



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

**Part 2: Programmer's imaging kernel
system application programme interface**

The logo for Standards Australia, featuring a stylized graphic of overlapping circles and a swoosh.

STANDARDS
Australia

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 25 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 / S ISO/IEC 12087.2:2015.

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Standards may also be withdrawn.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 the 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 2: Programmer's imaging kernel system application programme interface

First published as AS ISO/IEC 12087.2:2015.

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 045 1

PREFACE

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

The objective of this Standard is to establish the specification for the application programme interface (API) called the Programmers Imaging Kernel System (PIKS). PIKS is intended to provide a rich set of both low-level and high-level services on image and image-derived data objects that can be used as building blocks for a broad range of common imaging applications.

This Standard is identical with, and has been reproduced from ISO/IEC 12087-2:1994, *Information technology—Computer graphics and image processing—Image Processing and Interchange (IPI)—Functional specification, Part 2: Programmer's imaging kernel system application program interface* and its Corrigendum 1 (1997) which has been added at the end of the source text.

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 mark.

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 12087 Information technology—Computer graphics and image processing—Image Processing and Interchange (IPI)—Functional specification	AS ISO/IEC 12087 Information technology—Computer graphics and image processing—Image Processing and Interchange (IPI)—Functional specification
12087-1 Part 1: Common architecture for imaging	12087.1 Part 1: Common architecture for imaging
12087-3 Part 3: Image Interchange Facility (IIF)	12087.3 Part 3: Image Interchange Facility (IIF)

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	Symbols and abbreviations	5
4	Programmer's Imaging Kernel System specification	11
4.1	PIKS imaging model	11
4.1.1	Image data objects	12
4.1.2	Non-image data objects	14
4.1.3	Data object creation	15
4.2	PIKS operators, tools, data object repository utilities, and system mechanisms	15
4.2.1	Operators	15
4.2.2	Tools	16
4.2.3	Data object repository	16
4.2.3.1	Impulse response function arrays	17
4.2.3.2	Dither arrays	17
4.2.3.3	Colour conversion matrices	17
4.2.4	Utilities	18
4.2.5	System mechanisms	18

	<i>Page</i>
4.3 PIKS operator model	30
4.3.1 Non-image to non-image operators	30
4.3.2 Image to non-image operators	30
4.3.3 Image to image operators	31
4.3.4 Neighbourhood operators	31
4.3.5 Operator index assignment	33
4.4 PIKS system mechanisms	35
4.4.1 Data object allocation	35
4.4.2 Match point control	35
4.4.3 ROI control	36
4.4.4 ROI data object creation and manipulation	40
4.4.5 Asynchronous control	41
4.4.6 Element chaining	43
4.4.6.1 Chain construction	44
4.4.6.2 Chain execution	45
4.4.7 Virtual register control	47
4.4.7.1 Virtual registers for storage of PIKS temporary variables	48
4.4.7.2 Virtual registers for asynchronous control	48
4.4.7.3 Virtual registers for chain iteration and conditional execution	48
4.4.7.4 Virtual registers for auditing asynchronous state	49
4.4.8 Global element control	49
4.4.9 Composite image management	50
4.4.9.1 Composite image identifier arrays	51
4.4.9.2 Composite image identifier lists	51
4.4.9.3 Composite image identifier records	52
4.4.10 PIKS error handling	53
4.4.11 PIKS operational states	55
4.5 PIKS utilities	57
4.5.1 Inquiry	57
4.5.2 Import and export	57
4.5.2.1 Data object import and export utilities	58
4.5.2.2 PIKS to application data type conversion	58

	<i>Page</i>	
5	PIKS conformance and extension methods	61
5.1	Foundation profile	61
5.2	Application profiles	61
5.3	Full profile	61
5.4	IIF capability profiles	61
5.5	PIKS profile definitions	61
5.6	Extension methods	74
6	PIKS element specification template	75
7	PIKS element specifications	83

Annexes:

A	Definitions of mathematical functions	817
A.1	Conventional mathematical symbols	817
A.2	Operational symbols	817
A.3	Mathematical functions	819
B	PIKS element support - source image structure	825
C	PIKS element support - destination image structure	835
D	PIKS element support - source and destination image data type	845
E	PIKS operator support - operator order	855
F	PIKS element functionality	861
G	PIKS data object repository	871
G.1	Impulse response function arrays	872
G.2	Dither arrays	888
G.3	Colour conversion matrices	889
H	PIKS image resampling	907
J	PIKS error codes	917
K	Bibliography of image processing books	925
L	Alphabetical listings of PIKS elements by profile	927
L.1	Alphabetical listing of PIKS elements in the Foundation profile	928
L.2	Alphabetical listing of PIKS elements in the Technical profile	932
L.3	Alphabetical listing of PIKS elements in the Scientific profile	938
L.4	Alphabetical listing of PIKS elements in the Full profile	946

List of figures

1	PIKS imaging model	11
2	PIKS operator model: non-image to non-image operators	30
3	PIKS operator model: image to non-image operators	31
4	PIKS operator model: image to image operators	32
5	Operator index assignment	33
6	Example of match point translation for image subtraction	37
7	Examples of ROI operation	39
8	Example of the relationship between a ROI and an image	41
9	Examples of PIKS element chains	45
10	PIKS operational state diagram	55
11	PIKS interface to the IIF gateway and an application	57
12	PIKS to application interface	59

List of tables

1	PIKS image objects	13
2	PIKS operators listed by functional class	19
3	PIKS tools listed by functional class	24
4	PIKS utilities listed by functional class	25
5	PIKS system mechanisms listed by functional class	28
6	PIKS data type codes	60
7	External physical data types supported by PIKS	60
8	Data types of PIKS non-image data objects supported in the Foundation profile	62
9	PIKS conformance profiles	63
10	Number of operators, tools, utilities, mechanisms, and tool mechanisms in each profile	64
11	PIKS elements in each profile	65

AUSTRALIAN STANDARD

Information technology—Computer graphics and image processing—Image Processing and Interchange (IPI)—Functional specification**Part 2:****Programmer's imaging kernel system application programme interface****1 Scope**

This part of ISO/IEC 12087 establishes the specification of the application program interface (API), called the Programmer's Imaging Kernel System (PIKS). ISO/IEC 12087-1 establishes the conceptual and architectural definitions of the Common Architecture for Imaging (CAI). ISO/IEC 12087-3 establishes the specification of the Image Interchange Facility (IIF).

PIKS is intended to provide a rich set of both low-level and high-level services on image and image-derived data objects. These services can be used as building blocks for a broad range of common imaging applications.

A conscious effort has been made by the developers of PIKS to create a standard that does not favor any particular computing system. Implementations of PIKS should be possible on computing systems ranging in architecture from general purpose computers to specialised hardware accelerators, ranging in size from personal computers to mainframe supercomputers, and ranging in connectivity from stand-alone machines to distributed computing networks.

Where applicable, PIKS relies on other APIs and data format standards to provide capabilities that are not unique to imaging. The following lists contain a summary of technological capabilities provided by PIKS and not provided by PIKS. However, it should be noted that PIKS functionality may be useful as a pre-processor or co-processor for many of the technologies in the "Not provided by PIKS" list.