ITUPublications
Recommendations

International Telecommunication Union
Standardization Sector

Recommendation

ITU-T F.748.21 (12/2022)

SERIES F: Non-telephone telecommunication services

Multimedia services

Requirements and framework for feature-based distributed intelligent systems

ITU-T F-SERIES RECOMMENDATIONS

NON-TELEPHONE TELECOMMUNICATION SERVICES

TELEGRAPH SERVICE	
Operating methods for the international public telegram service	F.1–F.19
The gentex network	F.20–F.29
Message switching	F.30–F.39
The international telemessage service	F.40–F.58
The international telex service	F.59–F.89
Statistics and publications on international telegraph services	F.90–F.99
Scheduled and leased communication services	F.100–F.104
Phototelegraph service	F.105–F.109
MOBILE SERVICE	
Mobile services and multidestination satellite services	F.110–F.159
TELEMATIC SERVICES	
Public facsimile service	F.160–F.199
Teletex service	F.200–F.299
Videotex service	F.300–F.349
General provisions for telematic services	F.350–F.399
MESSAGE HANDLING SERVICES	F.400–F.499
DIRECTORY SERVICES	F.500–F.549
DOCUMENT COMMUNICATION	
Document communication	F.550–F.579
Programming communication interfaces	F.580–F.599
DATA TRANSMISSION SERVICES	F.600–F.699
MULTIMEDIA SERVICES	**F.700–F.799**
ISDN SERVICES	F.800–F.849
UNIVERSAL PERSONAL TELECOMMUNICATION	F.850–F.899
ACCESSIBILITY AND HUMAN FACTORS	F.900–F.999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T F.748.21

Requirements and framework for feature-based distributed intelligent systems

Summary

Recommendation ITU-T F.748.21 introduces a classification of features and framework for feature-based distributed intelligent systems relevant to intelligent scenarios, specifying the service requirement, functional requirements and security requirements for feature-based distributed intelligent systems. Use cases for such systems are also described.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T F.748.21	2022-12-14	16	11.1002/1000/15198

Keywords

Feature classification, intelligent system, requirements.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Page

Recommendation ITU-T F.748.21

Requirements and framework for feature-based distributed intelligent systems

1 Scope

This Recommendation introduces a classification of features and framework for feature-based distributed intelligent systems relevant to intelligent scenarios, specifying the service requirement, functional requirements, performance requirements and security requirements for feature-based distributed intelligent systems.

The scope of this Recommendation includes:

− Classification of features in intelligent systems;

− Framework of feature-based distributed intelligent systems;

− Functional requirements for feature-based distributed intelligent systems;

− Security requirements for feature-based distributed intelligent systems;

− Use cases of feature-based intelligent applications.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

None.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 front-end equipment [b-ITU-T K.142]: Outdoor camera and associated ancillary equipment.

3.1.2 personally identifiable information [b-ISO/IEC 29100]: Any information that (a) can be used to identify the PII principal to whom such information relates, or (b) is or might be directly or indirectly linked to a PII principal.

NOTE – To determine whether a PII principal is identifiable, account should be taken of all the means which can reasonably be used by the privacy stakeholder holding the data, or by any other party, to identify that natural person.

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AI Artificial Intelligence

BRIEF	Binary Robust Independent Elementary Features
FAST	Features from Accelerated Segment Test
FDIS	Feature-based Distributed Intelligent Systems
FTP	File Transfer Protocol
HOG	Histogram Of Gradient
IoT	Internet of Things
IPC	Internet Protocol Camera
ITS	Intelligent Transportation System
PII	Personally Identifiable Information
RoI	Region of Interest
RTP	Real Time Transport Protocol
RTSP	Real Time Streaming Protocol
SIFT	Scale-Invariant Feature Transform
SURF	Speeded-Up Robust Features
UDP	User Datagram Protocol
YOLO	You Only Look Once

5 Conventions

In this Recommendation:

– The keywords "**is required to**" indicate a requirement which must be strictly followed and from which no deviation is permitted if conformance to this document is to be claimed.

– The keywords "**is recommended**" indicate a requirement which is recommended but which is not absolutely required. Thus this requirement needs not be present to claim conformance.

– The keyword "**may**" indicates an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the specification.

6 Overview

A feature-based intelligent system extracts and transmits features instead of original data, i.e., video, audio, text and point cloud. Compared with the intelligent systems based on original multimedia data, which require a large amount of storage and transmission bandwidth, and limited technological solutions use multimedia coding to reduce data volume; it could save bandwidth by extracting effective information from multimedia data.

In some latency sensitive intelligent systems, e.g., autonomous driving and industrial detection and control, it is crucial to process the data in real-time. Using features in addition to original data might allow fast searching and other AI tasks. A feature-based intelligent system is able to reduce latency in some scenarios.

