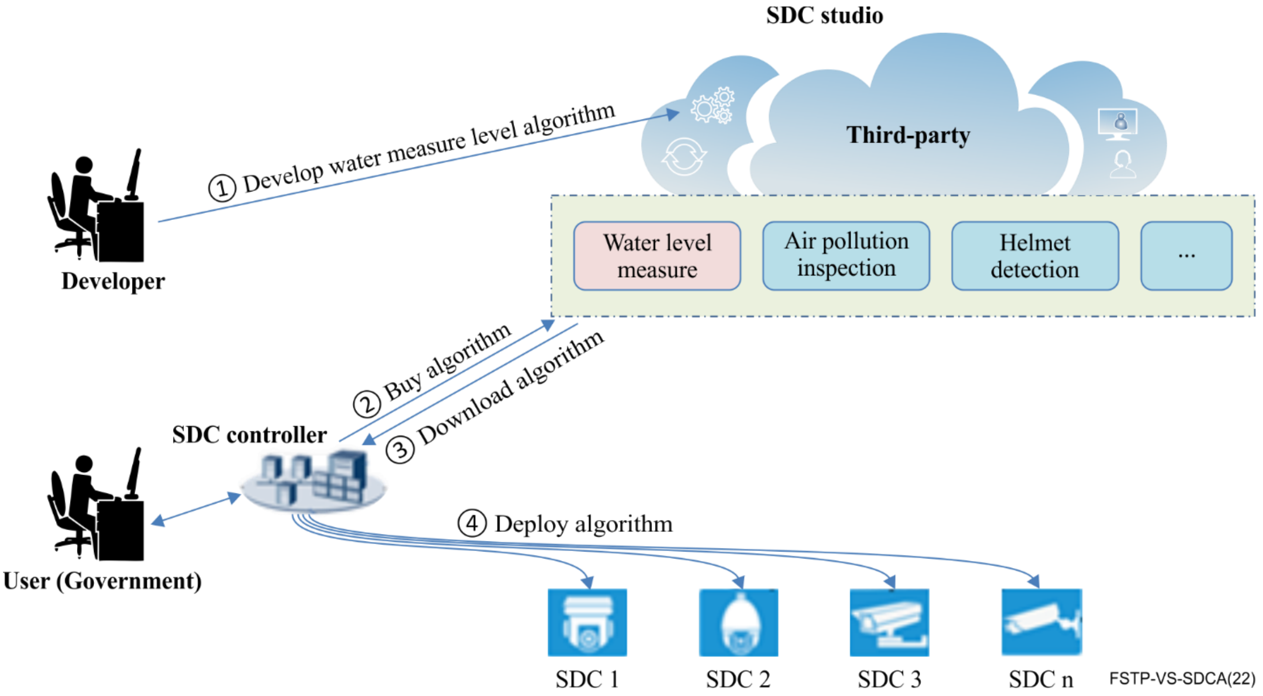


Figure 5-1 – The procedure of completing a new feature in the SDC system

5.2 Intelligent inspection of power-system

The power line operation and management are important for the whole power system, with the development in AI technology and video surveillance, intelligent inspections based on video surveillance tend to be common, and a large number of surveillance cameras are applied to the power grid inspections.

Due to massive scenarios and rapid changes in the power grid inspection, SDC's online algorithm deployment capabilities can effectively fulfil the effective deployment of various inspection algorithms (such as defect identification, power line failure detection, personnel safety detection, etc.) without other monitoring services interruption. In addition, the cables and substations of the power system are usually built, in remote areas and outdoors. These cameras are required to have an environment-adaptive function to ensure that a high-definition video is captured in any environment (such as rain, snow, fog, etc.), to meet the requirements for an accurate power inspection. In this case, the camera also needs to achieve the automatically adjusting image enhancement parameters.

5.3 Intelligent transformation

A large number of non-intelligent cameras or single-intelligent cameras (only has one intelligent algorithm inside) are deployed in the customer's existing environment. These cameras cannot meet the ever-increasing intelligent analysis requirements. Replacing all these non-intelligent cameras with intelligent ones would be very expensive. Hence, how to use non-intelligent cameras to achieve intelligent analysis? If intelligent analysis features are all deployed in the video surveillance platform, this method costs more response time and cannot meet the requirements of real-time scenarios (for example, face recognition, licence plate recognition, etc.).

