



The APEX-CV Base Library

UG-10328-01-12

Copyright

Copyright © 2018 NXP Semiconductors Corporation ("NXP") All rights reserved.

This document contains information which is proprietary to NXP Semiconductors and may be used for non-commercial purposes within your organization in support of NXP Semiconductors' products. No other use or transmission of all or any part of this document is permitted without written permission from NXP Semiconductors, and must include all copyright and other proprietary notices. Use or transmission of all or any part of this document in violation of any applicable Canadian or other legislation is hereby expressly prohibited.

User obtains no rights in the information or in any product, process, technology or trademark which it includes or describes, and is expressly prohibited from modifying the information or creating derivative works without the express written consent of NXP Semiconductors.

Disclaimer

NXP Semiconductors assumes no responsibility for the accuracy or completeness of the information presented which is subject to change without notice. In no event will NXP Semiconductors be liable for any direct, indirect, special, incidental or consequential damages, including lost profits, lost business or lost data, resulting from the use of or reliance upon the information, whether or not NXP Semiconductors has been advised of the possibility of such damages.

Mention of non-NXP Semiconductors products or services is for information purposes only and constitutes neither an endorsement nor a recommendation.

Uncontrolled Copy

The master of this document is stored on NXP Semiconductors' document management system. Viewing of the master electronically ensures access to the current issue. Any hardcopies are considered uncontrolled copies.

Version	Details of Change	Author	Date
01	Initial Revision	Doxygen	July 4, 2014
02	Update to add Median, Convolve, Sobel, Prewitt filters, RGB2Y and RGB2YUV color conversion, Histogram and Integral Image	C. Garrard	August 1, 2014
03	Update to add Bilinear/Linear Interpolation, Accumulate, and Accumulate Squared	A. Saechao, J. Cairns	October 1, 2014
04	Update to supported box, dilate, and sobel data types	A. Saechao	October 31, 2014
05	APEX-CV base update: abs, clz	G. Billig	January 29, 2015
06	APEX-CV base update: added optimized filters under OPT namespace	A. Saechao	October 30, 2015
07	APEX-CV base update: added Histogram Equalization	N. Zhu	May. 13, 2016
08	APEX-CV base update: added VX Thresholding, Table Lookup and Channel Extract	A. Grigore	February 10, 2017
09	APEX-CV base update: update documentation to RTM 1.0 content	S. Francois	July 10, 2017
10	APEX-CV base update: update documentation to RTM 1.1 content	K. Pham	March 06, 2018
11	APEX-CV base add: Scharr filter, Convert bit depth, Min, Max, Pixel-wise Multiplication, APEX-CV base update: AbsDiff, Accumulate, Accumulate Squared, Accumulate Weighted, Addition, Subtract, Box Filter, Gaussian Filter	K. Pham	August 11, 2018
12	Umat replace by SUMat, APEX-CV base add: Mean, MeanStddev, Phase, APEX-CV base update: Median Filter	K. Pham	Dec 06, 2018

Contents

1	APEX-CV Base Library	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	6
3.1	apexcv::Abs Class Reference	6
3.1.1	Detailed Description	6
3.1.2	Member Function Documentation	6
3.1.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	6
3.1.2.2	Process()	7
3.1.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	7
3.1.2.4	SelectApexCore(int aApexId)	7
3.2	apexcv::AbsDiff Class Reference	8
3.2.1	Detailed Description	8
3.2.2	Member Function Documentation	8
3.2.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	8
3.2.2.2	Process()	8
3.2.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	9
3.2.2.4	SelectApexCore(int aApexId)	9
3.3	apexcv::Accumulate Class Reference	9
3.3.1	Detailed Description	10
3.3.2	Member Function Documentation	10
3.3.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	10
3.3.2.2	Process()	10
3.3.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	10
3.3.2.4	SelectApexCore(int aApexId)	11
3.4	apexcv::AccumulateSquared Class Reference	11
3.4.1	Detailed Description	11

3.4.2	Member Function Documentation	12
3.4.2.1	GetScale(uint8_t &aScale)	12
3.4.2.2	Initialize(vsdk::SUMat &aSrc, const uint8_t acScale, vsdk::SUMat &aDst)	12
3.4.2.3	Process()	12
3.4.2.4	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	12
3.4.2.5	SelectApexCore(int aApexId)	13
3.4.2.6	SetScale(const uint8_t aScale)	13
3.5	apexcv::AccumulateWeighted Class Reference	13
3.5.1	Detailed Description	14
3.5.2	Member Function Documentation	14
3.5.2.1	GetAlpha(float &aAlpha)	14
3.5.2.2	Initialize(vsdk::SUMat &aSrc, const float acAlpha, vsdk::SUMat &aDst)	14
3.5.2.3	Process()	15
3.5.2.4	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	15
3.5.2.5	SelectApexCore(int aApexId)	15
3.5.2.6	SetAlpha(const float acAlpha)	16
3.6	apexcv::Add Class Reference	16
3.6.1	Detailed Description	16
3.6.2	Member Function Documentation	17
3.6.2.1	GetPolicy(apexcv::eConvertPolicy &aPolicy)	17
3.6.2.2	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	17
3.6.2.3	Process()	17
3.6.2.4	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	17
3.6.2.5	SelectApexCore(int aApexId)	18
3.6.2.6	SetPolicy(apexcv::eConvertPolicy aPolicy)	18
3.7	apexcv::BilateralFilter Class Reference	18
3.7.1	Detailed Description	19
3.7.2	Member Function Documentation	19
3.7.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, int aSigmaColor, int aSigmaSpace, vsdk::SUMat &aDst)	19
3.7.2.2	Process()	20
3.7.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	20
3.7.2.4	SelectApexCore(int aApexId)	20
3.7.2.5	SetSigmaColor(int aSigmaColor)	20
3.7.2.6	SetSigmaSpace(int aSigmaSpace)	21
3.8	apexcv::BitwiseAND Class Reference	21
3.8.1	Detailed Description	21

3.8.2	Member Function Documentation	22
3.8.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	22
3.8.2.2	Process()	22
3.8.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	22
3.8.2.4	SelectApexCore(int aApexId)	23
3.9	apexcv::BitwiseNOT Class Reference	23
3.9.1	Detailed Description	23
3.9.2	Member Function Documentation	23
3.9.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	23
3.9.2.2	Process()	24
3.9.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	24
3.9.2.4	SelectApexCore(int aApexId)	24
3.10	apexcv::BitwiseOR Class Reference	25
3.10.1	Detailed Description	25
3.10.2	Member Function Documentation	25
3.10.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	25
3.10.2.2	Process()	25
3.10.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	26
3.10.2.4	SelectApexCore(int aApexId)	26
3.11	apexcv::BitwiseXOR Class Reference	26
3.11.1	Detailed Description	27
3.11.2	Member Function Documentation	27
3.11.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	27
3.11.2.2	Process()	27
3.11.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	28
3.11.2.4	SelectApexCore(int aApexId)	28
3.12	apexcv::BoxFilter Class Reference	28
3.12.1	Detailed Description	29
3.12.2	Member Function Documentation	29
3.12.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	29
3.12.2.2	Process()	29
3.12.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	29
3.12.2.4	SelectApexCore(int aApexId)	30
3.13	apexcv::BoxFilterHT Class Reference	30
3.13.1	Detailed Description	30
3.13.2	Member Function Documentation	31
3.13.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	31

3.13.2.2	Process()	31
3.13.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	31
3.13.2.4	SelectApexCore(int aApexId)	32
3.14	apexcv::CensusFilter Class Reference	32
3.14.1	Detailed Description	32
3.14.2	Member Function Documentation	32
3.14.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	32
3.14.2.2	Process()	33
3.14.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	33
3.14.2.4	SelectApexCore(int aApexId)	33
3.15	apexcv::Clz Class Reference	34
3.15.1	Detailed Description	34
3.15.2	Member Function Documentation	34
3.15.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	34
3.15.2.2	Process()	35
3.15.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	35
3.15.2.4	SelectApexCore(int aApexId)	35
3.16	apexcv::ColorConverter Class Reference	35
3.16.1	Detailed Description	36
3.16.2	Member Enumeration Documentation	36
3.16.2.1	ConversionType	36
3.16.3	Member Function Documentation	37
3.16.3.1	Initialize(vsdk::SUMat &aSrc, ConversionType aCT, int aR2YFactor, int aG2YFactor, int aB2YFactor, vsdk::SUMat &aDst)	37
3.16.3.2	Initialize(vsdk::SUMat &aSrc, ConversionType aCT, vsdk::SUMat &aDst)	37
3.16.3.3	Process()	37
3.16.3.4	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	38
3.16.3.5	SelectApexCore(int aApexId)	38
3.16.3.6	SetFactors(int aR2YFactor, int aG2YFactor, int aB2YFactor)	38
3.17	apexcv::ColorConverterHT Class Reference	39
3.17.1	Detailed Description	39
3.17.2	Member Enumeration Documentation	39
3.17.2.1	ConversionType	39
3.17.3	Member Function Documentation	40
3.17.3.1	Initialize(vsdk::SUMat &aSrc, ConversionType aCT, uint8_t aR2YFactor, uint8_t aG2YFactor, uint8_t aB2YFactor, uint16_t aShiftFactor, vsdk::SUMat &aDst)	40
3.17.3.2	Process()	40

3.17.3.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	40
3.17.3.4	SelectApexCore(int aApexId)	41
3.17.3.5	SetFactors(uint8_t aR2YFactor, uint8_t aG2YFactor, uint8_t aB2YFactor, uint16_t aShiftFactor)	41
3.18	apexcv::ConvertDepth Class Reference	41
3.18.1	Detailed Description	42
3.18.2	Member Function Documentation	42
3.18.2.1	GetPolicyType(apexcv::eConvertPolicy &aPolicy)	42
3.18.2.2	GetShift(int32_t &aShift)	42
3.18.2.3	Initialize(vsdk::SUMat &aSrc, const int32_t acShift, vsdk::SUMat &aDst)	43
3.18.2.4	Process()	43
3.18.2.5	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	43
3.18.2.6	SelectApexCore(int aApexId)	43
3.18.2.7	SetPolicyType(apexcv::eConvertPolicy aPolicy)	44
3.18.2.8	SetShift(const int32_t acShift)	44
3.19	apexcv::ConvolveFilter Class Reference	44
3.19.1	Detailed Description	45
3.19.2	Member Function Documentation	45
3.19.2.1	Initialize(vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[9], int aFilterScale, vsdk::SUMat &aDst)	45
3.19.2.2	Initialize(vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[25], int aFilterScale, vsdk::SUMat &aDst)	46
3.19.2.3	Process()	46
3.19.2.4	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	46
3.19.2.5	SelectApexCore(int aApexId)	46
3.19.2.6	SetFilterCoeff(signed char(&filterCoeff)[9])	47
3.19.2.7	SetFilterCoeff(signed char(&filterCoeff)[25])	47
3.19.2.8	SetFilterScale(int aFilterScale)	47
3.20	apexcv::ConvolveFilterHT Class Reference	48
3.20.1	Detailed Description	48
3.20.2	Member Function Documentation	49
3.20.2.1	Initialize(vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[9], signed char aFilterScale, vsdk::SUMat &aDst)	49
3.20.2.2	Initialize(vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[25], signed char aFilterScale, vsdk::SUMat &aDst)	49
3.20.2.3	Process()	49
3.20.2.4	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	50
3.20.2.5	SelectApexCore(int aApexId)	50

3.20.2.6	SetFilterCoeff(signed char(&filterCoeff)[9])	50
3.20.2.7	SetFilterCoeff(signed char(&filterCoeff)[25])	50
3.20.2.8	SetFilterScale(signed char aFilterScale)	51
3.21	apexcv::DerivativeXFilterHT Class Reference	51
3.21.1	Detailed Description	52
3.21.2	Member Function Documentation	52
3.21.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, signed char aK0, signed char aK1, signed char aK2, vsdk::SUMat &aDst)	52
3.21.2.2	Process()	52
3.21.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	52
3.21.2.4	SelectApexCore(int aApexId)	53
3.21.2.5	SetK0(signed char k0)	53
3.21.2.6	SetK1(signed char k1)	53
3.21.2.7	SetK2(signed char k2)	53
3.22	apexcv::DerivativeYFilterHT Class Reference	54
3.22.1	Detailed Description	54
3.22.2	Member Function Documentation	54
3.22.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, signed char aK0, signed char aK1, signed char aK2, vsdk::SUMat &aDst)	54
3.22.2.2	Process()	55
3.22.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	55
3.22.2.4	SelectApexCore(int aApexId)	55
3.22.2.5	SetK0(signed char k0)	56
3.22.2.6	SetK1(signed char k1)	56
3.22.2.7	SetK2(signed char k2)	56
3.23	apexcv::DilateFilter Class Reference	56
3.23.1	Detailed Description	57
3.23.2	Member Function Documentation	57
3.23.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	57
3.23.2.2	Process()	57
3.23.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	57
3.23.2.4	SelectApexCore(int aApexId)	58
3.24	apexcv::ErodeFilter Class Reference	58
3.24.1	Detailed Description	58
3.24.2	Member Function Documentation	59
3.24.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	59
3.24.2.2	Process()	59