With the recent advance of hardware technologies, the front-end equipment is capable of performing more calculations than before, and part of the computation load can be moved to the front end, rather than having most works done on the cloud side, with the front-end equipment performing only as data collectors and data transmitters. The end-to-end systems in the intelligent system can be divided into feature extraction and feature-based intelligent applications, the former deployed at the front end and the latter deployed at the back end, and the feature, i.e., an intermediate layer of the neural network, is extracted and communicated to offload the computing resource of the cloud side, balancing the computational load at the back end especially when the number of devices grows explosively. Figure 6-1 shows an end-to-end pipeline supporting intelligent applications based on features.

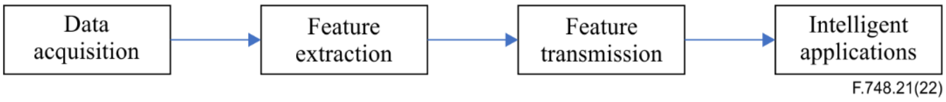

Figure 6-1 – An end-to-end pipeline of feature-based application

The fundamental process of feature-based distributed intelligent systems is as follows. First, the original data is fed into the feature extraction module, the data is processed and the different types of features are extracted according to the application configuration. Then the feature is compressed and packed for transmission and storage. Later, the intelligent application module receives the encoded bitstream and decodes it into original data or features. Last, the AI task is performed based on the selected feature. Use cases are presented in Appendix I.

7 Classification of features in intelligent systems

Features are widely used in machine vision, image processing and audio recognition, and so on. They are designed or learned to describe information or properties of the whole or partial content of an image, a video or an audio segment.

Various types of features are used in feature-based distributed intelligent systems (FDISs) to support AI tasks. The features are classified as signal domain features, latent domain features and semantic features.

For each classification, some of the commonly used features in AI tasks are listed in this section. The features extracted and transmitted will be selected and configured according to the combination of AI tasks supporting intelligent applications. The mapping of the AI task and features supporting the given AI task can be found in Appendix III.

The signal domain feature focuses on the colour and texture structure of an image, e.g., colour histogram, key points, edges and shapes, or motions and optical flows of a video, or signal properties of an audio segment, e.g., temporal shape, energy and spectral shape. The latent domain features describe the key information related to the semantics of the raw data. The semantic features describe semantic information, e.g., object, classification, velocity and action.

7.1 Signal domain features

Based on the types of raw data from which the features are extracted, the signal domain features can be classified into two types: visual features and audio features. Visual features are extracted from the visual data such as images, videos or point cloud. Audio features are extracted from the audio signal.

7.1.1 Visual features

7.1.1.1 Colour features

Colour features are extensively used for the extraction of visual features and are typically used in colour index-based image retrieval. Colour features are invariant to the translation or viewing angle. There are large numbers of colour features which are discussed in the literature including colour percentage, colour correlogram, colour histogram, colour coherence vector (CCV), and colour moments. Among these, colour moment is the simplest and most effective feature.

Colour features are extracted by calculating the statistical information of the pixels' colour distribution in the image. The features are represented in the vector format, in which each dimension represents one type of statistical information, such as value counts, mean value, standard derivation and so on.

7.1.1.2 Edge features

Edge features indicate the area where the image brightness changes sharply or has discontinuities. Edge features are used to mark the boundaries of objects or surfaces, which can significantly reduce the amount of data recommended to be processed and also filter out information that may be regarded as relevant areas.

Edge features are typically represented in the format of an image mask that has value in each position indicating whether a pixel belongs to an edge or not. The popular methods to detect edge features are Canny and Sobel.

Canny

Canny edge detector is an edge detection operator that uses a multistage algorithm to detect a wide range of edges in images.

Sobel

Sobel operator is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function.

7.1.1.3 Corner features

Corner features are also known as interest point features, which indicate the intersection of two edges. Corner features have well-defined positions and can be robustly detected.

Corner features are represented in the format of a vector that records the spatial position of the corner or interest point. A bundle of corner features can be used as an index of an image in the image retrieval task. The common methods to detect corner features are Harris corner detection and features from accelerated segment test (FAST) corner detection.

Harris corner detection

The Harris corner detector is a corner detection operator for extracting corners and inferring features of an image. It takes the differential of the corner score into account with reference to direction.

FAST: Features from accelerated segment test

FAST is a corner detection method. It is faster than many other well-known feature extraction methods, such as the difference of Gaussians (DoG) used by the scale-invariant feature transform (SIFT) and Harris detectors. The FAST corner detector is suitable for real-time video processing applications given its high-speed performance.

7.1.1.4 Feature descriptions

For an object in an image, descriptions of key points or interesting points can be used to identify the object while in image retrieval. To form a reliable feature description that is agnostic to image scale, rotation and illumination change, a feature description with complex information such as key points, scale and gradients is designed.