In such a situation, the SDC can support connecting to other (N ≥ 1) non-intelligent cameras, obtain video from these cameras, complete video decoding and intelligent analysis, and generate analysis results to users. Therefore, SDC can be a potential way to complete the intelligent transmission of existing surveillance networks and reuse non-intelligent surveillance cameras.

5.4 Intelligent transportation system (ITS)

Case 1: Highway agglomerate fog detection

Most highway accidents are caused by agglomerate fog, which is quicker, thicker, and more dangerous than normal fog. Highway agglomerate fog is also called a "flowing killer". Therefore, the deployment of an agglomerate fog detection algorithm in front-end cameras to realize timely fog warnings and reduce traffic accidents is particularly important for the traffic management department.

If the traffic department decides to deploy an agglomerate fog detection algorithm in a specific area, or in a specific season to realize timely fog warning, these cameras need to support on-demand algorithm deployment which means the new algorithm can be installed and deleted at any time according to the user's demand. The whole procedure of a new algorithm deployment is shown in Figure 5-2, and the steps are as follows:

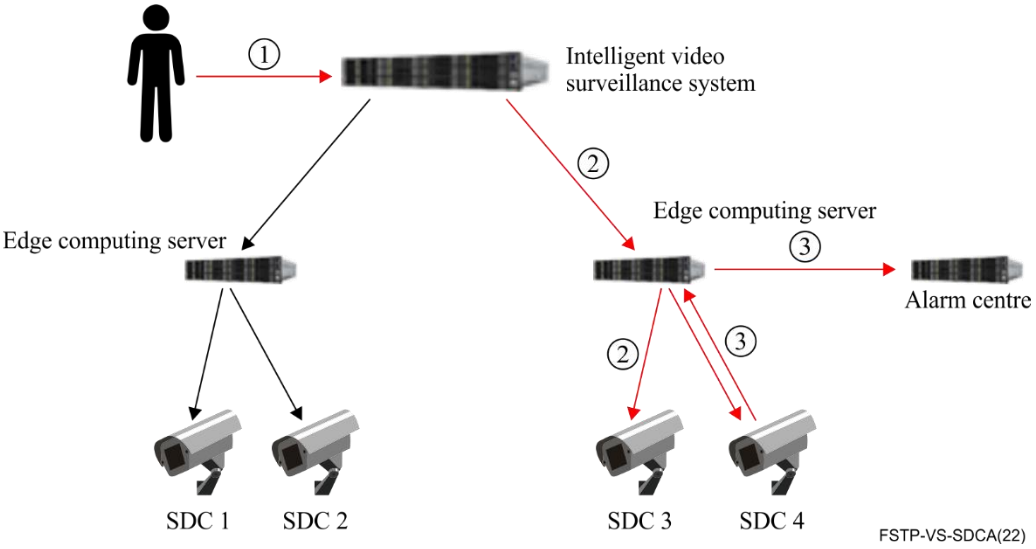


Figure 5-2 – Procedure of an agglomerate fog algorithm deployment

Step 1: The user logs into the image verification system (IVS) through the customer unit (CU) (defined in [ITU-T H.626]) chooses some intelligent premise units (IPUs) (defined in [ITU-T H.626]), which are deployed in the interested regions and completes the algorithm deployment function configuration. Then, the user submits the new algorithm deployment request to the IVS.

Step 2: The request is accepted by the specific edge computing server and the algorithm package is downloaded from the IVS centre or from a specific algorithm cloud server to this edge computing server. Then this edge computing server executes the deployment command to specific SDCs, the agglomerate fog detection algorithm is deployed in SDC 3 and SDC 4 without other business interruption.

Step 3: When the SDC detects the fog occurred, the alarm is reported to the edge computing server and this alarm is forwarded timely to the alarm centre.

Case 2: Electric vehicle licence plate recognition

Currently, a certain intersection camera can only perform licence plate recognition for traditional non-electric vehicles. Due to the issuance of electric vehicle licence plates in recent years, more electric licence plates have been used, and motorcycle licence plate recognition is also needed as the demand for accident identification and traffic accident detection increases. In this case, traffic administrators need to deploy a new licence plate detection algorithm to specific front-end cameras flexibly.

