



Information technology—Computer graphics and image processing—Image Processing and Interchange (IPI)—Functional specification

Part 1: Common architecture for imaging



This Australian Standard® was prepared by Committee IT-031, Computer Modelling and Simulation. It was approved on behalf of the Council of Standards Australia on 20 April 2015. This Standard was published on 12 May 2015.

The following are represented on Committee IT-031:

- ANZLIC—The Spatial Information Council
 - Department of Defence (Australia)
 - Simulation Australia
-

This Standard was issued in draft form for comment as DR AS ISO/IEC 12087.1:2015.

Standards Australia wishes to acknowledge the participation of the expert individuals that contributed to the development of this Standard through their representation on the Committee and through the public comment period.

Keeping Standards up-to-date

Australian Standards® are living documents that reflect progress in science, technology and systems. To maintain their currency, all Standards are periodically reviewed, and new editions are published. Between editions, amendments may be issued.

Standards may also be withdrawn. It is important that readers assure themselves they are using a current Standard, which should include any amendments that may have been published since the Standard was published.

Detailed information about Australian Standards, drafts, amendments and new projects can be found by visiting www.standards.org.au

Standards Australia welcomes suggestions for improvements, and encourages readers to notify us immediately of any apparent inaccuracies or ambiguities. Contact us via email at mail@standards.org.au, or write to Standards Australia, GPO Box 476, Sydney, NSW 2001.

Australian Standard[®]

**Information technology—Computer
graphics and image processing—Image
Processing and Interchange (IPI)—
Functional specification**

**Part 1: Common architecture for
imaging**

First published as AS ISO/IEC 12087.1:2015.

COPYRIGHT

© Standards Australia Limited

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher, unless otherwise permitted under the Copyright Act 1968.

Published by SAI Global Limited under licence from Standards Australia Limited, GPO Box 476, Sydney, NSW 2001, Australia

ISBN 978 1 76035 044 4

PREFACE

This Standard was prepared by the Standards Australia Committee IT-031, Computer Modelling and Simulation.

The objective of this Standard is to provide an architectural model for the representation and manipulation of images in a digital form. Based on this model, it defines an application program interface and an image interchange format. It is applicable to all areas that involve the processing, manipulation, or transfer of image data.

This Standard is identical with, and has been reproduced from ISO/IEC 12087-1:1995, *Information technology—Computer graphics and image processing—Image Processing and Interchange (IPI)—Functional specification, Part 1: Common architecture for imaging*.

As this Standard is reproduced from an International Standard, the following applies:

- (a) In the source text ‘this part of ISO/IEC 12087’ should read ‘this Australian Standard’.
- (b) A full point substitutes for a comma when referring to a decimal marker.

References to International Standards should be replaced by references to Australian or Australian/New Zealand Standards, as follows:

<i>Reference to International Standard</i>		<i>Australian Standard</i>	
ISO/IEC		AS ISO/IEC	
9973	Information technology—Computer graphics, image processing and environmental data representation—Procedures for registration of items	9973	Information technology—Computer graphics, image processing and environmental data representation—Procedures for registration of items

Only normative references that have been adopted as Australian or Australian/New Zealand Standards have been listed.

The terms ‘normative’ and ‘informative’ have been used in this Standard to define the application of the annexes to which they apply. A ‘normative’ annex is an integral part of a Standard, whereas an ‘informative’ annex is only for information and guidance.

CONTENTS

1	Scope	1
2	Normative References	3
3	Definitions and abbreviations	4
3.1	Definitions	4
3.2	Abbreviations	5
3.3	Diagrammatic Conventions	5
4	The IPI architecture	7
4.1	IPI imaging architecture	7
4.1.1	IPI imaging model	7
4.1.2	IPI operator processing model	8
4.2	IPI basic data types	9
4.2.1	IPI elementary data types	10
4.2.2	IPI compound data types	10
4.3	IPI image data types	11
4.3.1	IPI derived elementary image data types	11
4.3.2	IPI derived compound image data types	11
4.3.3	IPI derived image attributes	13
4.4	IPI derived non-image data types	14
4.4.1	IPI derived image annotation data types	14
4.4.2	IPI derived image-related non-image data types	14
5	IPI-PIKS architecture	16
5.1	IPI-PIKS imaging model	16
5.1.1	IPI-PIKS neighbourhood control	16
5.1.2	IPI-PIKS image control	17
5.2	IPI-PIKS system control	17
5.2.1	Data object management	17
5.2.2	Operational synchronicity	18
5.2.3	Element chaining	18
5.2.4	Error management	18
5.3	IPI-PIKS basic data types	18
5.3.1	IPI-PIKS elementary data types	18
5.3.2	IPI-PIKS compound data types	19
5.4	IPI-PIKS derived image data descriptions	22
5.4.1	IPI-PIKS derived data types	22
5.4.2	IPI-PIKS compound image data types	22
5.4.3	Composite images	23
5.4.4	IPI-PIKS image object attributes	24
5.5	IPI-PIKS derived non-image data structures	26
5.6	IPI-PIKS data pragmata	34

	<i>Page</i>
6 IPI-IIF-specific architecture	35
6.1 IPI-IIF imaging model	35
6.2 IPI-IIF basic data types	35
6.3 IPI-IIF derived data types	36
6.3.1 IPI-IIF derived image data types	36
6.3.2 IPI-IIF image attributes	36
6.3.3 IPI-IIF derived non-image data types	37
6.3.3.1 IPI-IIF image annotation data types	37
6.3.3.2 IPI-IIF image-related non-image data types	37
7 Relationship between IPI-PIKS and IPI-IIF	41
8 Conformance	42
8.1 Conformance of functionality	42
8.2 Conformance of accuracy and precision	42
8.3 Extensions	42
8.4 Conformance profiles	43
8.4.1 Types of profile	43
8.4.2 Application profile registration	44
8.4.3 Profiles defined by IPI	44
Annexes	46
A Structured image data types	46
B Structure codes	48
C The representation of colour	49
D Language-Independent Data Types	56
D.1 Bit	56
D.2 Boolean	56
D.3 Character	57
D.4 Complex	58
D.5 Enumerated	58
D.6 Null	59
D.7 Integer	59
D.8 Real	60
D.9 State	61
D.10 Array	61
D.11 Choice	62
D.12 List	63
D.13 Pointer	64
D.14 Range	65
D.15 Record	65
D.16 Set	66
D.17 Character String	67
D.18 Table	68
E Bibliography	69