The feature descriptions are represented in a data structure consisting of spatial position, scale and description vector and so on. Typically, a bundle of feature descriptions is used together to identify a single object or a whole image. The common feature description generation methods are SIFT, speeded-up robust features (SURF), histogram of gradient (HOG), and binary robust independent elementary features (BRIEF).

SIFT

SIFT is a classical feature with the property of invariance to scale, rotation, luminance and viewport. Its robustness makes it widely used in image retrieval, object detection and image coding.

SIFT is defined to describe the keypoint inside an image by performing the following operations:

i) Keypoint detection in scale pyramid

To detect keypoints invariant to scale, a scale pyramid is built by filtering the input image with the Gaussian filter with a different scale. The scale pyramid generated by a Gaussian filter with different parameters is referred to as an octave. The difference between the Gaussian space is derived and keypoints are detected in each different space in each of the octaves.

ii) Gradient computation for keypoints

To ensure rotational invariance, gradients of each keypoint are derived by comparing the pixel values in the window centred by the keypoint.

iii) Feature description of keypoints

Gradients in the window centred by the keypoint are gathered together to form a gradient vector, which serves as the feature of the keypoint.

SURF

SURF is partly inspired by the SIFT descriptor. The standard version of SURF is several times faster than SIFT. To detect interest points, SURF uses an integer approximation of the determinant of Hessian blob detector, which can be computed with three integer operations using a precomputed integral image. Its feature descriptor is based on the sum of the Haar wavelet response around the point of interest. It can also be computed with the aid of the integral image.

HOG

HOG first counts occurrences of gradient orientation in localized portions of an image. The second step is orientation binning by creating the cell histograms. Each pixel within the cell casts a weighted vote for an orientation-based histogram channel based on the values found in the gradient computation. To account for changes in illumination and contrast, the gradient strengths must be locally normalized, which requires grouping the cells together into larger, spatially connected blocks. The HOG descriptor is then the concatenated vector of the components of the normalized cell histograms from all of the block regions.

BRIEF

BRIEF converts image patches into a binary feature vector so that together they can represent an object. In BRIEF, each key point is described by a feature vector which is a 128−512 bit string. BRIEF deals with the image at the pixel level so it is very noise-sensitive. By pre-smoothing the patch, this sensitivity can be reduced, thus increasing the stability and repeatability of the descriptors. BRIEF uses a Gaussian kernel for smoothing images. BRIEF is very fast both to build and match. BRIEF outperforms other fast descriptors such as SURF and SIFT in terms of speed and in terms of recognition rate in many cases.

7.1.1.5 Motion features

Motion features indicate the change in the position of an object relative to its surroundings or a change in the surroundings relative to an object. Motion features are detected from video sequences where a temporal change of contents occurs.

Typically, motion features are represented in the format of a motion vector indicating the direction and distance between the starting position and the ending position of an object or an image area when it moves in a video for a time slot.

7.1.2 Audio features

audio features include time domain, spectral domain and cepstral domain features.

Time domain features

The windowing technique is employed and the long non-stationary signal is analysed as short chunks of the quasi-stationary signal. Windowing can be seen as multiplying a signal with a window function that is zero everywhere except in the region of interest. The resultant windowed signal is the subset of the original signal which is passed through the window; for the rest of the time the signal is zero.

Frequency domain features

To analyse a signal in terms of frequency, the time domain signal is converted into a frequency domain signal using Fourier transform or auto-regression analysis. Frequency domain analysis is a tool of utmost importance in audio signal processing.

Cepstral domain features

A cepstrum is obtained by taking the inverse Fourier transform of the logarithm of the spectrum of the signal. There is a complex, power, phase and real cepstrum. Among all of these, the power cepstrum is the one most relevant to speech signal processing. The analysis of the cepstrum is called cepstrum analysis, quefrency analysis (equivalent to frequency analysis in spectrum domain) or liftering (equivalent to the filtering in spectrum domain). The cepstrum features are mainly used in pitch detection, speech recognition and speech enhancement.

7.2 Latent domain features

Deep neural networks (DNN) have been widely used for intelligent analysis, where the DNN may be allocated on the mobile or client side. With the benefit of cloud computing, resources and energy at the front end can be significantly reduced, which is referred to as collaborative intelligence. Instead of images or videos, deep features in the latent domain are extracted from the raw data, compressed by occupying fewer bits than the compression of raw data occupies and then transported to the cloud server for further analysis, as shown in Figure 7-1.