Case 3: Highway spillages detection

Spillages on roads and motorways can cause serious accidents and should be reported to the Department for infrastructure immediately. Therefore, the requirement for highway spillage detection is severe. This new algorithm needs to be deployed in the specific front-end cameras of highway roads.

5.5 Intelligent retail

SDC could be applied in smart retail solution scenarios on a large scale, proving customers with intelligent operation and management for stores including customer portraits, remote inspections, heat map analysis and other intelligent operation management, thereby helping chain stores solve anti-theft, high operation costs, and unreasonable placement problems of commodities. The deployment architecture for an intelligent retail solution is shown in Figure 5-3.

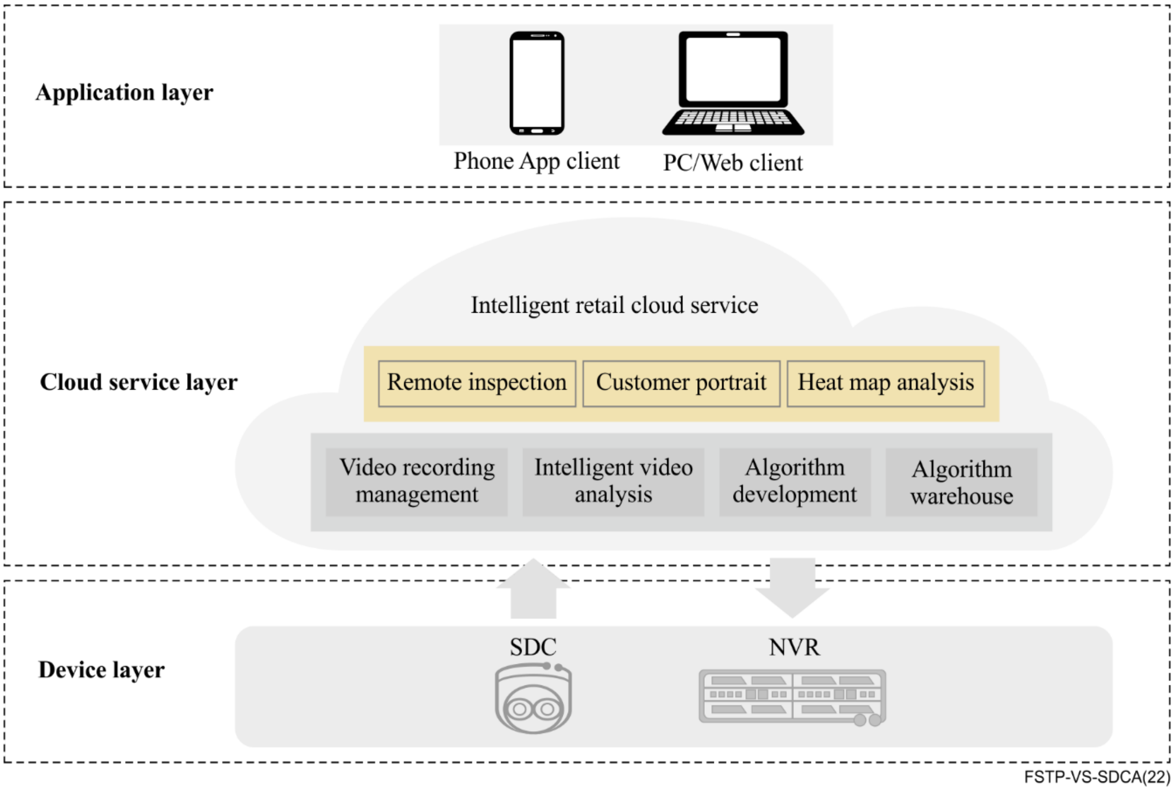


Figure 5-3 – Deployment architecture of an intelligent retail solution

Based on the sustainable algorithms upgrade function of SDCs, this solution could support the algorithm collaboration of cloud, edge, and devices. When a store manager wants to add a new intelligent analysis function in their entire monitoring system, such as off-job inspection (detecting whether the staff leaves work during working time) or dress code detection (detecting whether the chefs or employees wear work clothes or not), the store manager can browse all these algorithms online in the algorithm warehouse, select the most suitable algorithm and deploy the specific algorithm through the SDC website or platform. Rather than replacing these devices with advanced ones, this solution achieves the addition of intelligent functions and solves operational problems quickly and economically.