3.24.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	59
3.24.2.4	SelectApexCore(int aApexId)	59
3.25	apexcv::ExtractChannel Class Reference	60
3.25.1	Detailed Description	60
3.25.2	Member Function Documentation	60
3.25.2.1	Initialize(vsdk::SUMat &aSrc, int aChannelIndex, vsdk::SUMat &aDst)	60
3.25.2.2	Process()	61
3.25.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	61
3.25.2.4	SelectApexCore(int aApexId)	61
3.26	apexcv::GaussianFilter Class Reference	62
3.26.1	Detailed Description	62
3.26.2	Member Function Documentation	62
3.26.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	62
3.26.2.2	Process()	62
3.26.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	63
3.26.2.4	SelectApexCore(int aApexId)	63
3.27	apexcv::Histogram Class Reference	63
3.27.1	Detailed Description	64
3.27.2	Member Function Documentation	64
3.27.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	64
3.27.2.2	Process()	64
3.27.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	64
3.27.2.4	SelectApexCore(int aApexId)	65
3.28	apexcv::InsertChannel Class Reference	65
3.28.1	Detailed Description	65
3.28.2	Member Function Documentation	66
3.28.2.1	Initialize(vsdk::SUMat &aSrc, int aChannelIndex, vsdk::SUMat &aDst)	66
3.28.2.2	Process()	66
3.28.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	66
3.28.2.4	SelectApexCore(int aApexId)	66
3.29	apexcv::IntegrallImage Class Reference	67
3.29.1	Detailed Description	67
3.29.2	Member Function Documentation	67
3.29.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	67
3.29.2.2	Process()	68
3.29.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	68
3.29.2.4	SelectApexCore(int aApexId)	68

3.30	apexcv::InterpolationBicubicGrayscale Class Reference	69
3.30.1	Detailed Description	69
3.30.2	Member Function Documentation	69
3.30.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aOffsetX, vsdk::SUMat &aOffsetY, vsdk::SUMat &aDst)	69
3.30.2.2	Process()	70
3.30.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aOffsetX, vsdk::SUMat &aOffsetY, vsdk::SUMat &aDst)	70
3.30.2.4	SelectApexCore(int aApexId)	70
3.31	apexcv::InterpolationBilinearGrayscale Class Reference	70
3.31.1	Detailed Description	71
3.31.2	Member Function Documentation	71
3.31.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDelta, vsdk::SUMat &aDst)	71
3.31.2.2	Process()	71
3.31.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDelta, vsdk::SUMat &aDst)	72
3.31.2.4	SelectApexCore(int aApexId)	72
3.32	apexcv::InterpolationLinearGrayscale Class Reference	72
3.32.1	Detailed Description	73
3.32.2	Member Function Documentation	73
3.32.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDeltaX, vsdk::SUMat &aDst)	73
3.32.2.2	Process()	73
3.32.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDeltaX, vsdk::SUMat &aDst)	73
3.32.2.4	SelectApexCore(int aApexId)	74
3.33	apexcv::Magnitude Class Reference	74
3.33.1	Detailed Description	74
3.33.2	Member Function Documentation	75
3.33.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	75
3.33.2.2	Process()	75
3.33.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	75
3.33.2.4	SelectApexCore(int aApexId)	76
3.34	apexcv::Max Class Reference	76
3.34.1	Detailed Description	76
3.34.2	Member Function Documentation	76
3.34.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	76
3.34.2.2	Process()	77
3.34.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	77
3.34.2.4	SelectApexCore(int aApexId)	77

3.35	apexcv::Mean Class Reference	78
3.35.1	Detailed Description	78
3.35.2	Member Function Documentation	78
3.35.2.1	Initialize(vsdk::SUMat &aSrc)	78
3.35.2.2	Process(float &aMean)	79
3.35.2.3	Process()	79
3.35.2.4	ReconnectIO(vsdk::SUMat &aSrc)	79
3.35.2.5	SelectApexCore(int aApexId)	79
3.36	apexcv::MeanStddev Class Reference	80
3.36.1	Detailed Description	80
3.36.2	Member Function Documentation	80
3.36.2.1	Initialize(vsdk::SUMat &aSrc)	80
3.36.2.2	Process()	80
3.36.2.3	Process(float &aMean, float &aStddev)	81
3.36.2.4	ReconnectIO(vsdk::SUMat &aSrc)	81
3.36.2.5	SelectApexCore(int aApexId)	81
3.37	apexcv::MedianFilter Class Reference	81
3.37.1	Detailed Description	82
3.37.2	Member Function Documentation	82
3.37.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	82
3.37.2.2	Process()	82
3.37.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	83
3.37.2.4	SelectApexCore(int aApexId)	83
3.38	apexcv::MergeChannel Class Reference	83
3.38.1	Detailed Description	84
3.38.2	Member Function Documentation	84
3.38.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aSrc4, vsdk::SUMat &aDst)	84
3.38.2.2	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aDst)	84
3.38.2.3	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	85
3.38.2.4	Process()	85
3.38.2.5	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aSrc4, vsdk::SUMat &aDst)	85
3.38.2.6	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aDst)	86
3.38.2.7	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	86
3.38.2.8	SelectApexCore(int aApexId)	86

3.39	apexcv::Min Class Reference	87
3.39.1	Detailed Description	87
3.39.2	Member Function Documentation	87
3.39.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	87
3.39.2.2	Process()	88
3.39.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	88
3.39.2.4	SelectApexCore(int aApexId)	88
3.40	apexcv::Mul Class Reference	89
3.40.1	Detailed Description	89
3.40.2	Member Function Documentation	89
3.40.2.1	GetPolicy(apexcv::eConvertPolicy &aPolicy)	89
3.40.2.2	GetScale(uint8_t &aScale)	90
3.40.2.3	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	90
3.40.2.4	Process()	90
3.40.2.5	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	91
3.40.2.6	SelectApexCore(int aApexId)	91
3.40.2.7	SetPolicy(apexcv::eConvertPolicy aPolicy)	91
3.40.2.8	SetScale(const uint8_t acScale)	92
3.41	apexcv::Phase Class Reference	92
3.41.1	Detailed Description	92
3.41.2	Member Function Documentation	92
3.41.2.1	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	92
3.41.2.2	Process()	93
3.41.2.3	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	93
3.41.2.4	SelectApexCore(int aApexId)	93
3.42	apexcv::PrewittXFilter Class Reference	94
3.42.1	Detailed Description	94
3.42.2	Member Function Documentation	94
3.42.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	94
3.42.2.2	Process()	95
3.42.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	95
3.42.2.4	SelectApexCore(int aApexId)	95
3.43	apexcv::PrewittYFilter Class Reference	95
3.43.1	Detailed Description	96
3.43.2	Member Function Documentation	96
3.43.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	96
3.43.2.2	Process()	96

3.43.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	97
3.43.2.4	SelectApexCore(int aApexId)	97
3.44	apexcv::SaturateFilterHT Class Reference	97
3.44.1	Detailed Description	98
3.44.2	Member Function Documentation	98
3.44.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	98
3.44.2.2	Process()	98
3.44.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	98
3.44.2.4	SelectApexCore(int aApexId)	99
3.45	apexcv::ScharrFilter Class Reference	99
3.45.1	Detailed Description	99
3.45.2	Member Function Documentation	99
3.45.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	99
3.45.2.2	Process()	100
3.45.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	100
3.45.2.4	SelectApexCore(int aApexId)	100
3.46	apexcv::ScharrXFilter Class Reference	101
3.46.1	Detailed Description	101
3.46.2	Member Function Documentation	101
3.46.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	101
3.46.2.2	Process()	102
3.46.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	102
3.46.2.4	SelectApexCore(int aApexId)	102
3.47	apexcv::ScharrXYFilter Class Reference	102
3.47.1	Detailed Description	103
3.47.2	Member Function Documentation	103
3.47.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)	103
3.47.2.2	Process()	103
3.47.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)	104
3.47.2.4	SelectApexCore(int aApexId)	104
3.48	apexcv::ScharrYFilter Class Reference	104
3.48.1	Detailed Description	105
3.48.2	Member Function Documentation	105
3.48.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	105
3.48.2.2	Process()	105
3.48.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	105

3.48.2.4	SelectApexCore(int aApexId)	106
3.49	apexcv::SeparableFilterHT Class Reference	106
3.49.1	Detailed Description	107
3.49.2	Member Function Documentation	107
3.49.2.1	Initialize(vsdk::SUMat &aSrc, signed char(&aFilterRow)[3], signed char(&aFilterCol)[3], vsdk::SUMat &aDst)	107
3.49.2.2	Initialize(vsdk::SUMat &aSrc, signed char(&aFilterRow)[5], signed char(&aFilterCol)[5], vsdk::SUMat &aDst)	107
3.49.2.3	Process()	108
3.49.2.4	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	108
3.49.2.5	SelectApexCore(int aApexId)	108
3.49.2.6	SetFilterCol(signed char(&aFilterCol)[3])	109
3.49.2.7	SetFilterCol(signed char(&aFilterCol)[5])	109
3.49.2.8	SetFilterRow(signed char(&aFilterRow)[3])	109
3.49.2.9	SetFilterRow(signed char(&aFilterRow)[5])	109
3.50	apexcv::SobelFilter Class Reference	109
3.50.1	Detailed Description	110
3.50.2	Member Function Documentation	110
3.50.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	110
3.50.2.2	Process()	110
3.50.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	110
3.50.2.4	SelectApexCore(int aApexId)	111
3.51	apexcv::SobelFilterHT Class Reference	111
3.51.1	Detailed Description	111
3.51.2	Member Function Documentation	112
3.51.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	112
3.51.2.2	Process()	112
3.51.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	112
3.51.2.4	SelectApexCore(int aApexId)	113
3.52	apexcv::SobelXFilter Class Reference	113
3.52.1	Detailed Description	113
3.52.2	Member Function Documentation	113
3.52.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	113
3.52.2.2	Process()	114
3.52.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	114
3.52.2.4	SelectApexCore(int aApexId)	114
3.53	apexcv::SobelXFilterHT Class Reference	115

3.53.1	Detailed Description	115
3.53.2	Member Function Documentation	115
3.53.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	115
3.53.2.2	Process()	116
3.53.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	116
3.53.2.4	SelectApexCore(int aApexId)	116
3.54	apexcv::SobelXYFilter Class Reference	116
3.54.1	Detailed Description	117
3.54.2	Member Function Documentation	117
3.54.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)	117
3.54.2.2	Process()	117
3.54.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)	118
3.54.2.4	SelectApexCore(int aApexId)	118
3.55	apexcv::SobelYFilter Class Reference	118
3.55.1	Detailed Description	119
3.55.2	Member Function Documentation	119
3.55.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	119
3.55.2.2	Process()	119
3.55.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	119
3.55.2.4	SelectApexCore(int aApexId)	120
3.56	apexcv::SobelYFilterHT Class Reference	120
3.56.1	Detailed Description	120
3.56.2	Member Function Documentation	121
3.56.2.1	Initialize(vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)	121
3.56.2.2	Process()	121
3.56.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	121
3.56.2.4	SelectApexCore(int aApexId)	122
3.57	apexcv::SplitChannel Class Reference	122
3.57.1	Detailed Description	122
3.57.2	Member Function Documentation	123
3.57.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::SUMat &aDst3, vsdk::SUMat &aDst4)	123
3.57.2.2	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::SUMat &aDst3)	123
3.57.2.3	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2)	123
3.57.2.4	Process()	124

3.57.2.5	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::SUMat &aDst3, vsdk::SUMat &aDst4)	124
3.57.2.6	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::SUMat &aDst3)	124
3.57.2.7	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2)	125
3.57.2.8	SelectApexCore(int aApexId)	125
3.58	apexcv::Subtract Class Reference	125
3.58.1	Detailed Description	126
3.58.2	Member Function Documentation	126
3.58.2.1	GetPolicy(apexcv::eConvertPolicy &aPolicy)	126
3.58.2.2	Initialize(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	126
3.58.2.3	Process()	127
3.58.2.4	ReconnectIO(vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)	127
3.58.2.5	SelectApexCore(int aApexId)	127
3.58.2.6	SetPolicy(apexcv::eConvertPolicy aPolicy)	129
3.59	apexcv::TableLookup Class Reference	129
3.59.1	Detailed Description	129
3.59.2	Member Function Documentation	129
3.59.2.1	Initialize(vsdk::SUMat &aSrc, vsdk::SUMat &acLut, vsdk::SUMat &aDst)	129
3.59.2.2	Process()	130
3.59.2.3	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &acLut, vsdk::SUMat &aDst)	130
3.59.2.4	SelectApexCore(int aApexId)	130
3.60	apexcv::Threshold Class Reference	131
3.60.1	Detailed Description	131
3.60.2	Member Function Documentation	131
3.60.2.1	GetOutputValues(uint8_t &aTrueVal, uint8_t &aFalseVal)	131
3.60.2.2	GetThreshold(uint32_t &aThreshold)	132
3.60.2.3	Initialize(vsdk::SUMat &aSrc, const uint32_t aThreshold, vsdk::SUMat &aDst)	132
3.60.2.4	Process()	132
3.60.2.5	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	133
3.60.2.6	SelectApexCore(int aApexId)	133
3.60.2.7	SetOutputValues(const uint8_t acTrueVal, const uint8_t acFalseVal)	133
3.60.2.8	SetThreshold(const uint32_t acThreshold)	134
3.61	apexcv::ThresholdRange Class Reference	134
3.61.1	Detailed Description	135
3.61.2	Member Function Documentation	135
3.61.2.1	GetOutputValues(uint8_t &aTrueVal, uint8_t &aFalseVal)	135

3.61.2.2	GetThresholds(uint32_t &aLowThreshold, uint32_t &aHighThreshold)	135
3.61.2.3	Initialize(vsdk::SUMat &aSrc, const uint32_t acLowThreshold, const uint32_t acHighThreshold, vsdk::SUMat &aDst)	135
3.61.2.4	Process()	136
3.61.2.5	ReconnectIO(vsdk::SUMat &aSrc, vsdk::SUMat &aDst)	136
3.61.2.6	SelectApexCore(int aApexId)	136
3.61.2.7	SetOutputValues(const uint8_t acTrueVal, const uint8_t acFalseVal)	137
3.61.2.8	SetThresholds(const uint32_t acLowThreshold, const uint32_t acHighThreshold) . . .	137

Bibliography	138
---------------------	------------

Index	139
--------------	------------

Chapter 1

APEX-CV Base Library

The APEX-CV Base library provides basic functionality for developers to design their own imaging-based applications while taking advantage of NXP's massively parallel APEX architecture. Currently various arithmetic operations, color conversions and image filters are provided as well as image calculations such as histogram and integral image as listed below.