List of figures

1	Relationship of the parts of ISO/IEC 12087	2
2	Diagrammatic conventions	6
3	Interfaces between application program, IPI-PIKS, and IPI-IIF	8
4	Fundamental operator processing model	9
5	The operator model used by IPI-PIKS	16
6	Relationship Between a Physical Volume and IPI-PIKS Horizontal, Vertical, and Depth Coordinates	23
7	Aggregation of Image References into a List	24
8	Colour Systems and Representations Used by IPI	50

List of tables

1	Codes for the externally-visible representations of IPI-PIKS-specific data types	21
2	Dimensions of an IPI-PIKS Data Object	22
3	IPI-IIF profiles that correspond to IPI-PIKS profiles	44
4	IPI-PIKS profiles that correspond to IPI-IIF profiles	45
5	<i>XYZ</i> tristimulus values for the white points of common illuminance	51
6	Supported types of colour representation, and their attributes	53
7	Standardized parameterisations of colours	53
8	Parameter values for the standardized colour representations (non-normative)	54
9	Mappings Between Colours and Image Channels	55

INTRODUCTION

The processing of images is a requirement of many application areas of information processing. Early work in these areas led to the development of many application program interfaces and a large number of image representations for interchange. The purpose of ISO/IEC 12087 is to provide an application program interface and an image interchange representation in order to increase the portability of application software.

ISO/IEC 12087 provides an architectural model for the representation and manipulation of images in a digital form. Based on this model, it defines an application program interface and an image interchange format. It is applicable to all application areas that involve the processing, manipulation, or transfer of image data.

ISO/IEC 12087 includes notes and exemplary material. Such material is non-normative: it is included solely to aid understanding and does not form part of ISO/IEC 12087.

ISO/IEC 12087 initially comprises three parts:

- 1 *Common architecture for imaging*, which describes the common architectural material on which the entire Standard is based;
- 2 *Programmer's imaging kernel system application program interface*, which defines processing operations to be carried out on image data;
- 3 *Image Interchange Facility (IIF)*, which defines how images may be interchanged between application programs.

Information may be interchanged between the application program, Programmer's Imaging Kernel System (IPI-PIKS), and Image Interchange Facility (IPI-IIF) (see figure). Data paths between all three components are standardized in ISO/IEC 12087, as indicated by the solid lines; however, it is also permitted that implementations may use private, implementation-dependent data paths, shown by dashed lines; such data paths are outside the scope of ISO/IEC 12087.

There are a great many types of application that involve the use of images. The Computer Graphics Reference Model [ISO 11072] identifies six main function classes (see figure 0.1):

- image analysis — transformation of digital images to image and non-image data; this encompasses basic functions such as histogram generation, mean value determination, image classification, *etc.*, but does not include image understanding using artificial intelligence techniques.
- image interpretation — the process of inferring symbolic scene descriptions from image data.
- image presentation — transformation of image data to a form suitable for an observer; *e.g.*, via video monitors, printers, film recorders, *etc.*
- image processing — transformation of digital images to digital images; *e.g.*, grey value contrast enhancement, edge detection, *etc.*
- image sensing — transformation of real-world information to digital images; *e.g.*, via cameras, optical scanners, *etc.*
- image synthesis — transformation of non-image data to image data; this encompasses functions such as the rendering of lines, creation of test images, simulation of sensor functions, letters of graphical text and symbols, *etc.*

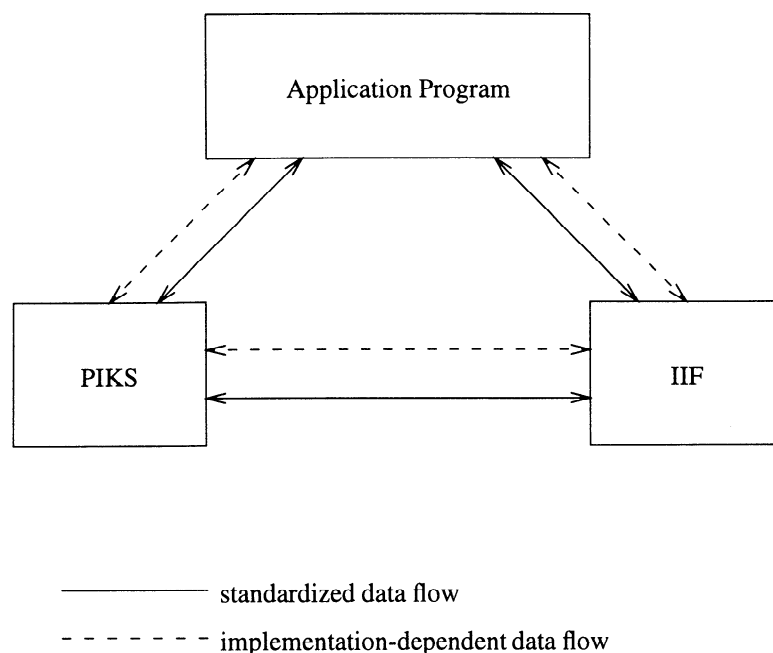


Figure 0.1 — Data flow between the application program, IPI-PIKS, and IPI-IIF

As figure indicates, all these function classes involve the manipulation of a digital image; some function classes also require information that is related to the data contained in the digital image but is itself non-image in nature. This *image-related* information is essential to many of the common operations performed on digital images and is therefore also described by ISO/IEC 12087.