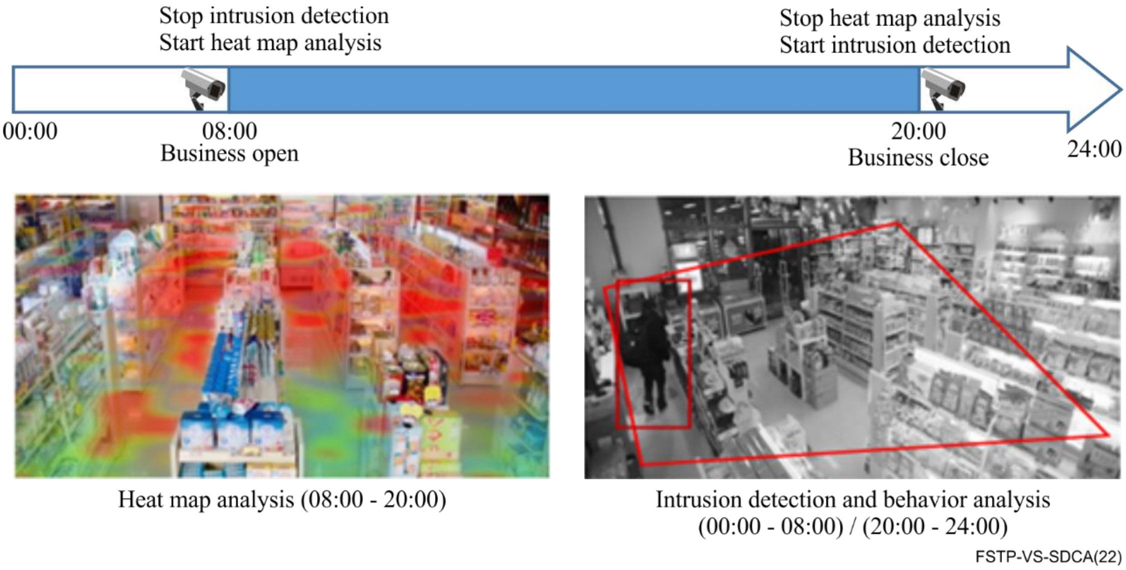


Figure 5-4 – Time-sharing and multiplexing of an intelligent algorithm in the SDC

SDCs can realize one camera for multiple purposes and adopt the way of time division and multiplexing of the cameras to support the integration of store security and smart operation. As shown in Figure 5-4, from 08:00 - 20:00, intrusion detection will be turned off and heat map analysis will be turned on, while at other times the heat map analysis will be turned off and intrusion detection will be turned on dynamically.

6 SDC ecosystem mechanism

6.1 SDC reference architecture (SDCRA)

6.1.1 Roles and activities in the SDC ecosystem

This clause introduces the related stakeholders in the SDC system, and their activities involved in developing and deploying a new algorithm. The whole procedure is shown in Figure 6-1.