- Arithmetic Operations:
 - Absolute value
 - Absolute difference
 - Accumulate
 - Accumulate squared
 - Accumulate weighted
 - Addition
 - Bitwise AND, NOT, OR, XOR
 - Count Leading Zeros
 - Magnitude
 - Max
 - Min
 - Pixel-wise Multiplication
 - Gradient Phase Computation
 - Subtraction
 - Table Lookup
 - Thresholding (binary)
 - Thresholding (range)
- Interpolation Operations:
 - Linear Grayscale
 - Bilinear Grayscale
 - Bicubic Grayscale
- Color Conversion and Channel Manipulation Operations:
 - Color conversion and color rotation

- Color conversion and color rotation (optimized)
 - Convert bit depth
 - Extract Channel
 - Insert Channel
 - Split Channel
 - Merge Channel
- Image Filters Operations:
 - Bilateral filter
 - Box filter
 - Box filter (optimized)
 - Census filter
 - Convolve filter
 - Convolve filter (optimized)
 - Derivative X filter (optimized)
 - Derivative Y filter (optimized)
 - Dilate filter
 - Erode filter
 - Gaussian filter
 - Median filter
 - Prewitt X filter
 - Prewitt Y filter
 - Saturate filter (optimized)
 - Scharr Filter
 - Scharr Filter X
 - Scharr Filter XY
 - Scharr Filter Y
 - Separable filter (optimized)
 - Sobel filter
 - Sobel filter (optimized)
 - Sobel X filter
 - Sobel X filter (optimized)
 - Sobel Y filter
 - Sobel Y filter (optimized)
 - Sobel XY filter
- Histogram Operations:
 - Histogram
 - Mean
 - Standard deviation
- Integral Image

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

apexcv::Abs	Absolute value, $aDst(x,y) = \text{abs}(aSrc(x,y))$	6
apexcv::AbsDiff	Absolute difference	8
apexcv::Accumulate	Accumulate	9
apexcv::AccumulateSquared	Accumulate Squared	11
apexcv::AccumulateWeighted	Accumulate Weighted	13
apexcv::Add	Add	16
apexcv::BilateralFilter	Bilateral filter	18
apexcv::BitwiseAND	Bitwise AND	21
apexcv::BitwiseNOT	Bitwise NOT	23
apexcv::BitwiseOR	Bitwise OR	25
apexcv::BitwiseXOR	Bitwise exclusive OR	26
apexcv::BoxFilter	Box filter	28
apexcv::BoxFilterHT	Box filter, Hand Tuned (optimized)	30
apexcv::CensusFilter	Census filter	32
apexcv::Clz	Count of Leading Zeros	34
apexcv::ColorConverter	Color converter class	35
apexcv::ColorConverterHT	Optimized color converter class containing support for converting image color types	39

apexcv::ConvertDepth	41
Converts image bit depth	
apexcv::ConvolveFilter	44
Convolve filter	
apexcv::ConvolveFilterHT	48
Convolve filter, Hand Tuned (optimized)	
apexcv::DerivativeXFilterHT	51
Derivative X filter, Hand Tuned (optimized)	
apexcv::DerivativeYFilterHT	54
Derivative Y filter, Hand Tuned (optimized)	
apexcv::DilateFilter	56
Dilate filter	
apexcv::ErodeFilter	58
Erode filter	
apexcv::ExtractChannel	60
Channel extract class containing support for extracting a single channel from a multi-channel image	
apexcv::GaussianFilter	62
Gaussian filter	
apexcv::Histogram	63
Histogram	
apexcv::InsertChannel	65
Channel insert class containing support for inserting a single channel in a multi-channel image	
apexcv::IntegralImage	67
Integral Image value	
apexcv::InterpolationBicubicGrayscale	69
Bicubic Grayscale Interpolation	
apexcv::InterpolationBilinearGrayscale	70
Bilinear Grayscale Interpolation	
apexcv::InterpolationLinearGrayscale	72
Linear Grayscale Interpolation	
apexcv::Magnitude	74
Magnitude	
apexcv::Max	76
Max	
apexcv::Mean	78
Mean	
apexcv::MeanStddev	80
MeanStddev	
apexcv::MedianFilter	81
Median filter	
apexcv::MergeChannel	83
Channel merge class containing support for merging multiple single channels images into a single multi-channel image	
apexcv::Min	87
Min	
apexcv::Mul	89
Multiplication	
apexcv::Phase	92
Phase	
apexcv::PrewittXFilter	94
Prewitt X filter	
apexcv::PrewittYFilter	95
Prewitt Y filter	

apexcv::SaturateFilterHT	
Saturate filter, Hand Tuned (optimized)	97
apexcv::ScharrFilter	
Scharr filter	99
apexcv::ScharrXFilter	
Scharr X filter	101
apexcv::ScharrXYFilter	
Scharr XY filter	102
apexcv::ScharrYFilter	
Scharr Y filter	104
apexcv::SeparableFilterHT	
Separable filter, Hand Tuned (optimized)	106
apexcv::SobelFilter	
Sobel filter	109
apexcv::SobelFilterHT	
Sobel filter, Hand Tuned (optimized)	111
apexcv::SobelXFilter	
Sobel X filter	113
apexcv::SobelXFilterHT	
Sobel X filter, Hand Tuned (optimized)	115
apexcv::SobelXYFilter	
Sobel XY filter	116
apexcv::SobelYFilter	
Sobel Y filter	118
apexcv::SobelYFilterHT	
Sobel Y filter, Hand Tuned (optimized)	120
apexcv::SplitChannel	
Channel split class containing support for splitting a single channel from a multi-channel image	122
apexcv::Subtract	
Subtract	125
apexcv::TableLookup	
Table Lookup	129
apexcv::Threshold	
Threshold	131
apexcv::ThresholdRange	
Threshold Range	134

Chapter 3

Class Documentation

3.1 apexcv::Abs Class Reference

Absolute value, $aDst(x,y) = \text{abs}(aSrc(x,y))$

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.1.1 Detailed Description

Absolute value, $aDst(x,y) = \text{abs}(aSrc(x,y))$

Object of this class computes the absolute value of every pixel.

Output dimensions are same as input.

Supported input type: VSDK_CV_8SC1, output type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels

3.1.2 Member Function Documentation

3.1.2.1 APEXCV_LIB_RESULT apexcv::Abs::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.1.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.1.2.3 APEXCV_LIB_RESULT apexcv::Abs::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.1.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

in	a↔ ApexId	ID of the APEX core used for performing the processing (0 or 1).
----	--------------	------------------------------------------------------------------

3.2 apexcv::AbsDiff Class Reference

Absolute difference.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.2.1 Detailed Description

Absolute difference.

Object of this class computes the absolute difference pixel for every pixel.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1

Supported input type: VSDK_CV_16SC1, output type: VSDK_CV_16SC1

Supported width: 128 to 2048 pixels

3.2.2 Member Function Documentation

3.2.2.1 APEXCV_LIB_RESULT apexcv::AbsDiff::Initialize (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).

3.2.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.2.2.3 APEXCV_LIB_RESULT apexcv::AbsDiff::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).

3.2.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

a↔ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------	------------------------------------------------------------------

3.3 apexcv::Accumulate Class Reference

Accumulate.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- **APEXCV_LIB_RESULT Process ()**

Start processing and return when done.

3.3.1 Detailed Description

Accumulate.

Object of this class accumulates *aSrc* into *aDst*.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1

Supported width: 128 to 2048 pixels

3.3.2 Member Function Documentation

3.3.2.1 APEXCV_LIB_RESULT apexcv::Accumulate::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_16SC1).

3.3.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.3.2.3 APEXCV_LIB_RESULT apexcv::Accumulate::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<code>in</code>	<code>aSrc</code>	Source image buffer (VSDK_CV_8UC1).
<code>in, out</code>	<code>aDst</code>	Destination image buffer (VSDK_CV_16SC1).

3.3.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.4 apexcv::AccumulateSquared Class Reference

Accumulate Squared.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, const uint8_t acScale, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SetScale** (const uint8_t aScale)
Set Scale.
- APEXCV_LIB_RESULT **GetScale** (uint8_t &aScale)
Get Scale.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.4.1 Detailed Description

Accumulate Squared.

Object of this class accumulates a squared value from *aSrc* to *_aDst*.
Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1
Supported width: 128 to 2048 pixels

3.4.2 Member Function Documentation

3.4.2.1 APEXCV_LIB_RESULT apexcv::AccumulateSquared::GetScale (uint8_t & aScale)

Get Scale.

This function allows to read the scale value.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aScale	Scale amount. Right shift the square of aSrc by aScale (0 <= aScale <= 15)
----	--------	----------------------------------------------------------------------------

3.4.2.2 APEXCV_LIB_RESULT apexcv::AccumulateSquared::Initialize (vsdk::SUMat & aSrc, const uint8_t acScale, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	acScale	Scale amount. Right shift the square of aSrc by aScale (0 <= aScale <= 15)
in, out	aDst	Destination image buffer (VSDK_CV_16SC1).

3.4.2.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.4.2.4 APEXCV_LIB_RESULT apexcv::AccumulateSquared::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_16SC1).

3.4.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

a↔ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------	------------------------------------------------------------------

3.4.2.6 APEXCV_LIB_RESULT apexcv::AccumulateSquared::SetScale (const uint8_t aScale)

Set Scale.

This function allows to change the scale value.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aScale	Scale amount. Right shift the square of aSrc by aScale (0 <= aScale <= 15)
----	--------	----------------------------------------------------------------------------

3.5 apexcv::AccumulateWeighted Class Reference

Accumulate Weighted.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, const float acAlpha, vsdk::SUMat &aDst)

Initialize object (required).

- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)

Reconnect IO (optional).

- APEXCV_LIB_RESULT **SetAlpha** (const float acAlpha)

Set Alpha.

- APEXCV_LIB_RESULT **GetAlpha** (float &aAlpha)

Set Alpha.

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.5.1 Detailed Description

Accumulate Weighted.

Object of this class accumulates a weight value from *aSrc* to *aDst*

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1 or

Supported width: 128 to 2048 pixels.

3.5.2 Member Function Documentation

3.5.2.1 APEXCV_LIB_RESULT apexcv::AccumulateWeighted::GetAlpha (float & *aAlpha*)

Set Alpha.

This function allows to read the alpha value.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aAlpha</i>	Weight amount. Scalar value with a value in the range of [0, 1]
----	---------------	-----------------------------------------------------------------

3.5.2.2 APEXCV_LIB_RESULT apexcv::AccumulateWeighted::Initialize (vsdk::SUMat & *aSrc*, const float *acAlpha*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in	<i>acAlpha</i>	Weight amount. Scalar value with a value in the range of [0, 1]
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.5.2.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.5.2.4 APEXCV_LIB_RESULT apexcv::AccumulateWeighted::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.5.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.5.2.6 APEXCV_LIB_RESULT apexcv::AccumulateWeighted::SetAlpha (const float *acAlpha*)

Set Alpha.

This function allows to change the alpha value.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>acAlpha</i>	Weight amount. Scalar value with a value in the range of [0, 1]
----	----------------	-----------------------------------------------------------------

3.6 apexcv::Add Class Reference

Add.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT [SetPolicy](#) (apexcv::eConvertPolicy aPolicy)
Set Policy type.
- APEXCV_LIB_RESULT [GetPolicy](#) (apexcv::eConvertPolicy &aPolicy)
Get Policy type.
- APEXCV_LIB_RESULT [SelectApexCore](#) (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT [Process](#) ()
Start processing and return when done.

3.6.1 Detailed Description

Add.

Object of this class adds pixel value from *aSrc1* and *aSrc2* pixel by pixel. *aDst* can be VSDK_CV_8UC1 only if both source images are VSDK_CV_8UC1 and *aDst* is explicitly set to VSDK_CV_8UC1. It is otherwise VSDK_CV_16SC1. Supported input 1 type: VSDK_CV_8UC1, input 2 type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1 or Supported input 1 type: VSDK_CV_8UC1, input 2 type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1 or Supported input 1 type: VSDK_CV_8UC1, input 2 type: VSDK_CV_16SC1, output type: VSDK_CV_16SC1 or Supported input 1 type: VSDK_CV_16SC1, input 2 type: VSDK_CV_16SC1, output type: VSDK_CV_16SC1 Supported width: 128 to 2048 pixels.

3.6.2 Member Function Documentation

3.6.2.1 APEXCV_LIB_RESULT apexcv::Add::GetPolicy (apexcv::eConvertPolicy & *aPolicy*)

Get Policy type.

This function allows to read the value of the overflow policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	<i>aPolicy</i>	Overflow policy type.
-----	----------------	-----------------------

3.6.2.2 APEXCV_LIB_RESULT apexcv::Add::Initialize (vsdk::SUMat & *aSrc1*, vsdk::SUMat & *aSrc2*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.6.2.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.6.2.4 APEXCV_LIB_RESULT apexcv::Add::ReconnectIO (vsdk::SUMat & *aSrc1*, vsdk::SUMat & *aSrc2*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.6.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.6.2.6 APEXCV_LIB_RESULT apexcv::Add::SetPolicy (apexcv::eConvertPolicy *aPolicy*)

Set Policy type.

This function allows to change the overflow policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aPolicy</i>	Overflow policy type
----	----------------	----------------------

3.7 apexcv::BilateralFilter Class Reference

Bilateral filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, int aSigmaColor, int aSigmaSpace, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SetSigmaColor** (int aSigmaColor)
Set sigmaColor.
- APEXCV_LIB_RESULT **SetSigmaSpace** (int aSigmaSpace)
Set sigmaSpace.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.7.1 Detailed Description

Bilateral filter.