ISO/IEC 12087 is also concerned with *image interchange*, the interchange of digital images among imaging applications; this serves for the communication of image data and related non-image data among imaging applications.

The term ‘digital image’ used in [ISO 11072] is synonymous with the term ‘image’ as used in ISO/IEC 12087. It is important to realize the distinction between ‘image’ (or ‘digital image’) as used in ISO/IEC 12087 and the term ‘image’ as it may be used colloquially: in ISO/IEC 12087, ‘image’ (or ‘digital image’) refers to a particular representation of image data within a computer system. An image may not be viewed directly. To view an image, an explicit presentation step is involved, as figure indicates. Image data that are in a form suitable for viewing by an observer are termed ‘presentable’ image data in ISO/IEC 12087.

NOTE 1 Some application areas, which might loosely be termed “image understanding,” utilize data derived from an image by means of some analysis; such applications are therefore omitted from this ISO/IEC 12087. However, ISO/IEC 12087 may be used by such applications.

This part of ISO/IEC 12087 fulfills the following purposes:

- a) It provides an overview of ISO/IEC 12087;
- b) It defines a Common Architecture for Imaging, an abstract architectural model for the representation

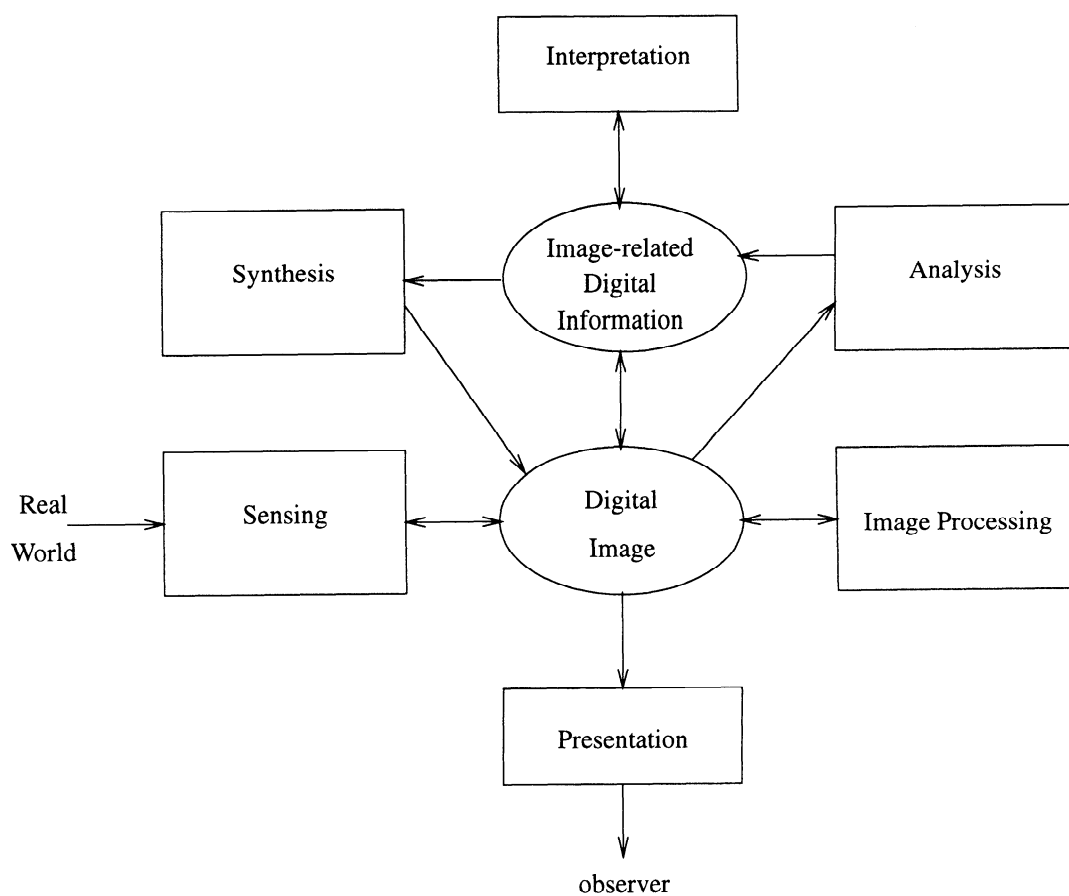


Figure 0.2 — Classes of operations on images

and processing of image data. The purpose of this model is to define a common set of data types and a common image representation for use with all other parts of ISO/IEC 12087 and to provide a standardized framework upon which future imaging standards may be built, allowing simplified conversion of existing applications to the new standard.

- c) It defines rules to which conforming implementations shall adhere and the mechanism by which conformance is achieved.

AUSTRALIAN STANDARD

Information technology—Computer graphics and image processing—Image Processing and Interchange (IPI)—Functional specification**Part 1:**
Common architecture for imaging**1 Scope**

ISO/IEC 12087 is concerned with the manipulation, processing, and interchange of all types of digital images. The main purpose of this part is to define a generic, unifying imaging architecture to which other parts of ISO/IEC 12087 conform. This part of ISO/IEC 12087 also defines those “specializations” or “delineations” of the generic imaging architecture that are required to support IPI-PIKS and IPI-IIF.

The relationship of the different parts of ISO/IEC 12087 is shown in figure 1. This part of ISO/IEC 12087 describes material that applies throughout ISO/IEC 12087, including topics such as data types available for use in image data and image-related data, and a model for the processing of digital images by operators. These topics are presented in a general form, since it is intended that subsequent imaging standards will conform to the same architectural model.

Derived from this general description are more constrained descriptions of the same topics. The principal reason for this process of delineation is to restrict the range of data representations for IPI-PIKS and IPI-IIF, while simultaneously ensuring that IPI-IIF is capable of interchanging both IPI-PIKS data objects and objects that cannot be represented or manipulated within IPI-PIKS.