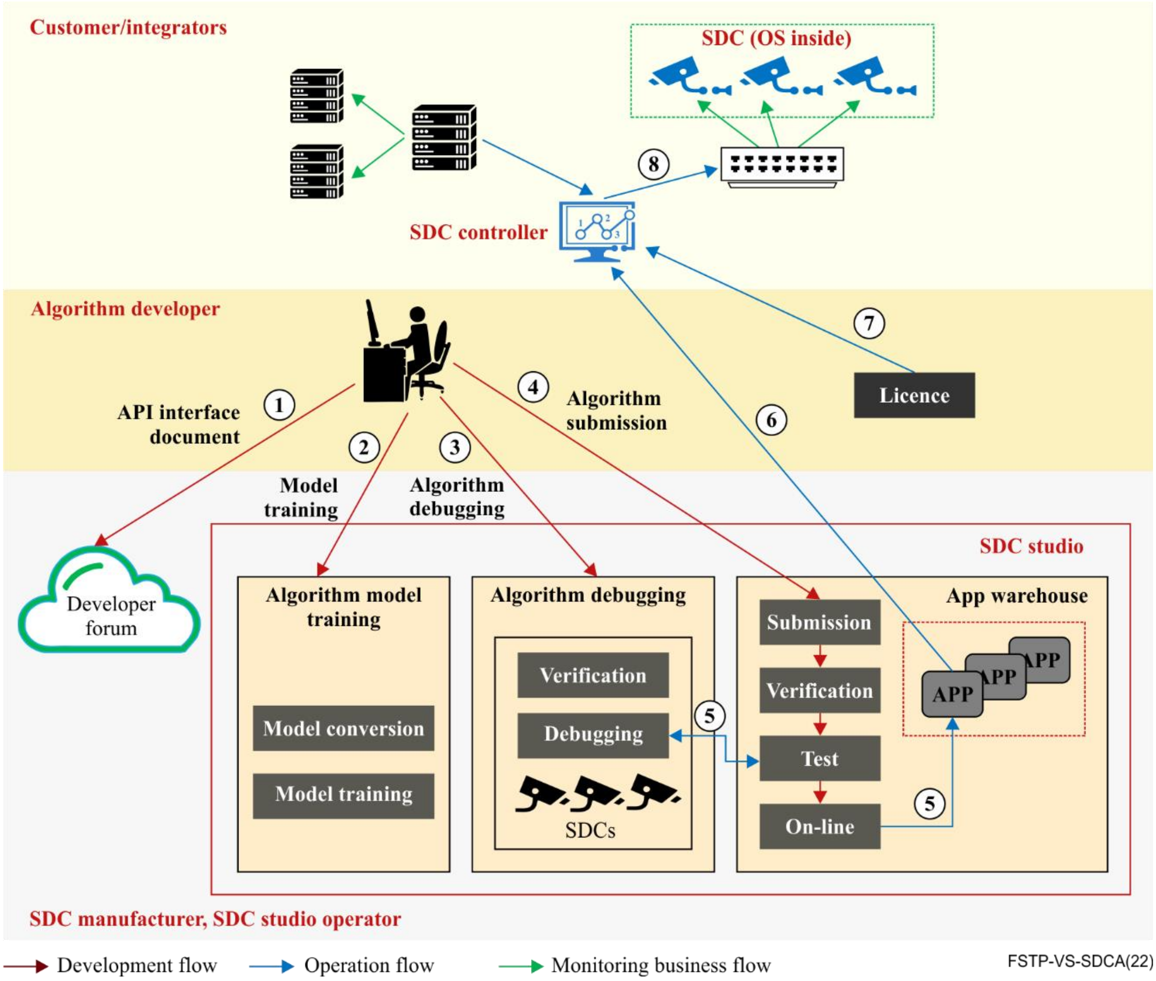


Figure 6-1 – Algorithm development, deployment, and management in the SDC system

There are four roles in the SDC system: customers (or integrators), algorithm developers, SDC manufacturers and SDC studio operators. Different roles have different activities, as follows:

– Algorithm developer: develop, debug and test the algorithm.

– SDC manufacturer: provide SDC hardware resource, SDC operating system (OS), and SDC studio service.

– Customer: execute the monitoring business, SDC control and configuration, algorithm deployment, etc.

– SDC studio operator: verify and release algorithm in the app warehouse and is responsible for software charging and operation.

The activities sequence description in this whole procedure are as follows:

a) The developer downloads the API interface documents and model conversion tools for the SDC developer forum;

b) The developer converts the model or trains the model in the SDC studio;

c) The developer develops an algorithm base on the API interface and debugs it in the SDC studio;

d) Developer packages the model and APP software and submits the package to the app warehouse;

e) SDC studio operator verifies and tests the APP software and releases it (online) in the app warehouse;

f) Customer purchases the app from the app warehouse through the SDC controller;

g) Customer gets an authenticated algorithm licence;

h) Customer deploys the algorithm in their SDCs through the SDC controller.

6.1.2 User view

The user view of the SDC system is used to show the different domains from the roles' perspective, which can help users to understand the high-level design of a system or product. The user view of the SDC system is shown in Figure 6-2.