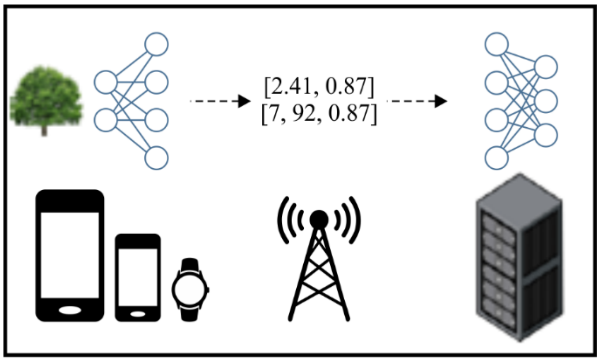

Figure 7-1 – Collaborative intelligence realized by client and cloud

The latent domain features are the intermediate representation output by the convolutional layers or fully connected layers in the neural network. They can be different when extracted from different layers of the neural network. From the shallow layer to the deep layer, the information of deep features changes from the pixel domain to the semantic domain, as shown in Figure 7-2. The data format of the latent domain features is often tensors with multiple dimensions, typically three dimensions including weight, height and channel. Figure 7-2 shows the visualization of the deep features from different layers. The feature maps in the same layer but different channels are shown in the same row.

The latent domain feature is related to the high-level semantic information of the content instead of describing the exact texture, colour or shape of the content.

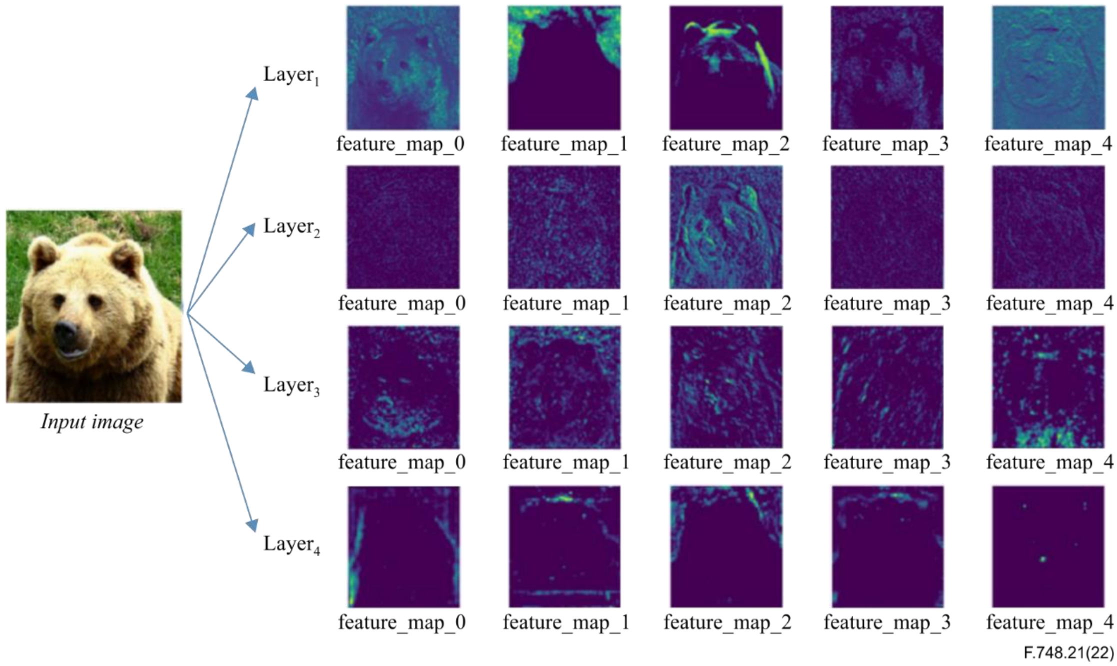

Figure 7-2 – Visualization of the latent domain features from different layers

7.3 Semantic feature

Semantic features are highly related to specific AI tasks. Commonly used semantic features can be classified into regions, types and semantic labels.

7.3.1 Region features

Segmentation mask feature

Image segmentation is the process of partitioning a digital image into multiple image segments, and its goal is to simplify the representation of an image into something more meaningful and easier to analyse. Image segmentation is typically used to locate objects and boundaries in images.

The segmentation mask feature is represented by a mask format which assigns a label to every pixel in an image such that pixels with the same label share certain characteristics.

Region of interest feature

A region of interest (RoI) is a proposed region from an image, where the region attracts human or machine interest. While watching a video or image, people's attention is more probably concentrated on interesting objects (such as persons, dogs or cats), moving objects or events, leading them to ignore the other contents in the same video or image. Machine learning has been widely used to extract the RoI from the video or image to highlight the important area of the video or image.

Typically, a salient mask is generated to indicate the RoI, which assigns different saliency values for each pixel to indicate the importance of the pixel, e.g., the pixel region containing a pedestrian or car is assigned a high saliency value while the region containing the sky or water is assigned a low saliency value.