Object of this class applies a bilateral filter on *aSrc*. *aSigmaColor* represents the weight of color/intensity difference and *aSigmaSpace* represents the weight of spacial difference. See: [1] for more information.

Supported window size: 5 x 5

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.7.2 Member Function Documentation

3.7.2.1 APEXCV_LIB_RESULT apexcv::BilateralFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, int aSigmaColor, int aSigmaSpace, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 5.
in	<i>aSigmaColor</i>	Sigma value for color space.
in	<i>aSigmaSpace</i>	Sigma value for distance space.
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.7.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.7.2.3 APEXCV_LIB_RESULT apexcv::BilateralFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
<i>in, out</i>	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.7.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.7.2.5 APEXCV_LIB_RESULT apexcv::BilateralFilter::SetSigmaColor (int aSigmaColor)

Set sigmaColor.

This function allows to change the value of sigmaColor

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSigmaColor</i>	Sigma value for color space.
----	--------------------	------------------------------

3.7.2.6 APEXCV_LIB_RESULT apexcv::BilateralFilter::SetSigmaSpace (int *aSigmaSpace*)

Set sigmaSpace.

This function allows to change the value of sigmaSpace

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>aSigmaSpace</i>	Sigma value for distance space.
--------------------	---------------------------------

3.8 apexcv::BitwiseAND Class Reference

Bitwise AND.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.8.1 Detailed Description

Bitwise AND.

Object of this class performs a bitwise AND between pixel value of *aSrc1* and *aSrc2* pixel by pixel.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1, or

Supported input type: VSDK_CV_16UC1, output type: VSDK_CV_16UC1, or

Supported input type: VSDK_CV_32UC1, output type: VSDK_CV_32UC1

Supported width: 128 to 2048 pixels.

3.8.2 Member Function Documentation

3.8.2.1 APEXCV_LIB_RESULT apexcv::BitwiseAND::Initialize (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).

3.8.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.8.2.3 APEXCV_LIB_RESULT apexcv::BitwiseAND::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).

3.8.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.9 apexcv::BitwiseNOT Class Reference

Bitwise NOT.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.9.1 Detailed Description

Bitwise NOT.

Object of this class performs a bitwise NOT of pixel value of *aSrc* pixel by pixel.
Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1
Supported width: 128 to 2048 pixels.

3.9.2 Member Function Documentation

3.9.2.1 APEXCV_LIB_RESULT apexcv::BitwiseNOT::Initialize (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.9.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.9.2.3 APEXCV_LIB_RESULT apexcv::BitwiseNOT::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.9.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

a↔ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------	------------------------------------------------------------------

3.10 apexcv::BitwiseOR Class Reference

Bitwise OR.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.10.1 Detailed Description

Bitwise OR.

Object of this class performs a bitwise OR between pixel value of *aSrc1* and *aSrc2* pixel by pixel.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1, or

Supported input type: VSDK_CV_16UC1, output type: VSDK_CV_16UC1, or

Supported input type: VSDK_CV_32UC1, output type: VSDK_CV_32UC1

Supported width: 128 to 2048 pixels.

3.10.2 Member Function Documentation

3.10.2.1 APEXCV_LIB_RESULT apexcv::BitwiseOR::Initialize (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).

3.10.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.10.2.3 APEXCV_LIB_RESULT apexcv::BitwiseOR::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).

3.10.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.11 apexcv::BitwiseXOR Class Reference

Bitwise exclusive OR.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)

Initialize object (required).

- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)

Reconnect IO (optional).

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.11.1 Detailed Description

Bitwise exclusive OR.

Object of this class performs a bitwise XOR between pixel value of *aSrc1* and *aSrc2* pixel by pixel.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1, or

Supported input type: VSDK_CV_16UC1, output type: VSDK_CV_16UC1, or

Supported input type: VSDK_CV_32UC1, output type: VSDK_CV_32UC1

Supported width: 128 to 2048 pixels.

3.11.2 Member Function Documentation

3.11.2.1 APEXCV_LIB_RESULT apexcv::BitwiseXOR::Initialize (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).

3.11.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.11.2.3 APEXCV_LIB_RESULT apexcv::BitwiseXOR::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16UC1 or VSDK_CV_32UC1).

3.11.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.12 apexcv::BoxFilter Class Reference

Box filter.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.12.1 Detailed Description

Box filter.

Object of this class applies a box filter to *aSrc*. Supported window size: 3x3, 5x5, 7x7 and 9x9 (for VSDK_CV_8UC1)

Supported window size: 3x3 and 5x5 (for VSDK_CV_16SC1)

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1 and VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.12.2 Member Function Documentation

3.12.2.1 APEXCV_LIB_RESULT apexcv::BoxFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1)
in	<i>aWindowSize</i>	Window size (3, 5, 7 or 9)
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1)

3.12.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.12.2.3 APEXCV_LIB_RESULT apexcv::BoxFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).

3.12.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.13 apexcv::BoxFilterHT Class Reference

Box filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.13.1 Detailed Description

Box filter, Hand Tuned (optimized).

Object of this class applies a Box filter to aSrc. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3

aDst and *aSrc* must have identical dimensions.
Supported input type: VSDK_CV_8UC1.
Supported width: 128 to 2048 pixels.

3.13.2 Member Function Documentation

3.13.2.1 APEXCV_LIB_RESULT apexcv::BoxFilterHT::Initialize (vsdk::SUMat & *aSrc*, int *aWindowSize*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.13.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.13.2.3 APEXCV_LIB_RESULT apexcv::BoxFilterHT::ReconnectIO (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.13.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.14 apexcv::CensusFilter Class Reference

Census filter.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.14.1 Detailed Description

Census filter.

Object of this class applies a census filter to *aSrc*. Supported window size: 3 x 3
aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.14.2 Member Function Documentation

3.14.2.1 APEXCV_LIB_RESULT apexcv::CensusFilter::Initialize (vsdk::SUMat & *aSrc*, int *aWindowSize*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source Image buffer (VSDK_CV_8UC1).
in	aWindowSize	Window Size (3).
in, out	aDst	Destination Image buffer (VSDK_CV_8UC1).

3.14.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.14.2.3 APEXCV_LIB_RESULT apexcv::CensusFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.14.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftarrow$ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
---------------------------------	------------------------------------------------------------------

3.15 apexcv::Clz Class Reference

Count of Leading Zeros.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.15.1 Detailed Description

Count of Leading Zeros.

Object of this class counts the number of leading zeros in the pixel value of *aSrc* pixel by pixel.

Supported input type: VSDK_CV_8UC1, VSDK_CV_8SC1, VSDK_CV_16UC1 and VSDK_CV_16SC1

Supported output type: VSDK_CV_8UC1

Supported width: 128 to 2048 pixels.

3.15.2 Member Function Documentation

3.15.2.1 APEXCV_LIB_RESULT apexcv::Clz::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1, VSDK_CV_8SC1, VSDK_CV_16UC1 or VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.15.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.15.2.3 APEXCV_LIB_RESULT apexcv::Clz::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1, VSDK_CV_8SC1, VSDK_CV_16UC1 or VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.15.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------------------	------------------------------------------------------------------

3.16 apexcv::ColorConverter Class Reference

Color converter class.

Public Types

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, **ConversionType** aCT, int aR2YFactor, int aG2YFactor, int aB2YFactor, vsdk::SUMat &aDst)
Convert function.
- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, **ConversionType** aCT, vsdk::SUMat &aDst)
Convert function. ..
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO. ..
- APEXCV_LIB_RESULT **SetFactors** (int aR2YFactor, int aG2YFactor, int aB2YFactor)
Set factors for RGB888x to Y.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.16.1 Detailed Description

Color converter class.

Object of this class performs color conversions of images.

See **ConversionType** for supported conversions.

Supported width: 128 to 2048 pixels.

3.16.2 Member Enumeration Documentation

3.16.2.1 enum apexcv::ColorConverter::ConversionType

List of conversion types.

Enumerator

- eRGB565_TO_RGB888X** 16-bit RGB565 (VSDK_CV_16UC1) to 32-bit representation of RGB888X (VSDK_CV_32SC1)
- eRGB888X_TO_RGB565** 32-bit representation of RGB888X (VSDK_CV_32SC1) to 16-bit RGB565 (VSDK_CV_16UC1)
- eRGB888X_TO_Y** 4-tuple 8-bit R, G, B, X (VSDK_CV_8UC4) to 8-bit Y (VSDK_CV_8UC1)
- eRGB888X_TO_YUV** 4-tuple 8-bit R, G, B, X (VSDK_CV_8UC4) to 4-tuple 8-bit Y, U, V, X (VSDK_CV_8UC4)
- eRGB888_TO_GREY** 3-tuple 8-bit R, G, B (VSDK_CV_8UC3) to 8-bit Grey (VSDK_CV_8UC1), $(R*21 + G*72 + B*7)$
- eBGR888_TO_GREY** 3-tuple 8-bit B, G, R (VSDK_CV_8UC3) to 8-bit Grey (VSDK_CV_8UC1), $(R*21 + G*72 + B*7)$
- eGREY_TO_RGB888** 8-bit grey (VSDK_CV_8UC1) to 3-tuple 8-bit B, G, R (VSDK_CV_8UC3), duplication on all 3 channels

3.16.3 Member Function Documentation

3.16.3.1 APEXCV_LIB_RESULT apexcv::ColorConverter::Initialize (vsdk::SUMat & aSrc, ConversionType aCT, int aR2YFactor, int aG2YFactor, int aB2YFactor, vsdk::SUMat & aDst)

Convert function.

Converts an image from one type to another based on [ConversionType](#). R2YFactor, G2YFactor and B2YFactor are Q0.8 fixed point values used with RGB888X_TO_Y following the formula: $Y = \left\lfloor \frac{R2YFactor}{256} * R + \frac{G2YFactor}{256} * G + \frac{B2YFactor}{256} * B + 0.5 \right\rfloor$. For example, conversion following Recommendation ITU-R BT.601-7 (<http://www.itu.int/rec/R-REC-BT.601-7-201103-I/en>) would use factor values of 77(0.299), 150(0.587) and 29(0.114).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer (Use the type corresponding to the ConversionType selected).
in	aCT	Color conversion type. See ConversionType
in	aR2YFactor	Conversion factor for red used with RGB888X_TO_Y
in	aG2YFactor	Conversion factor for green used with RGB888X_TO_Y
in	aB2YFactor	Conversion factor for blue used with RGB888X_TO_Y
in, out	aDst	Destination memory buffer (Use the type corresponding to the ConversionType selected).

3.16.3.2 APEXCV_LIB_RESULT apexcv::ColorConverter::Initialize (vsdk::SUMat & aSrc, ConversionType aCT, vsdk::SUMat & aDst)

Convert function. ..

Converts an image from one type to another based on [ConversionType](#). R2YFactor, G2YFactor and B2YFactor are Q0.8 fixed point values used with RGB888X_TO_Y following the formula: $Y = \left\lfloor \frac{R2YFactor}{256} * R + \frac{G2YFactor}{256} * G + \frac{B2YFactor}{256} * B + 0.5 \right\rfloor$. For example, conversion following Recommendation ITU-R BT.601-7 (<http://www.itu.int/rec/R-REC-BT.601-7-201103-I/en>) would use factor values of 77(0.299), 150(0.587) and 29(0.114).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer.
in	aCT	Color conversion type. See ConversionType
in, out	aDst	Destination memory buffer.

3.16.3.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.16.3.4 APEXCV_LIB_RESULT apexcvcv::ColorConverter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO. ..

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer (type should be the same one used when calling Initialize).
in, out	aDst	Destination memory buffer (type should be the same one used when calling Initialize).

3.16.3.5 APEXCV_LIB_RESULT apexcvcv::ApexcvcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.16.3.6 APEXCV_LIB_RESULT apexcvcv::ColorConverter::SetFactors (int aR2YFactor, int aG2YFactor, int aB2YFactor)

Set factors for RGB888x to Y.

This function allows to change factors without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>aR2YFactor</i>	Conversion factor for red used with RGB888X_TO_Y
<i>aG2YFactor</i>	Conversion factor for green used with RGB888X_TO_Y
<i>aB2YFactor</i>	Conversion factor for blue used with RGB888X_TO_Y

3.17 apexcv::ColorConverterHT Class Reference

Optimized color converter class containing support for converting image color types.

Public Types

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, **ConversionType** aCT, uint8_t aR2YFactor, uint8_t aG2YFactor, uint8_t aB2YFactor, uint16_t aShiftFactor, vsdk::SUMat &aDst)
Convert function.
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO.
- APEXCV_LIB_RESULT **SetFactors** (uint8_t aR2YFactor, uint8_t aG2YFactor, uint8_t aB2YFactor, uint16_t aShiftFactor)
Set factors for RGB888x to Y.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.17.1 Detailed Description

Optimized color converter class containing support for converting image color types.

This class is an interface for using color conversion functions on the host.

3.17.2 Member Enumeration Documentation

3.17.2.1 enum apexcv::ColorConverterHT::ConversionType

List of conversion types.

Enumerator

eHT_RGB888X_TO_Y 4-tuple 8 bit A, B, C, X (VSDK_CV_8UC4) to signed 16 bit Y (VSDK_CV_16SC1)

3.17.3 Member Function Documentation

3.17.3.1 APEXCV_LIB_RESULT apexcv::ColorConverterHT::Initialize (vsdk::SUMat & aSrc, ConversionType aCT, uint8_t aR2YFactor, uint8_t aG2YFactor, uint8_t aB2YFactor, uint16_t aShiftFactor, vsdk::SUMat & aDst)

Convert function.