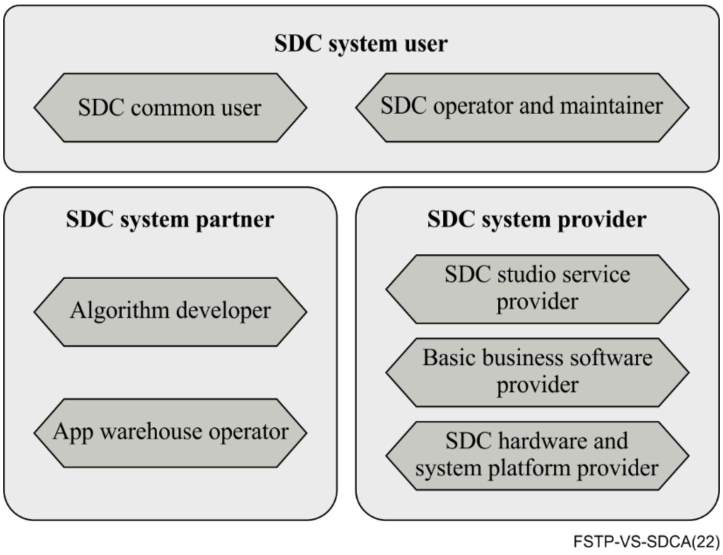

SDC system user
SDC common user
SDC operator and maintainer
SDC system partner
Algorithm developer
App warehouse operator
SDC system provider
SDC studio service provider
Basic business software provider
SDC hardware and system platform provider
FSTP-VS-SDCA(22)

Figure 6-2 – User view of the SDC system

There are three types of roles in the SDC system: SDC system user, SDC system partner and SDC system provider, each has sub-roles. SDC common user and SDC operator and maintainer are sub-roles of the SDC system user. The algorithm developer and the app warehouse operator are sub-roles of the SDC system partner. SDC studio service provider, basic business software provider and the SDC hardware and system platform provider are sub-roles of the SDC system provider. Each sub-roles are involved in different activities. These activities are as follows:

– The activities of SDC common user: Pan/tilt/zoom (PTZ) control, live video streaming, recording operation, intelligent analysis operation, SDC configuration, etc.

– The activities of the SDC operator and maintainer: user management, alarm event management, algorithm management, security management, advanced configuration of SDC, etc.

– The activities of algorithms developer: model training or model conversion, algorithm development, algorithm debugging, and algorithm submission in an SDC studio.

– The activities of the app warehouse operator: algorithm verification and testing, algorithm on-line, algorithm charging management, and algorithm marketing operation.

– The activities of an SDC studio service provider: providing app warehouse service, providing mode conversion tools set, providing algorithm management and operation tools (such as licence generation tool, algorithm on-line/off-line tool, charging tool, etc.).

– The activities of a basic business software provider: providing basic monitoring business software (such as live video streaming, recording, PTZ control, etc.).

– The activities of an SDC hardware and system platform provider: providing SDC basic hardware, providing basic hardware capability service-oriented interface (SOI), and providing basic common software and capability service-oriented interface.

6.1.3 Functional view

The SDC system functional architecture is defined in [ITU-T F.735.2], there are three functional modules in the SDC system: SDC (SDC OS included), SDC studio and SDC controller. For a detailed description of these three modules, please refer to clause 6 in [ITU-T F.735.2].

6.2 Procedure of algorithm development and deployment in an SDC system

This clause illustrates the entire procedure from the algorithm development to deployment, including the activities and different roles in the SDC system. The procedure can be illustrated in Figure 6-3. There are four phases in the entire algorithm life cycle from the developer to the end user.