Bounding box feature

A bounding box is an imaginary rectangle that serves as a point of reference for object detection and creates a collision box for that object. Data annotators draw these rectangles over images, outlining the object of interest within each image by defining its X and Y coordinates. This makes it easier for machine learning algorithms to find what they are looking for and to determine collision paths and it conserves valuable computing resources. Bounding boxes are one of the most popular image annotation techniques in deep learning. Compared with other image processing methods, this method can reduce costs and increase annotation efficiency.

The bounding box is represented in the format of a data structure including the position of the up-left corner of the bounding box and the size of the rectangular box.

7.3.2 Semantic labels

Class labels

The classical problem in computer vision, image processing and machine vision is that of determining whether or not image data contains some specific object, feature or activity. Different varieties of the recognition problem are described: 1) Object recognition (also called object classification) is when one or several pre-specified or learned objects or object classes can be recognized, usually together with their 2D positions in the image or 3D poses in the scene. 2) Identification is when an individual instance of an object is recognized. Examples include identification of a specific person's face or fingerprint, identification of handwritten digits or identification of a specific vehicle. 3) Detection is when the image data are scanned for a specific condition. Detection based on relatively simple and fast computations is sometimes used for finding smaller regions of interesting image data which can be further analysed by more computationally demanding techniques to produce a correct interpretation.

The format of the class label is either an index indicating the extract class of a pixel or region or a vector indicating the probability of each class that a pixel or region belongs to.

Motion labels

Properties of moving objects in videos include position, moving speed, moving direction and trace.

Vehicle attribute labels

Similar to persons, cars have their own attributes such as licence plate, size, position, shape, vehicle type, primary colour and so on.

8 Service requirements of feature-based distributed intelligent systems

8.1 General requirement

Feature-based distributed intelligent systems (FDISs) are designed to support feature-based intelligent applications. There are some different categories in FDISs.

8.2 Data acquisition

Original input data of feature-based distributed intelligent systems are recommended to be acquired using specific front-end equipment based on scenarios. For instance, image data in industrial applications is recommended to be captured by industrial cameras with high sensitivity and low noise. It is recommended to back up a copy of the original data in case of accidental damage in the front-end. Moreover, the original data is recommended to have sufficient resolution, especially for image and audio data.

8.3 Pre-processing

The pre-processing procedure is recommended to pre-process the original data (image, video, or audio) into particular formats, in order to extract features such as textures for the given AI tasks.

8.4 Data representation

FDISs are recommended to adapt feature extraction and feature representation scheme to establish an expression of the original data with its spatial-temporal characteristics and AI tasks to support it.

8.5 Data transmission

FDISs are recommended to implement the transformation of data from its original form to another, and meanwhile implementing its inverse process, i.e., expressing the original data in a lossy or lossless manner by using fewer data.

8.6 Feature analysis

FDISs are recommended to perform the AI task base on features with respect to the original data with comparable performance.

FDISs are recommended to satisfy the performance requirements of the applications. The recommended performance of AI tasks can be found in Appendix II.

8.7 Management

FDISs are recommended to implement fundamental functions such as user management, authority management and system logging.

9 Framework of feature-based intelligent systems

FDISs are composed of eight functional domains: application domain, development domain, algorithm domain, service domain, data domain, device domain, management domain and security domain, as illustrated in Figure 9-1.