ISO/IEC 12087 permits multiple Application Program Interface (API)s to be developed, each of which must be

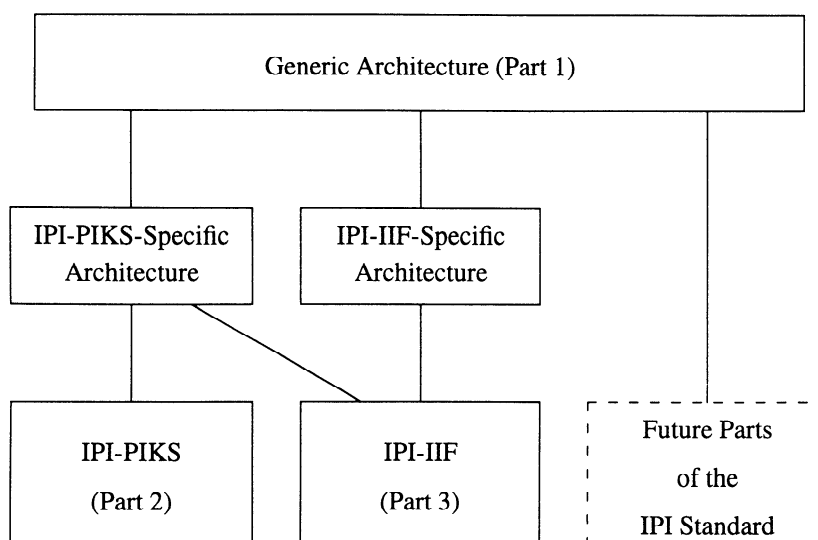


Figure 1 — Relationship of the parts of ISO/IEC 12087

based on specific delineations of the imaging model described herein. Each API will be specified in a separate part of ISO/IEC 12087. Any subsequent APIs developed as part of ISO/IEC 12087 must conform to the common architecture described in this document, and must be extensions of the APIs described in ISO/IEC 12087-2 and ISO/IEC 12087-3.

ISO/IEC 12087 is intended for use in a wide variety of environments where digital images are handled.

NOTE 2 Application areas that are addressed by Image Processing and Interchange (IPI) include: image manipulation; image enhancement; image analysis; and image transport. Application areas that are not addressed by IPI include: computer graphics; image understanding; multimedia; device control; and window systems.

ISO/IEC 12087 is intended to conform with other International Standards developed to handle digital images. Such standards include the JPEG [ISO/IEC 10918-1:1994], and MPEG [ISO/IEC 11172-1:1993] compression standards, Open Systems Interconnect [ISO/IEC 8824:1990], and Office Document Architecture [ISO/IEC 8613]. Those aspects of ISO/IEC 12087 that are concerned with the acquisition and display of digital images conform with the Computer Graphics Reference Model [ISO 11072]. Furthermore, annex B of [ISO 11072] describes how imaging fits within the general framework of that model. ISO/IEC 12087-3 uses Abstract Syntax Notation 1 [ISO/IEC 8824:1990] in the definition of the image interchange format.

ISO/IEC 12087 complies directly with all standards listed in clause 2.

2 Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this part of the IPI Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on ISO/IEC 12087 are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 646:1991, *Information technology, ISO 7-bit coded character set for information interchange*.

ISO/IEC 8613:1989, *Information processing — Text and office systems — Office Document Architecture (ODA) and interchange format*.

ISO/IEC 8824:1990, *Information technology — Open Systems Interconnection — Specification of Abstract Syntax Notation One ASN.1*.

ISO/IEC 8825:1990, *Information technology — Open Systems Interconnection — Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*.

ISO/IEC 9973:1994, *Information technology, Computer graphics and image processing — Procedures for registration of graphical items*.

ISO/IEC 10918-1:1994, *Information technology — Digital compression and coding of continuous-tone still images: Requirements and guidelines*.

ISO/IEC 11072:1992, *Information technology — Computer graphics — Computer Graphics Reference Model*.

ISO/IEC 11172-1:1993, *Information technology — Coding of moving pictures and associated audio for digital storage media up to about 1,5 Mbit/s — Part 1: Systems*.

CIE:1931, *Proceedings of the eighth session, Cambridge, England, 1931*. Bureau Centrale de la CIE, Paris.

CIE:1970, *International Lighting Vocabulary*. CIE publication no. 17, third edition, Bureau Centrale de la CIE.

CIE:1976, *CIE recommendations on uniform colour space, colour difference equations, psychometric colour terms*. Supplement no. 2 to CIE publication 15, Bureau Centrale de la CIE, Paris.

CCIR:1990a, *Report 476-1, Colorimetric standards in colour television*. Recommendations of the CCIR 1990, Annex to Volume XI – Part 1 – Broadcasting Service (Television), CCIR, Geneva.

CCIR:1990b, *Report 624-4, Characterization of television systems*. Recommendation of the CCIR 1990, Annex to Volume XI – Part 1 – Broadcasting Service (Television), CCIR, Geneva.

CCIR:1990c, *Recommendation 709, Basic parameter values of the HDTV standard for the studio and for international programme exchange*. Recommendations of the CCIR 1990, Volume XI – Part 1 – Broadcasting Service (Television), CCIR, Geneva.

EBU:1975, *EBU standard for chromaticity tolerances for studio monitors*. Tech. 3213-E, Brussels.

DeMarsh:1974, *Colorimetric standards in U. S. color television. A report by the subcommittee on systems colorimetry of the SMPTE television committee*. L. E. DeMarsh, Journal of the Society of Motion Picture and Television Engineers, vol. 83.

NTSC:1954, *NTSC signal specifications*. Proceedings of the IRE, January 1954.

3 Definitions and abbreviations

3.1 Definitions

For the purpose of ISO/IEC 12087, the following definitions apply.

3.1.1 digital image: Synonymous with image.

3.1.2 element: A collective noun used to describe mechanism, operator, tool, and utility.

3.1.3 image: A data structure that contains pixels and image-related data.

3.1.4 image data: Digital values corresponding to pixels.

3.1.5 image-related data: Information that provides context for the interpretation of an image.