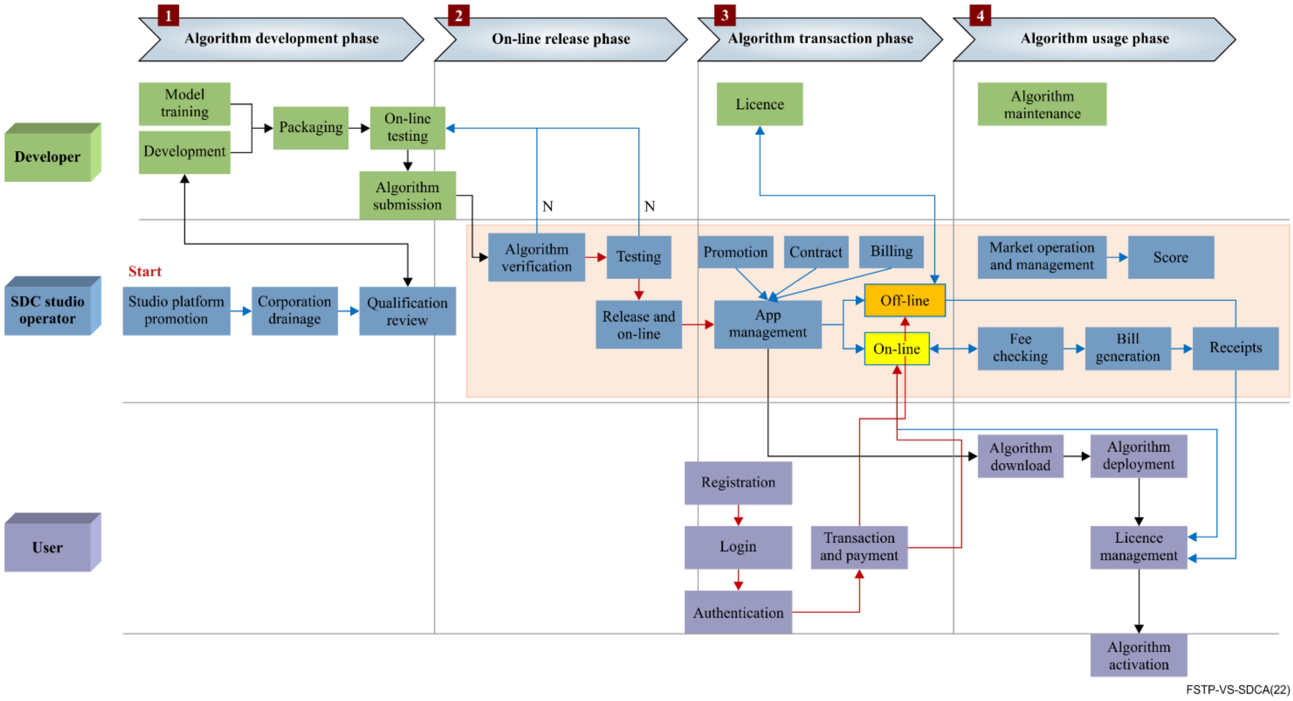


Figure 6-3 – Procedure of completing a new feature in the SDC system

The description of these phases are as follows:

a) Algorithm development phase:

   i) The SDC studio operator creates promotions for the SDC app warehouse, initiates corporation with the developer and completes the developer qualification review;

   ii) The developer trains the model and develops the algorithms, then packages the model and algorithm, completes online testing, and at last submits the package to the SDC studio.

b) Online release phase:

   The SDC studio operator verifies the algorithm, which is submitted by the developer, completes the algorithm testing, and then releases this algorithm online.

c) Algorithm transaction phase:

   i) The SDC studio provides the algorithm management service, including app promotion, billing, and contract generation;

   ii) The user registers in the SDC studio and logins, completes the authentication, chooses the specific algorithm and completes the payment through on-line or off-line way;

   iii) The APP licence is generated for the user; this licence is created by the developer.

d) Algorithm usage phase:

i) The SDC studio operator continues the fee checking, billing, and receipt generation according to the user's algorithm actual usage;

ii) User downloads the algorithm and completes the deployment in the user's SDCs, the algorithm is activated after the licence verification;

iii) In this phase the developer can receive feedback from the customer and make algorithm software maintenance.

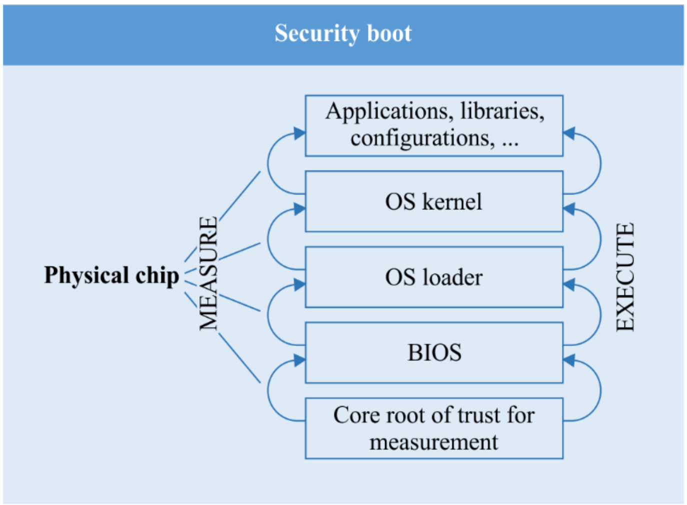
7 SDC security

7.1 SDC OS security

Massive cameras deployed outdoors are vulnerable to physical hijacking, tampering and information leakage. If these cameras are hijacked or invaded, it would immediately set off a chain reaction with unimaginable consequences. The information leakage can be prevented by the security of the video data transmission process, and the effective intrusion detection can prevent cameras from being hijacked.