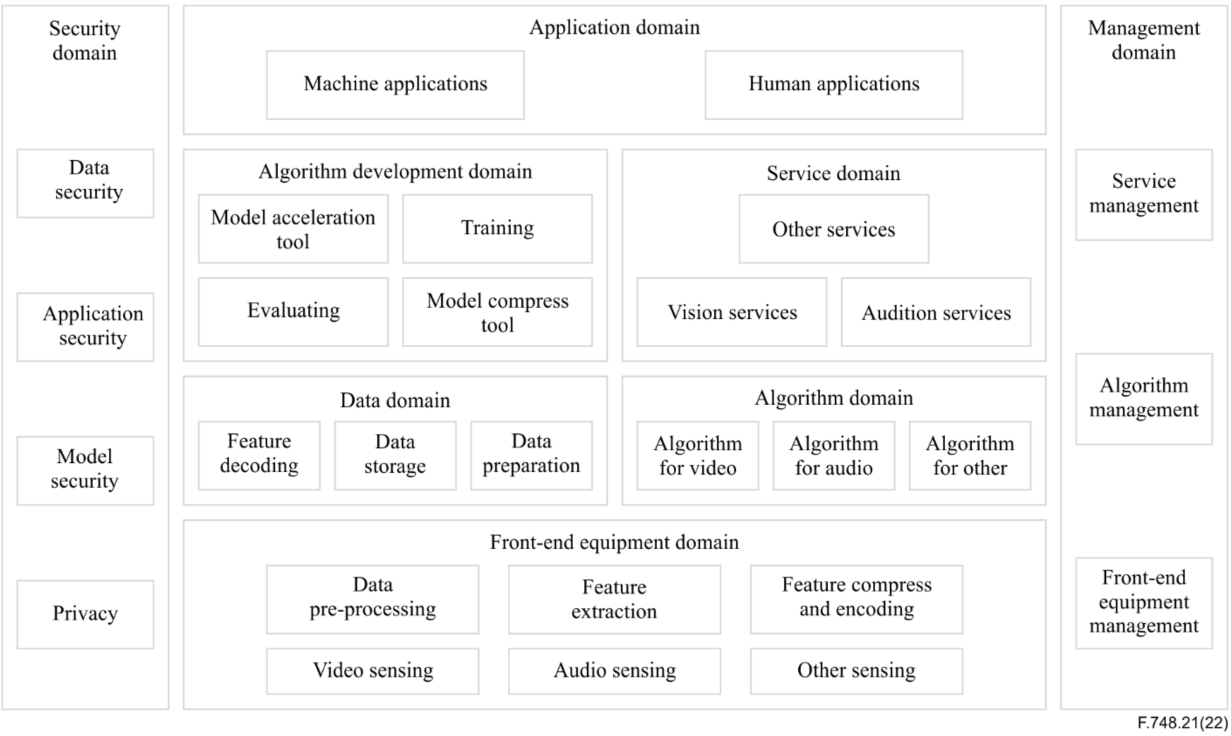

Figure 9-1 – Framework of feature-based distributed intelligent systems

9.1 Front-end equipment domain

The front-end equipment domain consists of data acquisition capabilities for generating different categories of feature to the data domain for further processing. For effective transmission and usage, the data should be pre-processed and compressed. The front-end equipment domain provides data pre-processing, feature extraction, feature compression and encoding, video sensing, audio sensing, and so on.

FDISs are recommended to support capturing different kinds of multimedia, such as video, audio and point cloud, by using sensors.

FDISs are recommended to generate different a category of feature by choosing a different algorithm.

FDISs are recommended to use a lower volume of data to represent the original data or feature in a lossy or lossless manner.

FDISs are recommended to support updating feature extraction and coding algorithm.

FDISs are recommended to support the pre-processing of acquired data, such as de-noising and image enhancement.

FDISs are recommended to guarantee the performance of AI tasks such as average precision or accuracy.

9.2 Data domain

In this domain, the transmitted data is decoded and transformed from a compressed bitstream into a reconstructed feature or reconstructed original data. It could be the inverse process of the data encoding procedure or the transcoding. The data domain provides decoding, data storage and data preparation.

FDISs are recommended to decode the transmitted data stream for future processing.

FDISs are recommended to support parallel scheduling and intelligent flow control to ensure that massive bitstreams can be processed timely.

FDISs are recommended to support storage of unstructured data and structured feature data after analysis, and provide management and control capabilities for supported feature-based tasks.

FDISs are recommended to support data decoding, data augmentation, data quality checking, data cleaning and data annotation.

FDISs are recommended to be able to handle noises in the sample data.

9.3 Algorithm domain

Feature-based intelligent applications are based on the analysis of multimedia data. Different applications and services need to establish different analysis models and match different algorithms. The algorithm domain implements the model to perform AI tasks. The algorithm domain provides various algorithms for video, audio, cloud point, and so on, and supports common deep learning neural network algorithm models such as residual neural networks (ResNet) and You Only Look Once (YOLO), and supports third-party algorithm libraries to meet different business needs.

FDISs are recommended to support multialgorithm frameworks, such as VSSNet, ResNet and YOLO.

FDISs are recommended to support various major tasks of image/video analysis, e.g., object detection and natural language processing (e.g., speech recognition).

FDISs are recommended to support plug-in third party algorithms.

9.4 Service domain

The service domain provides intelligent services based on algorithms such as the object classification service, quality verification service and guide service. FDISs customize a batch of analysis task templates for feature-based intelligent application scenarios, such as object classification service templates and object detection service templates. When customers use it, they only need to set the template parameters to create an analysis task, which will be analysed and processed by the FDIS.

FDISs are recommended to provide video analysis, image analysis and audio analysis capabilities.

FDISs are recommended to provide cloud point analysis.

9.5 Application domain

The application domain consists of human applications and machine applications.

After the feature processing, the features have been restored and usually passed through a certain deep learning model, in order to generate AI results within an acceptable performance loss for machine applications. Another pathway is to prepare the restored data/feature for human vision tasks. The metrics for AI task performance include precision, recall, accuracy, WER and SER, while the metrics regarding human sense include SSIM, PSNR and PESQ.