Converts an image from one type to another based on [ConversionType](#). R2YFactor, G2YFactor and B2YFactor are Q0.8 fixed point values used with RGB888X_TO_Y following the formula: $Y = \left\lfloor \frac{R2YFactor}{256} * R + \frac{G2YFactor}{256} * G + \frac{B2YFactor}{256} * B + 0.5 \right\rfloor$. For example, conversion following Recommendation ITU-R BT.601-7 (<http://www.itu.int/rec/R-REC-BT.601-7-201103-I/en>) would use factor values of 77(0.299), 150(0.587) and 29(0.114).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer (Use the type corresponding to the ConversionType selected).
in	aCT	Color conversion type. See ConversionType
in	aR2YFactor	Conversion factor for red used with RGB888X_TO_Y
in	aG2YFactor	Conversion factor for green used with RGB888X_TO_Y
in	aB2YFactor	Conversion factor for blue used with RGB888X_TO_Y
in	aShiftFactor	Shift factor. Use 0 by default. Used in HT_RGB888X_TO_Y.
in, out	aDst	Destination memory buffer (Use the type corresponding to the ConversionType selected).

3.17.3.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.17.3.3 APEXCV_LIB_RESULT apexcv::ColorConverterHT::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer (type should be the same one used when calling Initialize).
in, out	aDst	Destination memory buffer (type should be the same one used when calling Initialize).

3.17.3.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.17.3.5 APEXCV_LIB_RESULT apexcv::ColorConverterHT::SetFactors (uint8_t *aR2YFactor*, uint8_t *aG2YFactor*, uint8_t *aB2YFactor*, uint16_t *aShiftFactor*)

Set factors for RGB888x to Y.

This function allows to change factors without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aR2YFactor</i>	Conversion factor for red used with HT_RGB888X_TO_Y
in	<i>aG2YFactor</i>	Conversion factor for green used with HT_RGB888X_TO_Y
in	<i>aB2YFactor</i>	Conversion factor for blue used with HT_RGB888X_TO_Y
in	<i>aShiftFactor</i>	Shift factor. Use 0 by default. Used in HT_RGB888X_TO_Y.

3.18 apexcv::ConvertDepth Class Reference

Converts image bit depth.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, const int32_t acShift, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT [SetShift](#) (const int32_t acShift)
Set Shift.
- APEXCV_LIB_RESULT [GetShift](#) (int32_t &aShift)

Get Shift.

- APEXCV_LIB_RESULT **SetPolicyType** (apexcv::eConvertPolicy aPolicy)

Set Policy type.

- APEXCV_LIB_RESULT **GetPolicyType** (apexcv::eConvertPolicy &aPolicy)

Get Policy type.

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.18.1 Detailed Description

Converts image bit depth.

Object of this class performs converts image bit depth.

Up convert, shifting left. Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1

Down convert, shifting right. Supported input type: VSDK_CV_16SC1, output type: VSDK_CV_8UC1

Supported width: 128 to 2048 pixels.

3.18.2 Member Function Documentation

3.18.2.1 APEXCV_LIB_RESULT apexcv::ConvertDepth::GetPolicyType (apexcv::eConvertPolicy & aPolicy)

Get Policy type.

This function allows to read the value of the Policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	aPolicy	Policy.
-----	---------	---------

3.18.2.2 APEXCV_LIB_RESULT apexcv::ConvertDepth::GetShift (int32_t & aShift)

Get Shift.

This function allows to read the value of the shift.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	aShift	shift.
-----	--------	--------

3.18.2.3 APEXCV_LIB_RESULT apexcv::ConvertDepth::Initialize (vsdk::SUMat & aSrc, const int32_t acShift, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in	acShift	Source pixel value shift amount (0 <= aShift < 8).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).

3.18.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.18.2.5 APEXCV_LIB_RESULT apexcv::ConvertDepth::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).

3.18.2.6 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

a_{\leftrightarrow} <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------------------------	------------------------------------------------------------------

3.18.2.7 APEXCV_LIB_RESULT apexcv::ConvertDepth::SetPolicyType (apexcv::eConvertPolicy aPolicy)

Set Policy type.

This function allows to change the Policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aPolicy</i>	Policy type
----	----------------	-------------

3.18.2.8 APEXCV_LIB_RESULT apexcv::ConvertDepth::SetShift (const int32_t aShift)

Set Shift.

This function allows to change the shift fact value.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aShift</i>	Source pixel value shift amount (0 <= aShift < 8)
----	---------------	----------------------------------------------------

3.19 apexcv::ConvolveFilter Class Reference

Convolve filter.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[9], int aFilterScale, vsdk::SUMat &aDst)
Initialize object (required).

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[25], int aFilterScale, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SetFilterScale** (int aFilterScale)
Set Filter Scale.
- APEXCV_LIB_RESULT **SetFilterCoeff** (signed char(&filterCoeff)[9])
Set Filter Coefficients.
- APEXCV_LIB_RESULT **SetFilterCoeff** (signed char(&filterCoeff)[25])
Set Filter Coefficients.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.19.1 Detailed Description

Convolve filter.

Object of this class applies a Convolve filter to *aSrc*. Supported window size: 3 x 3 or 5 x 5
aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.19.2 Member Function Documentation

3.19.2.1 APEXCV_LIB_RESULT apexcv::ConvolveFilter::Initialize (vsdk::SUMat & aSrc, signed char(&) aFilterCoeff[9], int aFilterScale, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aFilterCoeff</i>	9 Coefficients for 3x3 kernel.
in	<i>aFilterScale</i>	Right Shift to scale the data.
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.19.2.2 APEXCV_LIB_RESULT apexcv::ConvolveFilter::Initialize (vsdk::SUMat & aSrc, signed char(&) aFilterCoeff[25], int aFilterScale, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source Image buffer (VSDK_CV_8UC1).
in	aFilterCoeff	25 Coefficients for 5x5 kernel.
in	aFilterScale	Right Shift to scale the data.
in, out	aDst	Destination Image buffer (VSDK_CV_8UC1).

3.19.2.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.19.2.4 APEXCV_LIB_RESULT apexcv::ConvolveFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.19.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.19.2.6 APEXCV_LIB_RESULT apexcvcv::ConvolveFilter::SetFilterCoeff (signed char(& filterCoeff[9])

Set Filter Coefficients.

This function allows to change the filter 9 coefficients for 3x3 kernels

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>filterCoeff</i>	9 Coefficients for 3x3 kernel.
----	--------------------	--------------------------------

3.19.2.7 APEXCV_LIB_RESULT apexcvcv::ConvolveFilter::SetFilterCoeff (signed char(& filterCoeff[25])

Set Filter Coefficients.

This function allows to change the filter 25 coefficients for 5x5 kernels

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>filterCoeff</i>	25 Coefficients for 5x5 kernel.
----	--------------------	---------------------------------

3.19.2.8 APEXCV_LIB_RESULT apexcvcv::ConvolveFilter::SetFilterScale (int aFilterScale)

Set Filter Scale.

This function allows to change the filter scale (right shift).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aFilterScale</i>	Set the filter Scale.
----	---------------------	-----------------------

3.20 apexcv::ConvolveFilterHT Class Reference

Convolve filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[9], signed char aFilterScale, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, signed char(&aFilterCoeff)[25], signed char aFilterScale, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SetFilterScale** (signed char aFilterScale)
Set Filter Scale.
- APEXCV_LIB_RESULT **SetFilterCoeff** (signed char(&filterCoeff)[9])
Set Filter Coefficients.
- APEXCV_LIB_RESULT **SetFilterCoeff** (signed char(&filterCoeff)[25])
Set Filter Coefficients.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.20.1 Detailed Description

Convolve filter, Hand Tuned (optimized).

Object of this class applies a generic convolution filter to *aSrc*. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3 and 5 x 5
aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.20.2 Member Function Documentation

3.20.2.1 APEXCV_LIB_RESULT apexcv::ConvolveFilterHT::Initialize (vsdk::SUMat & aSrc, signed char(&) aFilterCoeff[9], signed char aFilterScale, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source Image buffer (VSDK_CV_8UC1).
in	aFilterCoeff	9 Coefficients for 3x3 kernel.
in	aFilterScale	Right Shift to scale the data.
in, out	aDst	Destination Image buffer (VSDK_CV_8UC1).

3.20.2.2 APEXCV_LIB_RESULT apexcv::ConvolveFilterHT::Initialize (vsdk::SUMat & aSrc, signed char(&) aFilterCoeff[25], signed char aFilterScale, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source Image buffer (VSDK_CV_8UC1).
in	aFilterCoeff	25 Coefficients for 5x5 kernel.
in	aFilterScale	Right Shift to scale the data.
in, out	aDst	Destination Image buffer (VSDK_CV_8UC1).

3.20.2.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.20.2.4 APEXCV_LIB_RESULT apexcv::ConvolveFilterHT::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.20.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

a↔ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------	------------------------------------------------------------------

3.20.2.6 APEXCV_LIB_RESULT apexcv::ConvolveFilterHT::SetFilterCoeff (signed char(&) filterCoeff[9])

Set Filter Coefficients.

This function allows to change the filter 9 coefficients for 3x3 kernels

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	filterCoeff	9 Coefficients for 3x3 kernel.
----	-------------	--------------------------------

3.20.2.7 APEXCV_LIB_RESULT apexcv::ConvolveFilterHT::SetFilterCoeff (signed char(&) filterCoeff[25])

Set Filter Coefficients.

This function allows to change the filter 25 coefficients for 5x5 kernels

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>filterCoeff</i>	25 Coefficients for 5x5 kernel.
----	--------------------	---------------------------------

3.20.2.8 APEXCV_LIB_RESULT apexcvcv::ConvolveFilterHT::SetFilterScale (signed char *aFilterScale*)

Set Filter Scale.

This function allows to change the filter scale (right shift).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aFilterScale</i>	Set the filter Scale.
----	---------------------	-----------------------

3.21 apexcvcv::DerivativeXFilterHT Class Reference

Derivative X filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, int aWindowSize, signed char aK0, signed char aK1, signed char aK2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT [SetK0](#) (signed char k0)
Set K0.
- APEXCV_LIB_RESULT [SetK1](#) (signed char k1)
Set K1.
- APEXCV_LIB_RESULT [SetK2](#) (signed char k2)
Set K2.
- APEXCV_LIB_RESULT [SelectApexCore](#) (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT [Process](#) ()
Start processing and return when done.

3.21.1 Detailed Description

Derivative X filter, Hand Tuned (optimized).

Object of this class applies a Derivative X filter to *aSrc*. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.21.2 Member Function Documentation

3.21.2.1 APEXCV_LIB_RESULT apexcv::DerivativeXFilterHT::Initialize (vsdk::SUMat & *aSrc*, int *aWindowSize*, signed char *aK0*, signed char *aK1*, signed char *aK2*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in	<i>aK0</i>	K0
in	<i>aK1</i>	K1
in	<i>aK2</i>	K2
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_16SC1).

3.21.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.21.2.3 APEXCV_LIB_RESULT apexcv::DerivativeXFilterHT::ReconnectIO (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
<i>in, out</i>	<i>aDst</i>	Destination image buffer (VSDK_CV_16SC1).

3.21.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.21.2.5 APEXCV_LIB_RESULT apexcv::DerivativeXFilterHT::SetK0 (signed char *k0*)

Set K0.

This function allows to change the Input parameter K0

Returns

APEXCV_LIB_RESULT Error code.

3.21.2.6 APEXCV_LIB_RESULT apexcv::DerivativeXFilterHT::SetK1 (signed char *k1*)

Set K1.

This function allows to change the Input parameter K1

Returns

APEXCV_LIB_RESULT Error code.

3.21.2.7 APEXCV_LIB_RESULT apexcv::DerivativeXFilterHT::SetK2 (signed char *k2*)

Set K2.

This function allows to change the Input parameter K2

Returns

APEXCV_LIB_RESULT Error code.

3.22 apexcv::DerivativeYFilterHT Class Reference

Derivative Y filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, signed char aK0, signed char aK1, signed char aK2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SetK0** (signed char k0)
Set K0.
- APEXCV_LIB_RESULT **SetK1** (signed char k1)
Set K1.
- APEXCV_LIB_RESULT **SetK2** (signed char k2)
Set K2.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.22.1 Detailed Description

Derivative Y filter, Hand Tuned (optimized).

Object of this class applies a Derivative Y filter to *aSrc*. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.22.2 Member Function Documentation**3.22.2.1 APEXCV_LIB_RESULT apexcv::DerivativeYFilterHT::Initialize (vsdk::SUMat &aSrc, int aWindowSize, signed char aK0, signed char aK1, signed char aK2, vsdk::SUMat &aDst)**

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in	<i>aK0</i>	K0
in	<i>aK1</i>	K1
in	<i>aK2</i>	K2
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_16SC1).

3.22.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.22.2.3 APEXCV_LIB_RESULT apexcv::DerivativeYFilterHT::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_16SC1).

3.22.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftarrow$ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
---------------------------------	------------------------------------------------------------------

3.22.2.5 APEXCV_LIB_RESULT apexcv::DerivativeYFilterHT::SetK0 (signed char *k0*)

Set K0.

This function allows to change the Input parameter K0

Returns

APEXCV_LIB_RESULT Error code.

3.22.2.6 APEXCV_LIB_RESULT apexcv::DerivativeYFilterHT::SetK1 (signed char *k1*)

Set K1.

This function allows to change the Input parameter K1

Returns

APEXCV_LIB_RESULT Error code.

3.22.2.7 APEXCV_LIB_RESULT apexcv::DerivativeYFilterHT::SetK2 (signed char *k2*)

Set K2.

This function allows to change the Input parameter K2

Returns

APEXCV_LIB_RESULT Error code.

3.23 apexcv::DilateFilter Class Reference

Dilate filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.23.1 Detailed Description

Dilate filter.

Object of this class applies a Dilate filter to *aSrc*. Supported window size: 3 x 3
aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1 and VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.23.2 Member Function Documentation

3.23.2.1 APEXCV_LIB_RESULT apexcv::DilateFilter::Initialize (vsdk::SUMat & *aSrc*, int *aWindowSize*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core.
 To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in	<i>aWindowSize</i>	Window Size (3).
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).