3.1.6 mechanism: A software element that performs control and management tasks.

3.1.7 operator: A software element that performs manipulation of an image or set of images. Permitted operator types are image-to-image, image-to-non-image, and non-image-to-non-image.

NOTE 3 Operators may be based upon other operators — in which case they are said to be high-level operators — or may not, in which case they are said to be primitive. All classes of IPI-PIKS operators, tools, and utilities may be invoked from an application program, but only operators and utilities interface directly to the underlying system.

3.1.8 primitive operator: One of a small set of low-level operators that is fundamental to image processing.

3.1.9 pixel: An abbreviation for the term ‘picture element.’

NOTE 4 In some application areas, the term ‘pixel’ commonly refers to 2D images only, with separate terms (*e.g.*, ‘voxel’) being used to describe an element of a higher-dimensional image. In this Standard, the single term ‘pixel’ is used to refer an element of an image of any dimensionality.

3.1.10 picture element: A digital representation of one or more quantities at a particular location within an image. A picture element may have an associated physical metric, size, or interval.

3.1.11 tool: A software element that creates data objects to be used by operators. Examples of tools are test image generation and the generation of mathematical functions.

3.1.12 utility: A software element that performs basic image manipulation operations (*e.g.*, image import, access, and memory management).

3.2 Abbreviations

API	Application Program Interface
CAI	Common Architecture for Imaging
CCIR	Comité Consultatif International des Radiocommunications
CGI	Computer Graphics Interface
CGM	Computer Graphics Metafile
CIE	Commission Internationale d'Eclairage
CMY	Cyan, Magenta, Yellow (colour model)
CMYK	Cyan, Magenta, Yellow, Black (colour model)
EBU	European Broadcasting Union
GKS	Graphical Kernel System
JPEG	Joint Photographic Experts Group
IHS	Intensity, Hue, Saturation (colour representation)
IIF	Image Interchange Facility
MPEG	Motion Picture Experts Group
NMR	Nuclear Magnetic Resonance
NTSC	National Television Standards Committee
ODA	Office Document Architecture
ODIF	Office Document Interchange Format
PET	Positron Emission Tomography
PHIGS	Programmer's Hierarchical Interactive Graphics System
PIKS	Programmer's Imaging Kernel System
RGB	Red, Green, Blue (colour representation)
ROI	region of interest (defined in clause 4.4.2)
SMPTE	Society of Motion Picture and Television Engineers

3.3 Diagrammatic Conventions

Diagrams are used throughout ISO/IEC 12087 to support textual descriptions. The graphical items that comprise these diagrams have a consistent meaning, as shown in figure 2, unless indicated to the contrary.

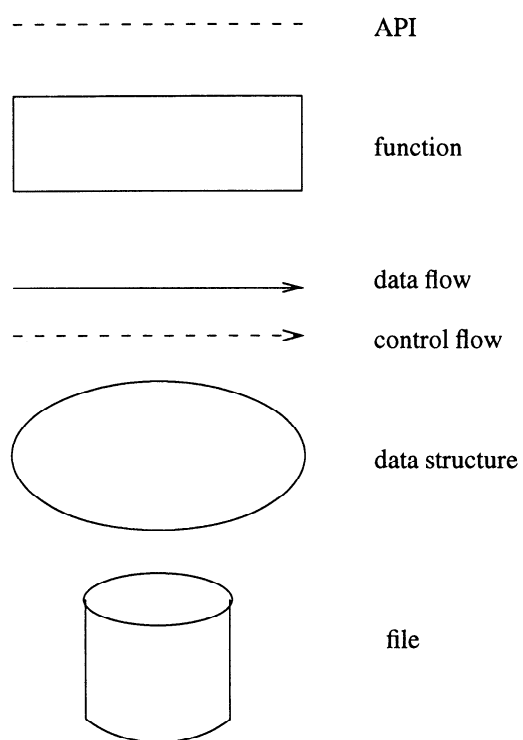


Figure 2 — Diagrammatic conventions

4 The IPI architecture

4.1 IPI imaging architecture

In general, imaging systems deal with several types of data falling in the general categories of image data and non-image data, all of which are normally related to physical phenomena. Examples of these categories are:

image data — grey-scale images, bi-tonal images, colour images, multichannel satellite data, image sequences, sequences of stereo pair images, *etc.*;

non-image data — look-up tables, histograms, text, graphics, regions of interest, *etc.*;

real world information — continuous physical variables such as light intensity and wavelength, pressure, temperature, continuous chemical variables, *etc.*

For the purpose of ISO/IEC 12087, three categories of data types are defined:

basic data types — these are the data types that must be supported by the system upon which ISO/IEC 12087 is implemented; the category includes both ‘elementary’ data types (*e.g.*, Boolean, integer) and ‘derived’ (structured) data types (*e.g.*, array, record).

image data types — these are derived data types that represent image data for the purposes of processing and interchange.

non-image data types — these are derived data types that represent image-related, but non-image, data (*e.g.*, region of interest).

These data types are defined in general terms in this clause. Note that data types are regarded as identical if they are of identical structure with respect to the categories defined in this clause.