FDISs are recommended to support human applications and/or machine applications in given scenarios.

FDISs are recommended to meet the performance of the tasks supporting human applications and/or machine applications.

9.6 Development domain

In this domain, a set tools suite will be provided for feature-based model development such as model acceleration, model compression and model algorithm visualization including labelling, training, modelling and evaluation.

FDISs are recommended to support labelling the training dataset.

FDISs are recommended to support optimizing the model.

FDISs are recommended to support model compression.

FDISs are recommended to support model acceleration.

FDISs are recommended to support the visualization of model development.

FDISs are recommended to support the model compiling.

FDISs are recommended to support the model performance evaluation.

9.7 Management domain

The management domain provides front-end equipment management, algorithm management, service management, data management and application management.

FDISs are recommended to manage and log the status and capabilities of the front-end equipment, algorithms, services, data and applications.

9.7.1 Front-end equipment management

FDISs are recommended to implement fundamental functions such as user management, authorization management, system logging and security guarantees.

FDISs are recommended to support remote control, maintenance, update, configuration and diagnosis of front-end equipment.

FDISs are recommended to support programmable control on front-end equipment.

FDISs are recommended to support configuration on front-end equipment sampling information, such as frequency and resolution.

9.7.2 Algorithm management

FDISs are recommended to support the management of different algorithms, such as algorithm scheduling, invoke, licence and lifecycle management.

9.7.3 Service management

FDISs are recommended to support analysis task management, data source management and storage management.

9.8 Security domain

FIDSs are recommended to achieve acceptable performance on both the front-end and the server-end, while guaranteeing data integrity, data access authority, personally identifiable information (PII), security and effectiveness. As a result, the systems are recommended to be designed with an endogenous safety scheme, together with management. Moreover, user access control is usually implemented through a registration mechanism in these systems.

9.8.1 Data security

FDISs are recommended to guarantee data security, application security, platform security, access security and equipment security.

FDISs are recommended to guarantee data integrity and data security during the transmission between different modules within the system, which includes data acquisition, storage, analysis, processing, transmission and display.

FDISs are recommended to support measures, such as access control, encryption and token to achieve data confidentiality.

FDISs are recommended to support measures, such as user access controls, version controls and verification mechanisms, to achieve data integrity.

FDISs are recommended to support measures, such as backup, recovery and firewall to achieve data availability.

FDISs are recommended to prevent malicious data being used during the training procedure.

9.8.2 Model security

FDISs are recommended to guarantee the security of the deep model in case of unexpected attacks. To prevent attackers from acquiring user PII by repeatedly querying the same model, model access verification is recommended.

FDISs are recommended to ensure the model encryption by methods such as model distillation. The system administrator is recommended to ensure the broad coverage of the training data to enhance model robustness.

9.8.3 Application security

With respect to hardware and software, it is recommended to ensure any bug in applications, models, platforms and chipsets can be repaired in a prompt response.

FDISs are recommended to handle unexpected events correctly.

FDISs are recommended to check authentication and authorization.

FDISs are recommended to support countermeasures, such as encryption and decryption programs, anti-virus programs, spyware detection and deletion programs, to achieve application security.

9.8.4 PII protection

For output data, the identity of the individual is required to be removed or masked from the reconstructed data so it cannot be detected.

Data acquisition equipment such as video cameras and sensors is recommended to follow PII protection rules.

Appendix I

Use cases of feature-based intelligent applications

(This appendix does not form an integral part of this Recommendation.)

This appendix shows examples. They are not requirements.

I.1 Smart city

With the rise of IoT, many tasks that were usually done by workers or police officers can now be automated by an intelligent system, e.g., detection of illegal parking, detection of littering and traffic diversion. The above tasks often utilize intelligent systems to perform a list of sub-tasks such as object detection and object tracking. Besides, there is a high degree of interconnectivity between different node sensors and devices. It is important for these devices to communicate with each other to optimize and efficiently solve tasks. Therefore, the efficiency of data transmission can be achieved by applying the proposed feature-based framework.

I.2 ITS

In intelligent transport systems (ITSs), cars may need to communicate between each other and other sensors in order to perform different tasks such as autonomous driving, lane detection and pedestrian detection. Sensors in the infrastructure may communicate features towards different vehicles, which then use these features to perform intelligent tasks.

I.3 Content rating

Due to the generation of a huge quantity of video/image content, various forms of media including conventional broadcasting and personal broadcasting are overflowing. Protecting certain groups of people (e.g., people under 18) from inappropriate content is becoming a big issue. Meanwhile, the traditional manual review is time-consuming and labour-intensive. Thus, smart systems are being developed for film rating, inappropriate action detection and auto-mosaic. Intelligent tasks such as action recognition and object detection are performed during the process.