3.23.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.23.2.3 APEXCV_LIB_RESULT apexcv::DilateFilter::ReconnectIO (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1 or VSDK_CV_16SC1).

3.23.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.24 apexcv::ErodeFilter Class Reference

Erode filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.24.1 Detailed Description

Erode filter.

Object of this class applies a Erode filter to aSrc. Supported window size: 3 x 3
aDst and aSrc must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.24.2 Member Function Documentation

3.24.2.1 APEXCV_LIB_RESULT apexcv::ErodeFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source Image buffer (VSDK_CV_8UC1).
in	aWindowSize	Window Size (3).
in, out	aDst	Destination Image buffer (VSDK_CV_8UC1).

3.24.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.24.2.3 APEXCV_LIB_RESULT apexcv::ErodeFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.24.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.25 apexcv::ExtractChannel Class Reference

Channel extract class containing support for extracting a single channel from a multi-channel image.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aChannelIndex, vsdk::SUMat &aDst)
Channel Extract function.
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.25.1 Detailed Description

Channel extract class containing support for extracting a single channel from a multi-channel image.

This class is an interface for using channel extract functions on the host.

3.25.2 Member Function Documentation

3.25.2.1 APEXCV_LIB_RESULT apexcv::ExtractChannel::Initialize (vsdk::SUMat & aSrc, int aChannelIndex, vsdk::SUMat & aDst)

Channel Extract function.

Extracts a channel from a multiple channel image based on its index.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC2, VSDK_CV_8UC3, VSDK_CV_8UC4
in	<i>aChannelIndex</i>	Index of the channel to extract. Starts at 1.
in, out	<i>aDst</i>	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.25.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.25.2.3 APEXCV_LIB_RESULT apexcv::ExtractChannel::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC2, VSDK_CV_8UC3, VSDK_CV_8UC4
in, out	<i>aDst</i>	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.25.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.26 apexcv::GaussianFilter Class Reference

Gaussian filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.26.1 Detailed Description

Gaussian filter.

Object of this class applies a Gaussian filter to *aSrc*. Supported window size: 3x3, 5x5, 7x7 or 9x9

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.26.2 Member Function Documentation

3.26.2.1 APEXCV_LIB_RESULT apexcv::GaussianFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size: 3, 5, 7 or 9
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.26.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.26.2.3 APEXCV_LIB_RESULT apexcv::GaussianFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.26.2.4 APEXCV_LIB_RESULT apexcv::ApexcHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.27 apexcv::Histogram Class Reference

Histogram.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.

- **APEXCV_LIB_RESULT Process ()**

Start processing and return when done.

3.27.1 Detailed Description

Histogram.

Object of this class computes the histogram of the image.

Output dimensions are 256x1 VSDK_CV_32SC1.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.27.2 Member Function Documentation

3.27.2.1 APEXCV_LIB_RESULT apexcv::Histogram::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer 256x1 (VSDK_CV_32SC1).

3.27.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.27.2.3 APEXCV_LIB_RESULT apexcv::Histogram::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<code>in</code>	<code>aSrc</code>	Source image buffer (VSDK_CV_8UC1).
<code>in, out</code>	<code>aDst</code>	Destination image buffer 256x1 (VSDK_CV_32SC1).

3.27.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>aApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------	------------------------------------------------------------------

3.28 apexcv::InsertChannel Class Reference

Channel insert class containing support for inserting a single channel in a multi-channel image.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aChannelIndex, vsdk::SUMat &aDst)
Channel Insert function.
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.28.1 Detailed Description

Channel insert class containing support for inserting a single channel in a multi-channel image.

This class is an interface for using channel insert functions on the host.

3.28.2 Member Function Documentation

3.28.2.1 APEXCV_LIB_RESULT apexcv::InsertChannel::Initialize (vsdk::SUMat & aSrc, int aChannelIndex, vsdk::SUMat & aDst)

Channel Insert function.

Inserts a channel from a multiple channel image based on its index.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC2, VSDK_CV_8UC3, VSDK_CV_8UC4
in	aChannelIndex	Index of the channel to insert. Starts at 1.
in, out	aDst	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.28.2.2 APEXCV_LIB_RESULT apexcv::ApexcHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.28.2.3 APEXCV_LIB_RESULT apexcv::InsertChannel::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC2, VSDK_CV_8UC3, VSDK_CV_8UC4
in, out	aDst	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.28.2.4 APEXCV_LIB_RESULT apexcv::ApexcHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.29 apexcv::IntegralImage Class Reference

Integral Image value.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.29.1 Detailed Description

Integral Image value.

Object of this class computes the sum of the pixel values located above and to the left of a given pixel.

Output dimensions are same as input.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.29.2 Member Function Documentation

3.29.2.1 APEXCV_LIB_RESULT apexcv::IntegralImage::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
<i>in, out</i>	<i>aDst</i>	Destination image buffer (VSDK_CV_32SC1).

3.29.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.29.2.3 APEXCV_LIB_RESULT apexcv::IntegralImage::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
<i>in, out</i>	<i>aDst</i>	Destination image buffer (VSDK_CV_32SC1).

3.29.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.30 apexcv::InterpolationBicubicGrayscale Class Reference

Bicubic Grayscale Interpolation.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, vsdk::SUMat &aOffsetX, vsdk::SUMat &aOffsetY, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aOffsetX, vsdk::SUMat &aOffsetY, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.30.1 Detailed Description

Bicubic Grayscale Interpolation.

Object of this class computes the horizontal cubic interpolation, followed by the vertical cubic interpolation on 4x4 patches.

Output dimensions are same as input dimensions.

Supported input type: VSDK_CV_8UC1, output is of identical type VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.30.2 Member Function Documentation

3.30.2.1 APEXCV_LIB_RESULT apexcv::InterpolationBicubicGrayscale::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aOffsetX, vsdk::SUMat & aOffsetY, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	aOffsetX	Delta image buffer (VSDK_CV_8UC1).
in	aOffsetY	Delta image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.30.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.30.2.3 APEXCV_LIB_RESULT apexcv::InterpolationBicubicGrayscale::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aOffsetX, vsdk::SUMat & aOffsetY, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	aOffsetX	Delta image buffer (VSDK_CV_8UC1).
in	aOffsetY	Delta image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.30.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------------------	------------------------------------------------------------------

3.31 apexcv::InterpolationBilinearGrayscale Class Reference

Bilinear Grayscale Interpolation.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, vsdk::SUMat &aDelta, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDelta, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.31.1 Detailed Description

Bilinear Grayscale Interpolation.

Object of this class computes the horizontal linear interpolation, followed by the vertical linear interpolation.

$$\text{Dst}'(x,y) = \text{Src}(x,y) + ((\text{Src}(x+1,y) - \text{Src}(x,y)) * \text{Delta}[0](x,y) + 128)/256$$

$$\text{Dst}(x,y) = \text{Dst}'(x,y) + ((\text{Dst}'(x,y+1) - \text{Dst}'(x,y)) * \text{Delta}[1](x,y) + 128)/256$$

Output dimensions are same as input dimensions.

Supported input type: VSDK_CV_8UC1, output is of identical type VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.31.2 Member Function Documentation

3.31.2.1 APEXCV_LIB_RESULT apexcv::InterpolationBilinearGrayscale::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDelta, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	aDelta	Delta image buffer (VSDK_CV_8UC2).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.31.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.31.2.3 APEXCV_LIB_RESULT apexcv::InterpolationBilinearGrayscale::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDelta, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	aDelta	Delta image buffer (VSDK_CV_8UC2).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.31.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.32 apexcv::InterpolationLinearGrayscale Class Reference

Linear Grayscale Interpolation.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDeltaX, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDeltaX, vsdk::SUMat &aDst)
Reconnect IO (optional).

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.32.1 Detailed Description

Linear Grayscale Interpolation.

Object of this class computes the horizontal linear interpolation between pixels.

Output dimensions are same as input dimensions.

$Dst(x,y) = Src(x,y) + ((Src(x+1,y) - Src(x,y)) * Delta(x,y) + 128)/256$

Supported input type: VSDK_CV_8UC1, output is of identical type VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.32.2 Member Function Documentation

3.32.2.1 APEXCV_LIB_RESULT apexcv::InterpolationLinearGrayscale::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDeltaX, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	aDeltaX	Delta image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.32.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.32.2.3 APEXCV_LIB_RESULT apexcv::InterpolationLinearGrayscale::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDeltaX, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	aDeltaX	Delta image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.32.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

a↔ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------	------------------------------------------------------------------

3.33 apexcv::Magnitude Class Reference

Magnitude.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.33.1 Detailed Description

Magnitude.

Object of this class computes the magnitude of pixel value of *aSrc1* and *aSrc2* pixel by pixel. Where $aDst = \text{SQRT}(\text{square}(aSrc1) + \text{square}(aSrc2))$

Supported input type: VSDK_CV_16SC1, output type: VSDK_CV_16UC1

Supported width: 128 to 2048 pixels.

3.33.2 Member Function Documentation

3.33.2.1 APEXCV_LIB_RESULT apexcv::Magnitude::Initialize (vsdk::SUMat & *aSrc1*, vsdk::SUMat & *aSrc2*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_16SC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_16UC1).

3.33.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.33.2.3 APEXCV_LIB_RESULT apexcv::Magnitude::ReconnectIO (vsdk::SUMat & *aSrc1*, vsdk::SUMat & *aSrc2*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_16SC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_16UC1).

3.33.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.34 apexcv::Max Class Reference

Max.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.34.1 Detailed Description

Max.

Object of this class returns the pixel-wise max values of two images.

Supported input type: VSDK_CV_8UC1, VSDK_CV_16SC1, output type: VSDK_CV_8UC1, VSDK_CV_16SC1

Supported width: 128 to 2048 pixels.

3.34.2 Member Function Documentation**3.34.2.1 APEXCV_LIB_RESULT apexcv::Max::Initialize (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)**

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.34.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.34.2.3 APEXCV_LIB_RESULT apexcv::Max::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.34.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
--------------------------------------	------------------------------------------------------------------

3.35 apexcv::Mean Class Reference

Mean.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc)
Reconnect IO (optional).
- APEXCV_LIB_RESULT [Process](#) (float &aMean)
Process.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.35.1 Detailed Description

Mean.

Object of this class computes the mean value of the image.
Supported input type: VSDK_CV_8UC1.

3.35.2 Member Function Documentation

3.35.2.1 APEXCV_LIB_RESULT apexcv::Mean::Initialize (vsdk::SUMat & aSrc)

Initialize object (required).

This function initializes the object and connect IO. Process() can be called after that to execute the processing in APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
-----------	-------------	-------------------------------------

3.35.2.2 APEXCV_LIB_RESULT apexcvcv::Mean::Process (float & aMean)

Process.

This function start and wait for kernel to complete, then calculate final mean value from output of kernel.

Returns

APEXCV_LIB_RESULT Error code.

3.35.2.3 APEXCV_LIB_RESULT apexcvcv::ApexcvcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.35.2.4 APEXCV_LIB_RESULT apexcvcv::Mean::ReconnectIO (vsdk::SUMat & aSrc)

Reconnect IO (optional).

This function allows to change the Input images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
----	------	-------------------------------------

3.35.2.5 APEXCV_LIB_RESULT apexcvcv::ApexcvcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.36 apexcv::MeanStddev Class Reference

MeanStddev.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **Process** (float &aMean, float &aStddev)
Process.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.36.1 Detailed Description

MeanStddev.

Object of this class computes the mean and standard deviation value of the image.

Output dimensions are 1x1 VSDK_CV_32SC1.

Supported input type: VSDK_CV_8UC1.

3.36.2 Member Function Documentation

3.36.2.1 APEXCV_LIB_RESULT apexcv::MeanStddev::Initialize (vsdk::SUMat & aSrc)

Initialize object (required).

This function initializes the object and connect IO. Process() can be called after that to execute the processing in APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
----	------	-------------------------------------

3.36.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.36.2.3 APEXCV_LIB_RESULT apexcv::MeanStddev::Process (float & aMean, float & aStddev)

Process.

This function start and wait for kernel to complete, then calculate final mean and standard deviation value from output of kernel.

Returns

APEXCV_LIB_RESULT Error code.

3.36.2.4 APEXCV_LIB_RESULT apexcv::MeanStddev::ReconnectIO (vsdk::SUMat & aSrc)

Reconnect IO (optional).

This function allows to change the Input images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
----	------	-------------------------------------

3.36.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------------------	------------------------------------------------------------------

3.37 apexcv::MedianFilter Class Reference

Median filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.37.1 Detailed Description

Median filter.

Object of this class applies a Median filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5 and 7x7
aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.37.2 Member Function Documentation

3.37.2.1 APEXCV_LIB_RESULT apexcv::MedianFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3 or 5 or 7
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.37.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.37.2.3 APEXCV_LIB_RESULT apexcv::MedianFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.37.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.38 apexcv::MergeChannel Class Reference

Channel merge class containing support for merging multiple single channels images into a single multi-channel image.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aSrc4, vsdk::SUMat &aDst)
Channel Merge function.
- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aDst)
Channel Merge function.
- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Channel Merge function.
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aSrc4, vsdk::SUMat &aDst)

Reconnect IO.

- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aDst)

Reconnect IO.

- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)

Reconnect IO.

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.38.1 Detailed Description

Channel merge class containing support for merging multiple single channels images into a single multi-channel image.

This class is an interface for using channel merge functions on the host.

3.38.2 Member Function Documentation

3.38.2.1 APEXCV_LIB_RESULT apexcv::MergeChannel::Initialize (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aSrc4, vsdk::SUMat &aDst)

Channel Merge function.