4.1.1 IPI imaging model

Figure 3 shows the relationship between the parts of an IPI implementation. In particular,

- IPI-PIKS and IPI-IIF are self-contained parts of ISO/IEC 12087, and may be used either independently or in conjunction by an application program;
- an application program may apply imaging functions via IPI-PIKS;
- an application program may import and export images directly via IPI-IIF;
- any interpretation of data types other than those of the three abovementioned categories must be performed by the application program;
- an application program controls how and to what extent IPI-PIKS and IPI-IIF are used;
- an application program may use other standards for information processing (*e.g.*, GKS, CGI, PHIGS) or for data interchange (*e.g.*, CGM, ODA/ODIF);

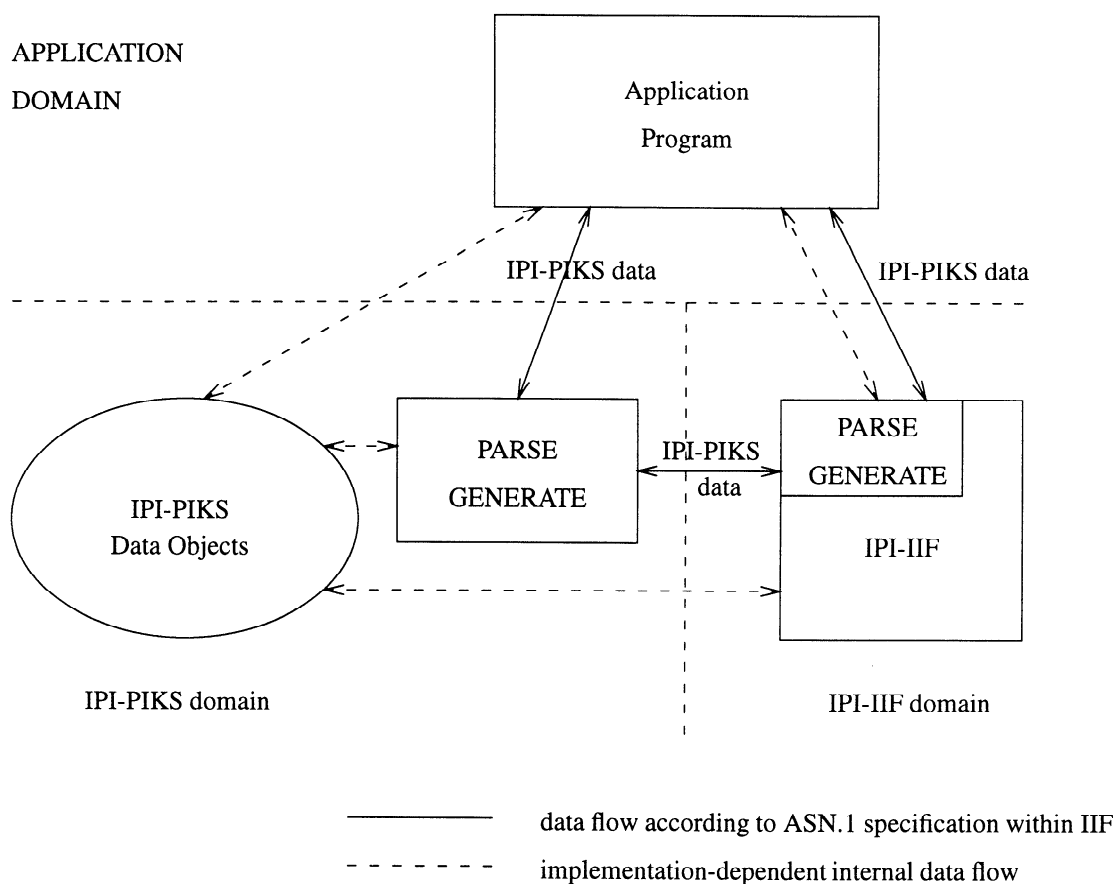


Figure 3 — Interfaces between application program, IPI-PIKS, and IPI-IIF

- an application program may perform user interaction as well as the display and presentation of images to the user;
- an application program is responsible for the consistent use of different standards.

Note that the parsing and generation of an IPI-IIF data stream is capable of being performed both within IPI-IIF itself, or in IPI-PIKS when importing or exporting images from an IPI-IIF data stream. The parsing and generation capability within IPI-PIKS is a subset of that available in IPI-IIF. See clause 7 for further details.

4.1.2 IPI operator processing model

The fundamental operator model for IPI contains three elements, which are summarized in figure 4. Note that this model does not apply to IPI-PIKS Tools, Utilities, and Mechanisms. The three elements are:

an operator — which performs one of three possible conversions:

- source image data to destination image data;

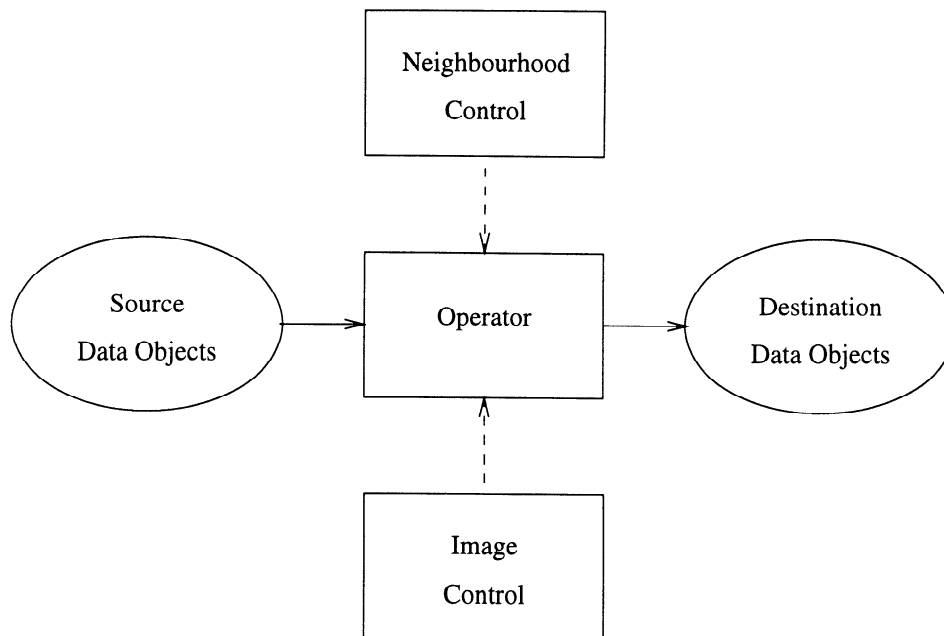


Figure 4 — Fundamental operator processing model

- source image data to destination non-image data;
- source non-image data to destination non-image data.

an image control element — which selects source image pixels to be processed, controls processing options, and selects which processed source pixels are to be recorded in the destination image.

a neighbourhood control element — a means of describing a region around a pixel, not bound to any one pixel, that identifies source image pixels to be processed and selects processing options, subject to the image control.