I.4 Smart manufacturing

With the rapid development of intelligent industry, the degree of automation production has been enhanced, making working environments which are unsuitable for manual work possible, and large-scale, continuous production a reality. Production efficiency and accuracy are greatly improved. Besides, interoperability is recommended for devices to post-process the features to perform multiple tasks to ease the job for humans: malfunction detection, machine component detection, fire detection, hazardous zone monitoring, and so on.

I.5 Smart agriculture

The rapid growth of machine vision ensures a high degree of automation in the agriculture industry. Various sensors and cameras are deployed to monitor the condition of crops or to track livestock. The feature-based framework can be applied to integrate the intelligent system for various tasks.

Appendix II

Performance of intelligent tasks

(This appendix does not form an integral part of this Recommendation.)

This appendix shows examples. They are not requirements.

II.1 Smart city

II.1.1 Object detection

In a feature-based distributed intelligent system, detection accuracy is recommended to achieve 85% and above, and processing speed differs according to specific algorithms. The algorithm processing time is recommended to be less than 200 ms/frame. The maximum framerate of the FDIS is recommended to be more than 25 fps. The delay is recommended to be within 2 s. The bitrate for 1080 P video is recommended to be less than 6 Mbit/s.

II.1.2 Image classification

In a feature-based distributed intelligent system, the classification accuracy is recommended to achieve 85% and above. The maximum framerate of the FDIS is recommended to be more than 25 fps. The delay is recommended to be within 2 s. The bitrate for 1080 P video is recommended to be less than 6 Mbit/s.

II.1.3 Object tracking

In a feature-based distributed intelligent system, the tracking accuracy is recommended to achieve 85% and above. The maximum framerate of the FDIS is recommended to be more than 25 fps. The delay is recommended to be within 2 s. The bitrate for 1080 P video is recommended to be less than 6 Mbit/s.

II.1.4 Action recognition

In a feature-based distributed intelligent system, the recognition accuracy is recommended to achieve 80% and above. The maximum framerate of a FDIS is recommended to be more than 25 fps. The delay is recommended to be within 2 s. The bitrate for 1080 P video is recommended to be less than 6 Mbit/s.

II.2 Smart manufacturing

In a feature-based distributed intelligent system, the detection accuracy is recommended to achieve 85% and above. The real-time task is recommended to be supported.

There are four main scenarios in smart manufacturing, namely, real-time control and application based on smart cameras, non-real-time application based on smart cameras, real-time application based on non-smart cameras, and application schemes based on industry Internet protocol cameras (IPCs).

The first scenario transfers a small amount of data, usually less than 100 bytes, through TCP/IP or user datagram protocol (UDP). The second scenario is time delay-insensitive, such as non-real-time backup, monitoring commands and documents. Such data is usually transferred through TCP/IP or file transfer protocol (FTP), and the camera usually uploads a video file of 5 MB. The third scenario is the mainstream application scheme nowadays, which contains GigE Vision and USB. The fourth scenario uses ITU-T H.264 to perform encoding. Streams are transmitted through the Real Time Streaming Protocol RTSP or Real Time Transport Protocol (RTP). As a comparison, the typical bitrate of 1080p@30fps video is 4 M, the typical bitrate of 2K@30fps is around 6 M, and when it comes to 4K@30fps, the bitrate increases to around 16 M.

As to the processing time consumed, in most industrial applications, time constraints differ from scenarios. For instance, in mobile phone surface lesion detection, the acceptable vision processing time is 500 ms ~ 1 s. In SPI/AIO (Solder Paste Inspection/Automatic Organic Inspection), the acceptable vision processing time is 200~300 ms. In solder joint inspection, as the cycle time is significantly longer, the acceptable vision processing time can reach 1~15 s.

Appendix III

Relation between tasks and features

(This appendix does not form an integral part of this Recommendation.)

In a FDIS, the features extracted and transmitted will be selected and configured according to the combination of AI tasks supporting intelligent applications. The mapping of the AI task and features supporting the given AI task is shown in Table III.1.

Table III.1 – Relation between AI tasks and features

AI task	Feature
Object detection	Bounding box, edge, colour, corner...
Classification	Class labels, latent domain features, etc.
Search	Feature descriptions (SIFT, SURF, HOG, BRIEF, etc.)
Object segmentation	Region features (RoI, segmentation mask), etc.
Object tracking	Bounding box, motion features

Bibliography

[b-ITU-T K.142] Recommendation ITU-T K.142 (2019), *Lightning protection and earthing of video surveillance systems*.

[b-ISO/IEC 29100] ISO/IEC 29100:2011, *Information technology – Security techniques – Privacy framework.*

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	**Non-telephone telecommunication services**
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Published in Switzerland
Geneva, 2023