7.1.1 Secure boot

The operating system (OS) boot process is the basis of all system behaviour. If the operating system is in an untrusted state, for example, if it has been tampered or destroyed before starting, then any security mechanism (shown in Figure 7-1) based on the system cannot assure the credibility of the system. The secure boot of the cameras can prevent the firmware or software from being tampered. The code of the security boot is solidified in the chip and the integrity of the boot programme is verified step by step during the device boot process. Only verified programmes are allowed to execute, thereby preventing malicious programmes from running on the device, effectively preventing camera physical hijacking, tampering and malicious swiping. It is recommended that the SDC support such a security boot mechanism.


Security boot
Applications, libraries, configurations, ...
OS kernel
OS loader
BIOS
Core root of trust for measurement
Physical chip
MEASURE
EXECUTE

FSTP-VS-SDCA(22)

Figure 7-1 – Secure boot mechanism

7.1.2 Intrusion detection and anti-hacking

The networked nature of the camera greatly increases its risk of being hijacked. If the camera is hacked, one faces the risk of business interruption, service instability and data theft. If the SDC supports an intrusion detection function and provides multi-layer intrusion detection capabilities, it will be able to detect hacker intrusion behaviour timely and prevent being hijacked.

7.2 SDC web security

7.2.1 Login password protection

SDC is required to provide relevant mechanisms including reminding the user to use a strong password instead of a simple one, default password modification, and password updates regularly to protect login passwords. The best practice is having initial accounts without passwords and force users to set strong passwords when they first login.

7.2.2 Web attack defence

Each SDC has a web portal used to manage the parameter configuration to ensure the normal operation of the camera. The best practise is that SDC provides a security mechanism to prevent traditional web attacks, such as a structured query language (SQL) injection, cross site scripting (XSS) and cross-site request forgery (CRSF), etc., through package verification of the browser, correctly using the Escape character and Hypertext transfer protocol (HTTP) header verification, etc.

NOTE – The Escape character is an alternative interpretation of the meta character (a character that has a special meaning to a computer programme, such as a shell interpreter or a regular expression engine).

7.3 SDC data protection

7.3.1 Data transmission leakage prevention

Unauthorized access and insecure network transmission are vulnerable to network eavesdropping, brute force, cookie replay, credential theft, confidential data leak, and data tampering induce attacks. The best practice is that SDC supports internationally recognized secure transmission protocols such as HTTP over transport layer security (TLS) (refer to [IETF RFC 9110], TLS 1.2 [IETF RFC 9110] and 802.1X [b-IEEE 802.1X-2010]), to ensure the security of the access network.

7.3.2 Data encrypted storage

Cameras are usually deployed in the outdoor areas. The SD card installed in the SDC may be easily unplugged and result in data leakage. SDC is recommended to support an encryption mechanism to protect these view pictures stored in the SD cards.

7.3.3 Key management security mechanism

Since data security depends on the encryption mechanism, and whether the encryption mechanism is secure depends on whether the key is secure, SDC is recommended to support a secure key management mechanism. A best practice for secure key management is a three-layer structure, including root key, key encryption key (KEK) and the data encryption key (DEK). The root key is used to encrypt KEK, and the KEK is used to encrypt DEK. In addition, the root key is protected by a hardware security mechanism (such as a one-time programmable (OTP)) and regularly updates the KEK.

Bibliography

[b-ISO/IEC 26550] ISO/IEC 26550:2015, *Software and systems engineering – Reference model for product line engineering and management.*
<https://www.iso.org/standard/69529.html>

[b-IEEE 802.1X-2010] IEEE 802.1X-2010, *IEEE Standard for Local and metropolitan area networks--Port-Based Network Access Control.*
<https://standards.ieee.org/ieee/802.1X/4384/>