NOTE 5 The image control and neighbourhood control elements serve distinct purposes. Image control is used to identify that region of the image within which processing will occur: it is necessarily bound to the image. An example of this is the region of interest, discussed in this clause and elsewhere. However, neighbourhood control is used to specify a small region which is not bound to the image; an example of this is an impulse response array used by convolution operators.

4.2 IPI basic data types

For the purpose of ISO/IEC 12087, two categories of basic data types exist: elementary data types and compound data types. Compound data types are constructed from elementary data types via construction mechanisms (see clause 4.2.2). Other data types, such as image data types and non-image data types, shall refer to basic data types.

The basic data types directly comply with [1], which defines both simple and structured data types by the properties that they exhibit. Detailed information regarding the definitions of the basic data types and the basic operations that are supported are defined in [1]. Data types that are required by IPI but not defined in [1] are fully defined below.

The lists of basic operations for the data types defined below are not intended to define strictly minimal and complete sets of operations in a mathematical sense, but rather indicate important basic operations that are necessary to cover practical applications. All the operations listed below are not necessarily available in IPI-PIKS; they are included to limit any future APIs.

4.2.1 IPI elementary data types

The elementary data types supported by IPI are as follows:

- bit
- Boolean
- character according to [ISO/IEC 646:1991]
- complex number
- enumerated
- null
- integer number
- real number
- state

These types are defined in annex 8.4.3. Each of these generic elementary data types may be delineated into one or more specific data types elsewhere in ISO/IEC 12087, which should then be treated as separate data types.

4.2.2 IPI compound data types

A compound data type is a data type that is constructed by one of the following mechanisms, also defined in annex 8.4.3, from basic data types:

- $N_1 \times \cdots \times N_n$ array
- character string
- choice
- list
- pointer
- range
- record
- set
- table

Images are comprised of a number of pixels. The pixel values of an image must be constructed from the basic data types listed in clause 4.2. The construction mechanisms used to combine pixels into images are the compound data types described in this clause. Similarly, all image-related and non-image data described by ISO/IEC 12087 are constructed from the same elementary data types via the same construction mechanisms. Consequently, all these 'structured' or 'constructed' data types are referred to as *derived* data types for

the purposes of ISO/IEC 12087.

4.3 IPI image data types

This clause considers the nature of the image data types that are represented and manipulated by ISO/IEC 12087. All these data types are derived from the elementary data types of clause 4.2.1 using the construction mechanisms described in clause 4.2.2. Two categories of image data type may be derived by this method:

- elementary image data type — or **fundamental image data type** in which the pixel values are of an elementary data type or a compound data type whose elements are themselves of basic data type;
- compound image data type — in which the construction mechanisms of clause 4.2.2 are used to combine images, which may themselves be either elementary or compound.

Both these categories are considered below.

4.3.1 IPI derived elementary image data types

An elementary image data type describes an image that is built from pixels whose values are referenced to a single spatial position. An elementary image may be constructed from elementary data types using any of the compound data types of described earlier in this clause.

Note that IPI-PIKS is capable of representing and processing only particular categories of elementary image data types internally, although its compound image aggregation facility (clause 5.4.3) permits the application program to represent a wider range of structures.

4.3.2 IPI derived compound image data types

The central idea behind compound images in ISO/IEC 12087 is to provide structured access to large sets of related images. Heterogeneous image and non-image data collections can be combined into structures of arbitrary complexity. Pixel values may also be of arbitrary complexity: they could be elementary data types for storing scalar amplitude information; but they could also be compound data types for associating more comprehensive and structured information with every pixel. Indeed, the compound image data types deliberately provide more than one way for structuring the same amount of data to optimally meet the demands of the application. Multi-spectral images, for example, depending on the access mechanisms needed by the application or the properties of the sensors, could be represented as:

- an elementary image whose pixel values are 1D arrays of integers representing the intensities of the respective channels;
- a 1D array of elementary images whose pixel values are integers representing the intensities in one of the frequency channels, if regularly-spaced frequency channels have to be accessed at random;
- a list of elementary images whose pixel values are integers representing the intensities in one of the frequency channels, if irregularly spaced frequency channels have to be accessed sequentially;

- a record of elementary images whose pixel values are integers representing the intensities in one of the frequency channels, if the frequency channels are of different spatial resolutions or have to be accessed by name.

In fact, a complete hierarchy of constructs can be achieved:

- individual pixels;
- 1, 2, 3, ..., n -dimensional arrays of pixels;
- images comprising arrays of pixels and additional information;
- compound images of arbitrary complexity comprising different image and non-image data types.

A compound image data type is a data type constructed by one of the following construction mechanisms from image data types:

$(N_1 \times \dots \times N_n)$ image array — an array of images of identical image data type that can be accessed individually by their indices $(i_1, \dots, i_n)$. Example operations: equal relation, setting and retrieving of individual components with random access. Examples of image arrays are identically-structured, local high-resolution images of regularly-spaced test patterns for the inspection of printed circuit boards or integrated circuits, regular spatio-temporal sequences of slice images through a beating heart, *etc.*

image list — an ordered image set. Individual images can be accessed only via successor/predecessor relations. Example operations: equal relation, is-empty, get-head, get-tail, append, cut, get-successor, get-predecessor, iterate-through-list. Examples of the data type are collections of related images that can be ordered according to their spatial, temporal, or frequency relation, *e.g.* stacks of Computerized Tomography or Magnetic Resonance images through a tumour in a human body, time sequences of a moving heart or moving objects in industrial scenes, images obtained at different wavelengths or energies, *e.g.* multi-spectral images, scattered X-ray images, spectroscopic sequences in Magnetic Resonance Imaging, *etc.*

image record — a collection of individually-named image data types and optional non-image data types. The record components may be of different type and are accessed via their names. Example operations: equal relation, append-component, remove-component, set-component, get-component. Examples of situations in which image records would prove useful include collections of images of an object that has been imaged with different sensors at different spatial, temporal, or frequency resolutions, *e.g.* images of increasing spatial resolution that form an image pyramid, medical images of the same patient, *etc.*

image set — a collection of images of identical image data type. The images cannot be accessed individually. It is possible only to check whether or not an image belongs to a set. Example operations: equal and subset relations, is-element-of, define-set-of, is-empty, apply-to-all-elements; set operations complement, union, intersection. Examples of image sets are sets of template images representing a certain class of objects, *e.g.* automobile parts, textures *etc.*, that are only used to check whether or not an image belongs to the class.