Merges a channel from a multiple channel image.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	aSrc2	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	aSrc3	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	aSrc4	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in, out	aDst	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC4

3.38.2.2 APEXCV_LIB_RESULT apexcv::MergeChannel::Initialize (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aSrc3, vsdk::SUMat &aDst)

Channel Merge function.

Merges a channel from a multiple channel image.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	aSrc2	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	aSrc3	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in, out	aDst	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC3

3.38.2.3 APEXCV_LIB_RESULT apexcv::MergeChannel::Initialize (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Channel Merge function.

Merges a channel from a multiple channel image.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	aSrc2	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in, out	aDst	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC2

3.38.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.38.2.5 APEXCV_LIB_RESULT apexcv::MergeChannel::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aSrc3, vsdk::SUMat & aSrc4, vsdk::SUMat & aDst)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
----	-------	--------------------------------------------------------------

Parameters

in	<i>aSrc2</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	<i>aSrc3</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	<i>aSrc4</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in, out	<i>aDst</i>	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC4

3.38.2.6 APEXCV_LIB_RESULT apexcv::MergeChannel::ReconnectIO (vsdk::SUMat & *aSrc1*, vsdk::SUMat & *aSrc2*, vsdk::SUMat & *aSrc3*, vsdk::SUMat & *aDst*)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	<i>aSrc2</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	<i>aSrc3</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in, out	<i>aDst</i>	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC3

3.38.2.7 APEXCV_LIB_RESULT apexcv::MergeChannel::ReconnectIO (vsdk::SUMat & *aSrc1*, vsdk::SUMat & *aSrc2*, vsdk::SUMat & *aDst*)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in	<i>aSrc2</i>	Source memory buffer. Accepted buffer types are VSDK_CV_8UC1
in, out	<i>aDst</i>	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC2

3.38.2.8 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and

can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
--------------------------------------	------------------------------------------------------------------

3.39 apexcv::Min Class Reference

Min.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.39.1 Detailed Description

Min.

Object of this class returns the pixel-wise min values of two images.

Supported input type: VSDK_CV_8UC1, VSDK_CV_16SC1, output type: VSDK_CV_8UC1, VSDK_CV_16SC1

Supported width: 128 to 2048 pixels.

3.39.2 Member Function Documentation

3.39.2.1 APEXCV_LIB_RESULT apexcv::Min::Initialize (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.39.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.39.2.3 APEXCV_LIB_RESULT apexcv::Min::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.39.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>aApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------	------------------------------------------------------------------

3.40 apexcv::Mul Class Reference

Multiplication.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SetScale** (const uint8_t acScale)
Set Scale.
- APEXCV_LIB_RESULT **GetScale** (uint8_t &aScale)
Get Scale.
- APEXCV_LIB_RESULT **SetPolicy** (apexcv::eConvertPolicy aPolicy)
Set Policy type.
- APEXCV_LIB_RESULT **GetPolicy** (apexcv::eConvertPolicy &aPolicy)
Get Policy type.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.40.1 Detailed Description

Multiplication.

Object of this class returns element-wise multiplication between two images then divide by 255 or right shift by a value in range of [0,15] (default is 0). *aDst* can be VSDK_CV_8UC1 only if both source images are VSDK_CV_8UC1 and *aDst* is explicitly set to VSDK_CV_8UC1. It is otherwise VSDK_CV_16SC1.

Supported aSrc1 type: VSDK_CV_8UC1, aSrc1 type: VSDK_CV_8UC1, aDst type: VSDK_CV_8UC1 or
Supported aSrc1 type: VSDK_CV_8UC1, aSrc1 type: VSDK_CV_8UC1, aDst type: VSDK_CV_16SC1 or
Supported aSrc1 type: VSDK_CV_8UC1, aSrc1 type: VSDK_CV_16SC1, aDst type: VSDK_CV_16SC1 or
Supported aSrc1 type: VSDK_CV_16SC1, aSrc1 type: VSDK_CV_16SC1, aDst type: VSDK_CV_16SC1
Supported width: 128 to 2048 pixels.

3.40.2 Member Function Documentation

3.40.2.1 APEXCV_LIB_RESULT apexcv::Mul::GetPolicy (apexcv::eConvertPolicy & aPolicy)

Get Policy type.

This function allows to read the value of the overflow policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	<i>aPolicy</i>	Overflow policy type.
-----	----------------	-----------------------

3.40.2.2 APEXCV_LIB_RESULT apexcv::Mul::GetScale (uint8_t & aScale)

Get Scale.

This function allows to read the scale value.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aScale</i>	Scale amount (a value in range of [0,15] or 255)
----	---------------	--------------------------------------------------

3.40.2.3 APEXCV_LIB_RESULT apexcv::Mul::Initialize (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc1</i>	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	<i>aSrc2</i>	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.40.2.4 APEXCV_LIB_RESULT apexcv::ApexcHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.40.2.5 APEXCV_LIB_RESULT apexcv::Mul::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.40.2.6 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

a↔ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------	------------------------------------------------------------------

3.40.2.7 APEXCV_LIB_RESULT apexcv::Mul::SetPolicy (apexcv::eConvertPolicy aPolicy)

Set Policy type.

This function allows to change the overflow policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aPolicy	Overflow policy type
----	---------	----------------------

3.40.2.8 APEXCV_LIB_RESULT apexcv::Mul::SetScale (const uint8_t acScale)

Set Scale.

This function allows to change the scale value: $(aSrc1 * aSrc2) / 255$ for 255, $(aSrc1 * aSrc2) >> acScale$ for [0,15]

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	acScale	Scale amount (a value in range of [0,15] or 255)
----	---------	--------------------------------------------------

3.41 apexcv::Phase Class Reference

Phase.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.41.1 Detailed Description

Phase.

Object of this class is computes the angles for each pixel and stores this in a VSDK_CV_8UC1 image. Where result is then translated to $0 \leq \text{result} < 2\pi$. Each result value is then mapped to the range 0 to 255 inclusive.

Supported input type: VSDK_CV_16SC1, output type: VSDK_CV_8UC1

Supported width: 128 to 2048 pixels.

3.41.2 Member Function Documentation**3.41.2.1 APEXCV_LIB_RESULT apexcv::Phase::Initialize (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)**

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Horizontal gradient (VSDK_CV_16SC1).
in	aSrc2	Vertical gradient (VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.41.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.41.2.3 APEXCV_LIB_RESULT apexcv::Phase::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Horizontal gradient (VSDK_CV_16SC1).
in	aSrc2	Vertical gradient (VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.41.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.42 apexcv::PrewittXFilter Class Reference

Prewitt X filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.42.1 Detailed Description

Prewitt X filter.

Object of this class applies a Prewitt X filter to *aSrc*. Supported window size: 3 x 3

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.42.2 Member Function Documentation

3.42.2.1 APEXCV_LIB_RESULT apexcv::PrewittXFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.42.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.42.2.3 APEXCV_LIB_RESULT apexcv::PrewittXFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
<i>in, out</i>	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.42.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.43 apexcv::PrewittYFilter Class Reference

Prewitt Y filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.43.1 Detailed Description

Prewitt Y filter.

Object of this class applies a Prewitt Y filter to *aSrc*. Supported window size: 3 x 3
aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.43.2 Member Function Documentation

3.43.2.1 APEXCV_LIB_RESULT apexcv::PrewittYFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.43.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.43.2.3 APEXCV_LIB_RESULT apexcv::PrewittYFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.43.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.44 apexcv::SaturateFilterHT Class Reference

Saturate filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.44.1 Detailed Description

Saturate filter, Hand Tuned (optimized).

Object of this class applies a Saturate filter to *aSrc*. This is a hand tuned (HT) implementation providing faster processing times. *aDst* and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_16SC1, output type: VSDK_CV_8SC1.

Supported width: 128 to 2048 pixels.

3.44.2 Member Function Documentation

3.44.2.1 APEXCV_LIB_RESULT apexcv::SaturateFilterHT::Initialize (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8SC1).

3.44.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.44.2.3 APEXCV_LIB_RESULT apexcv::SaturateFilterHT::ReconnectIO (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_16SC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8SC1).

3.44.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.45 apexcv::ScharrFilter Class Reference

Scharr filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.45.1 Detailed Description

Scharr filter.

Object of this class applies a Scharr filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5
aDst and *aSrc* must have identical dimensions.
 Supported input type: VSDK_CV_8UC1.
 Supported output type: VSDK_CV_8UC1.
 Supported width: 128 to 2048 pixels.

3.45.2 Member Function Documentation

3.45.2.1 APEXCV_LIB_RESULT apexcv::ScharrFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3 or 5
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.45.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.45.2.3 APEXCV_LIB_RESULT apexcv::ScharrFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.45.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.46 apexcv::ScharrXFilter Class Reference

Scharr X filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.46.1 Detailed Description

Scharr X filter.

Object of this class applies a Scharr X filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported output type: VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.46.2 Member Function Documentation**3.46.2.1 APEXCV_LIB_RESULT apexcv::ScharrXFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)**

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3 or 5
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_16SC1).

3.46.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.46.2.3 APEXCV_LIB_RESULT apexcv::ScharrXFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
<i>in, out</i>	<i>aDst</i>	Destination image buffer (VSDK_CV_16SC1).

3.46.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.47 apexcv::ScharrXYFilter Class Reference

Scharr XY filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.47.1 Detailed Description

Scharr XY filter.

Object of this class applies a Scharr X and Y filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported output type: VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.47.2 Member Function Documentation

3.47.2.1 APEXCV_LIB_RESULT apexcv::ScharrXYFilter::Initialize (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3 or 5
in, out	<i>aDstX</i>	Destination Image buffer, X (VSDK_CV_16SC1).
in, out	<i>aDstY</i>	Destination Image buffer, Y (VSDK_CV_16SC1).

3.47.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.47.2.3 APEXCV_LIB_RESULT apexcv::ScharrXYFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDstX, vsdk::SUMat & aDstY)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDstX	Destination image buffer, X (VSDK_CV_16SC1).
in, out	aDstY	Destination image buffer, Y (VSDK_CV_16SC1).

3.47.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.48 apexcv::ScharrYFilter Class Reference

Scharr Y filter.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.48.1 Detailed Description

Scharr Y filter.

Object of this class applies a Scharr Y filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported output type: VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.48.2 Member Function Documentation

3.48.2.1 APEXCV_LIB_RESULT apexcv::ScharrYFilter::Initialize (vsdk::SUMat & *aSrc*, int *aWindowSize*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3 or 5
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_16SC1).

3.48.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.48.2.3 APEXCV_LIB_RESULT apexcv::ScharrYFilter::ReconnectIO (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
<i>in, out</i>	<i>aDst</i>	Destination image buffer (VSDK_CV_16SC1).

3.48.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.49 apexcv::SeparableFilterHT Class Reference

Separable filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, signed char(&aFilterRow)[3], signed char(&aFilterCol)[3], vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, signed char(&aFilterRow)[5], signed char(&aFilterCol)[5], vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT [SetFilterRow](#) (signed char(&aFilterRow)[3])
set Filter Row.
- APEXCV_LIB_RESULT [SetFilterCol](#) (signed char(&aFilterCol)[3])
set Filter Col.
- APEXCV_LIB_RESULT [SetFilterRow](#) (signed char(&aFilterRow)[5])
set Filter Row.
- APEXCV_LIB_RESULT [SetFilterCol](#) (signed char(&aFilterCol)[5])

set Filter Column.

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.49.1 Detailed Description

Separable filter, Hand Tuned (optimized).

Object of this class applies a Separable filter to *aSrc*. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3 and 5 x 5

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.49.2 Member Function Documentation

3.49.2.1 APEXCV_LIB_RESULT apexcv::SeparableFilterHT::Initialize (vsdk::SUMat & *aSrc*, signed char(&) *aFilterRow*[3], signed char(&) *aFilterCol*[3], vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aFilterRow</i>	3x1 Horizontal filter parameters
in	<i>aFilterCol</i>	1x3 Vertical filter parameters
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_16SC1).

3.49.2.2 APEXCV_LIB_RESULT apexcv::SeparableFilterHT::Initialize (vsdk::SUMat & *aSrc*, signed char(&) *aFilterRow*[5], signed char(&) *aFilterCol*[5], vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aFilterRow</i>	5x1 Horizontal filter parameters
in	<i>aFilterCol</i>	1x5 Vertical filter parameters
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_16SC1).

3.49.2.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.49.2.4 APEXCV_LIB_RESULT apexcv::SeparableFilterHT::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_16SC1).

3.49.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.49.2.6 APEXCV_LIB_RESULT apexcv::SeparableFilterHT::SetFilterCol (signed char(&) aFilterCol[3])

set Filter Col.

This function allows to change filter coefficients.

Returns

APEXCV_LIB_RESULT Error code.[in] 1x3 Vertical filter parameters

3.49.2.7 APEXCV_LIB_RESULT apexcv::SeparableFilterHT::SetFilterCol (signed char(&) aFilterCol[5])

set Filter Column.

This function allows to change filter coefficients.

Returns

APEXCV_LIB_RESULT Error code.[in] 1x5 Vertical filter parameters

3.49.2.8 APEXCV_LIB_RESULT apexcv::SeparableFilterHT::SetFilterRow (signed char(&) aFilterRow[3])

set Filter Row.

This function allows to change filter coefficients.

Returns

APEXCV_LIB_RESULT Error code.[in] 3x1 Horizontal filter parameters

3.49.2.9 APEXCV_LIB_RESULT apexcv::SeparableFilterHT::SetFilterRow (signed char(&) aFilterRow[5])

set Filter Row.

This function allows to change filter coefficients.

Returns

APEXCV_LIB_RESULT Error code.[in] 5x1 Horizontal filter parameters

3.50 apexcv::SobelFilter Class Reference

Sobel filter.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.50.1 Detailed Description

Sobel filter.