Note that image arrays, lists, and sets are analogous to compound data types but that image records are not, since they may contain both pixel and non-pixel data. The operations that may be performed on these compound data types conform with those described in [1].

4.3.3 IPI derived image attributes

Image attributes carry information on how to correctly interpret the associated image data; they are inherently bound to images. Hence, image attributes are relevant to both IPI-PIKS and IPI-IIF. Image attributes may be attached either to elementary images (clause 4.3.1) or to compound images (clause 4.3.2). Attributes that are not attached to pixel rasters but to higher-level structures of compound images are effective for all lower level components (*i.e.*, the inheritance of attributes is hierarchical). However, it is possible to overwrite inherited attributes via their basic operations. Example operations for image attributes are: equal relation, attach-attribute, remove-attribute, set-attribute, get-attribute.

This clause describes image attributes in a purely textual and behavioural manner; for details on the content and data types of image attributes supported by IPI-PIKS and IPI-IIF, see clause 5.4.4 and clause 6.3.2.

channel attribute — Many types of images can be thought of as being comprised of several ‘channels’ of data. The channel attribute provides information about the sensor models underlying these channels that is necessary for the analysis of channel information, *e.g.* the number of bits used for quantizing the original sensor signals, sensitivity and noise characteristics of the sensors, spectral behaviour of the sensors, *etc.*

For example, a medical application might combine an MR and a PET data set into a single IPI image; the channel attribute would then be used to identify the MR and PET channels and to describe the characteristics of the two devices that captured them. Similarly, colour images are generally digitized into three separate channels, the first representing the red pixels, the second the green pixels and the third the blue pixels, and the channel attribute could be used to describe this.

colour attribute — Since colour images form such an important class to the human observer, it is essential that sufficient information is carried with a colour image for it to be reproduced accurately. Hence, a *colour attribute* is associated with colour images. Note that a colour image has a single colour description, but a separate channel description for each channel. This colour description encompasses all entities dealing with the mixing of amplitude values of multiple channels in order to achieve accurate colour reproduction; this may consist of the colour representation involved (see annex 8.4.3), test colours for regions of the image, *etc.*

control attribute — The purpose of a control attribute is to convey information that may be needed for subsequent processing or presentation of an image. This may include, for example, the match-point and the identifier of a region of interest associated with an IPI-PIKS image.

free-form attribute — The internal structure and syntax of the free-form description attribute is not specified within ISO/IEC 12087. It is application-dependent, and is provided to allow application-specific descriptive information to be attached to the pixel data of an image. It is left to the application to decide on the actual way of storing and manipulating this information. No IPI-PIKS functionality is provided to manipulate the content of the free-form description; any processing of this description must be performed entirely by the application program.

One example for a free-form description which is defined in detail by an application-specific standard is the header information as defined by the ACR-NEMA standard for medical applications. It provides specific formats for describing the nature of the image, orientation of the image with respect to the human body, patient data, *etc.* Another example is the FITS format widely adopted in the observational astronomy community, which allows tables of physical quantities associated with the recording process to be transferred as part of the image description.

Note that the use of the free-form description attribute for transmitting image-related semantics within IPI-IIF communication must be based on a transmitter/receiver agreement. Otherwise, these image-related semantics will be lost for the receiver.

representation attribute — The purpose of a representation attribute is to convey information regarding the spatial nature of the imaging system that recorded the image. This will typically provide information as to the size of the image, the data types of the pixel values, *etc.*, but it may also include details such as the sampling raster involved (*e.g.*, rectangular or hexagonal), and the pixel aspect ratio.

4.4 IPI derived non-image data types

The following data types are considered to be non-image data types:

- basic data types, as defined in clause 4.2;
- image annotation data types (*e.g.*, textual descriptions, graphical descriptions of boundaries, audio tracks of motion pictures);
- image-related data types (*e.g.*, colour look-up tables, histograms).

Non-image data types, unlike image attributes, are not necessarily bound to an image. Image annotation and image-related data types are described in the following paragraphs.

4.4.1 IPI derived image annotation data types

Broad categories of image annotation data types are intended to provide a flexible interface to other non-imaging standards. These data types can be integrated into image data types to allow for textual, auditory, and graphical descriptions of the image content according to some application-specific requirements. These IPI-derived image annotation data types must be constructed by an application program from the elementary data types discussed above.

Text, graphics, and audio data types are not defined within ISO/IEC 12087. These data types are referenced only via unique names. Their detailed structure must be defined by an application program in accordance to some other standards (*e.g.*, [ISO/IEC 8613], [3], [5]) and their consistent use has to be guaranteed by the application program.

4.4.2 IPI derived image-related non-image data types

These non-image data types are also constructed from the elementary data types described in clause 4.2.1. As with image attributes, these types are described in a textual way by outlining their required functionality; details of the specific delineations are given in clause 5 and clause 6.

feature list — a collection of pairs of elements. The first element of the pair is a coordinate of a pixel, while the second refers to a basic data type storing the associated feature. The feature list forms the basis of the value bounds collection described in clause 5 and clause 6.

histogram — a construction that counts the frequency with which pixel values occur with respect to some partition $\{V_1, \dots, V_N\}$ of the total range of pixel values. The partition elements V_i ($i = 1, \dots, N$)