Object of this class applies a Sobel filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.50.2 Member Function Documentation

3.50.2.1 APEXCV_LIB_RESULT apexcvcv::SobelFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3 or 5
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.50.2.2 APEXCV_LIB_RESULT apexcvcv::ApexcvcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.50.2.3 APEXCV_LIB_RESULT apexcvcv::SobelFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.50.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.51 apexcv::SobelFilterHT Class Reference

Sobel filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.51.1 Detailed Description

Sobel filter, Hand Tuned (optimized).

Object of this class applies a Sobel filter to aSrc. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1.

Supported width: 128 to 2048 pixels.

3.51.2 Member Function Documentation

3.51.2.1 APEXCV_LIB_RESULT apexcv::SobelFilterHT::Initialize (vsdk::SUMat & *aSrc*, int *aWindowSize*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
<i>aWindowSize</i>	Window Size, 3 or 5
<i>aDst</i>	Destination Image buffer (VSDK_CV_16SC1).

3.51.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.51.2.3 APEXCV_LIB_RESULT apexcv::SobelFilterHT::ReconnectIO (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_16SC1).

3.51.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.52 apexcv::SobelXFilter Class Reference

Sobel X filter.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.52.1 Detailed Description

Sobel X filter.

Object of this class applies a Sobel X filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5
aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.52.2 Member Function Documentation

3.52.2.1 APEXCV_LIB_RESULT apexcv::SobelXFilter::Initialize (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source Image buffer (VSDK_CV_8UC1).
in	aWindowSize	Window Size, 3 or 5
in, out	aDst	Destination Image buffer (VSDK_CV_8UC1).

3.52.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.52.2.3 APEXCV_LIB_RESULT apexcv::SobelXFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.52.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.53 apexcv::SobelXFilterHT Class Reference

Sobel X filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.53.1 Detailed Description

Sobel X filter, Hand Tuned (optimized).

Object of this class applies a Sobel X filter to *aSrc*. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8SC1.

Supported width: 128 to 2048 pixels.

3.53.2 Member Function Documentation**3.53.2.1 APEXCV_LIB_RESULT apexcv::SobelXFilterHT::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)**

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8SC1).

3.53.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.53.2.3 APEXCV_LIB_RESULT apexcv::SobelXFilterHT::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8SC1).

3.53.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
--------------------------	------------------------------------------------------------------

3.54 apexcv::SobelXYFilter Class Reference

Sobel XY filter.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDstX, vsdk::SUMat &aDstY)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.54.1 Detailed Description

Sobel XY filter.

Object of this class applies a Sobel X and Y filter to *aSrc*. Supported window size: 3 x 3
aDst and *aSrc* must have identical dimensions.
 Supported input type: VSDK_CV_8UC1.
 Supported width: 128 to 2048 pixels.

3.54.2 Member Function Documentation

3.54.2.1 APEXCV_LIB_RESULT apexcv::SobelXYFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDstX, vsdk::SUMat & aDstY)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in, out	<i>aDstX</i>	Destination Image buffer, X (VSDK_CV_8UC1).
in	<i>aDstY</i>	Destination Image buffer, Y (VSDK_CV_8UC1).

3.54.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.54.2.3 APEXCV_LIB_RESULT apexcv::SobelXYFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDstX, vsdk::SUMat & aDstY)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDstX	Destination image buffer (VSDK_CV_8UC1).
in, out	aDstY	Destination image buffer (VSDK_CV_8UC1).

3.54.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.55 apexcv::SobelYFilter Class Reference

Sobel Y filter.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).

- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)

Select the APEX Core.

- APEXCV_LIB_RESULT **Process** ()

Start processing and return when done.

3.55.1 Detailed Description

Sobel Y filter.

Object of this class applies a Sobel Y filter to *aSrc*. Supported window size: 3 x 3 and 5 x 5

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1.

Supported width: 128 to 2048 pixels.

3.55.2 Member Function Documentation

3.55.2.1 APEXCV_LIB_RESULT apexcv::SobelYFilter::Initialize (vsdk::SUMat & aSrc, int aWindowSize, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3 or 5
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8UC1).

3.55.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.55.2.3 APEXCV_LIB_RESULT apexcv::SobelYFilter::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.55.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.56 apexcv::SobelYFilterHT Class Reference

Sobel Y filter, Hand Tuned (optimized).

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, int aWindowSize, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.56.1 Detailed Description

Sobel Y filter, Hand Tuned (optimized).

Object of this class applies a Sobel Y filter to aSrc. This is a hand tuned (HT) implementation providing faster processing times. Supported window size: 3 x 3

aDst and *aSrc* must have identical dimensions.

Supported input type: VSDK_CV_8UC1, output type: VSDK_CV_8SC1.

Supported width: 128 to 2048 pixels.

3.56.2 Member Function Documentation

3.56.2.1 APEXCV_LIB_RESULT apexcv::SobelYFilterHT::Initialize (vsdk::SUMat & *aSrc*, int *aWindowSize*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source Image buffer (VSDK_CV_8UC1).
in	<i>aWindowSize</i>	Window Size, 3
in, out	<i>aDst</i>	Destination Image buffer (VSDK_CV_8SC1).

3.56.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.56.2.3 APEXCV_LIB_RESULT apexcv::SobelYFilterHT::ReconnectIO (vsdk::SUMat & *aSrc*, vsdk::SUMat & *aDst*)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8SC1).

3.56.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int *aApexId*) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a↔ ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
----------------------	------------------------------------------------------------------

3.57 apexcv::SplitChannel Class Reference

Channel split class containing support for splitting a single channel from a multi-channel image.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::SUMat &aDst3, vsdk::SUMat &aDst4)
Channel Split function.
- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::SUMat &aDst3)
Channel Split function.
- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2)
Channel Split function.
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::↔SUMat &aDst3, vsdk::SUMat &aDst4)
Reconnect IO.
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2, vsdk::↔SUMat &aDst3)
Reconnect IO.
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst1, vsdk::SUMat &aDst2)
Reconnect IO.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.57.1 Detailed Description

Channel split class containing support for splitting a single channel from a multi-channel image.

This class is an interface for using channel split functions on the host.

3.57.2 Member Function Documentation

3.57.2.1 APEXCV_LIB_RESULT apexcv::SplitChannel::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst1, vsdk::SUMat & aDst2, vsdk::SUMat & aDst3, vsdk::SUMat & aDst4)

Channel Split function.

Splits a channel from a multiple channel image.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC4
in, out	aDst1	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst2	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst3	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst4	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.57.2.2 APEXCV_LIB_RESULT apexcv::SplitChannel::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst1, vsdk::SUMat & aDst2, vsdk::SUMat & aDst3)

Channel Split function.

Splits a channel from a multiple channel image.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC3
in, out	aDst1	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst2	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst3	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.57.2.3 APEXCV_LIB_RESULT apexcv::SplitChannel::Initialize (vsdk::SUMat & aSrc, vsdk::SUMat & aDst1, vsdk::SUMat & aDst2)

Channel Split function.

Splits a channel from a multiple channel image.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC2
in, out	aDst1	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
	aDst2	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.57.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.57.2.5 APEXCV_LIB_RESULT apexcv::SplitChannel::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst1, vsdk::SUMat & aDst2, vsdk::SUMat & aDst3, vsdk::SUMat & aDst4)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC4
in, out	aDst1	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst2	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst3	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst4	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.57.2.6 APEXCV_LIB_RESULT apexcv::SplitChannel::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst1, vsdk::SUMat & aDst2, vsdk::SUMat & aDst3)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC3
in, out	aDst1	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst2	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst3	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.57.2.7 APEXCV_LIB_RESULT apexcvcv::SplitChannel::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst1, vsdk::SUMat & aDst2)

Reconnect IO.

This function allows to change the Input and Output images without re-initializing

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source memory buffer. Accepted buffer types are VSDK_CV_8UC2
in, out	aDst1	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1
in, out	aDst2	Destination memory buffer. Accepted buffer type is VSDK_CV_8UC1

3.57.2.8 APEXCV_LIB_RESULT apexcvcv::ApexcvcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.58 apexcvcv::Subtract Class Reference

Subtract.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc1, vsdk::SUMat &aSrc2, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SetPolicy** (apexcv::eConvertPolicy aPolicy)
Set Policy type.
- APEXCV_LIB_RESULT **GetPolicy** (apexcv::eConvertPolicy &aPolicy)
Get Policy type.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.58.1 Detailed Description

Subtract.

Object of this class subtracts the pixel value of *aSrc2* from *aSrc1* pixel by pixel. *aDst* can be VSDK_CV_8UC1 only if both source images are VSDK_CV_8UC1 and *aDst* is explicitly set to VSDK_CV_8UC1. It is otherwise VSDK_CV_16SC1.

Supported input 1 type: VSDK_CV_8UC1, input 2 type: VSDK_CV_8UC1, output type: VSDK_CV_8UC1 or
Supported input 1 type: VSDK_CV_8UC1, input 2 type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1 or
Supported input 1 type: VSDK_CV_8UC1, input 2 type: VSDK_CV_16SC1, output type: VSDK_CV_16SC1 or
Supported input 1 type: VSDK_CV_16SC1, input 2 type: VSDK_CV_8UC1, output type: VSDK_CV_16SC1 or
Supported input 1 type: VSDK_CV_16SC1, input 2 type: VSDK_CV_16SC1, output type: VSDK_CV_16SC1
Supported width: 128 to 2048 pixels.

3.58.2 Member Function Documentation

3.58.2.1 APEXCV_LIB_RESULT apexcv::Subtract::GetPolicy (apexcv::eConvertPolicy & aPolicy)

Get Policy type.

This function allows to read the value of the overflow policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	<i>aPolicy</i>	Overflow policy type.
-----	----------------	-----------------------

3.58.2.2 APEXCV_LIB_RESULT apexcv::Subtract::Initialize (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core.

To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.58.2.3 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.58.2.4 APEXCV_LIB_RESULT apexcv::Subtract::ReconnectIO (vsdk::SUMat & aSrc1, vsdk::SUMat & aSrc2, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc1	Source image buffer 1 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in	aSrc2	Source image buffer 2 (VSDK_CV_8UC1, VSDK_CV_16SC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1, VSDK_CV_16SC1).

3.58.2.5 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.58.2.6 APEXCV_LIB_RESULT apexcv::Subtract::SetPolicy (apexcv::eConvertPolicy aPolicy)

Set Policy type.

This function allows to change the overflow policy type.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

<i>in</i>	<i>aPolicy</i>	Overflow policy type
-----------	----------------	----------------------

3.59 apexcv::TableLookup Class Reference

Table Lookup.

Public Member Functions

- APEXCV_LIB_RESULT **Initialize** (vsdk::SUMat &aSrc, vsdk::SUMat &acLut, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT **ReconnectIO** (vsdk::SUMat &aSrc, vsdk::SUMat &acLut, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.59.1 Detailed Description

Table Lookup.

Object of this class translates the pixel value of *aSrc* through the lookup table *acLut* pixel by pixel.

Supported input type: VSDK_CV_8UC1

Supported width: 128 to 2048 pixels.

3.59.2 Member Function Documentation**3.59.2.1 APEXCV_LIB_RESULT apexcv::TableLookup::Initialize (vsdk::SUMat &aSrc, vsdk::SUMat &acLut, vsdk::SUMat &aDst)**

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	acLut	Look-up table for the transformation (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.59.2.2 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.59.2.3 APEXCV_LIB_RESULT apexcv::TableLookup::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & acLut, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in	acLut	Look-up table for the transformation (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.59.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
--------------------------------------	------------------------------------------------------------------

3.60 apexcv::Threshold Class Reference

Threshold.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, const uint32_t aThreshold, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT [SetThreshold](#) (const uint32_t acThreshold)
Set Threshold.
- APEXCV_LIB_RESULT [GetThreshold](#) (uint32_t &aThreshold)
Get Threshold.
- APEXCV_LIB_RESULT [SetOutputValues](#) (const uint8_t acTrueVal, const uint8_t acFalseVal)
Set Output Values.
- APEXCV_LIB_RESULT [GetOutputValues](#) (uint8_t &aTrueVal, uint8_t &aFalseVal)
Get Output Values.
- APEXCV_LIB_RESULT [SelectApexCore](#) (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT [Process](#) ()
Start processing and return when done.

3.60.1 Detailed Description

Threshold.

Object of this class thresholds the pixel value of *aSrc* with the value of *aThreshold* pixel by pixel.

True when $aSrc(x,y) > aThreshold$, otherwise false.

Default output values are 255 when true and 0 when false.

Supported input type: VSDK_CV_8UC1

Supported width: 128 to 2048 pixels.

3.60.2 Member Function Documentation

3.60.2.1 APEXCV_LIB_RESULT apexcv::Threshold::GetOutputValues (uint8_t & aTrueVal, uint8_t & aFalseVal)

Get Output Values.

This function allows to read the low and high output values.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	<i>aTrueVal</i>	true output value.
out	<i>aFalseVal</i>	false output value.

3.60.2.2 APEXCV_LIB_RESULT apexcv::Threshold::GetThreshold (uint32_t & aThreshold)

Get Threshold.

This function allows to read the value of the threshold.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	<i>aThreshold</i>	threshold.
-----	-------------------	------------

3.60.2.3 APEXCV_LIB_RESULT apexcv::Threshold::Initialize (vsdk::SUMat & aSrc, const uint32_t aThreshold, vsdk::SUMat & aDst)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source image buffer (VSDK_CV_8UC1).
in	<i>aThreshold</i>	Threshold value.
in, out	<i>aDst</i>	Destination image buffer (VSDK_CV_8UC1).

3.60.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on

a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.60.2.5 APEXCV_LIB_RESULT apexcv::Threshold::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	aSrc	Source image buffer (VSDK_CV_8UC1).
in, out	aDst	Destination image buffer (VSDK_CV_8UC1).

3.60.2.6 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

$a \leftrightarrow$ ApexId	ID of the APEX core used for performing the processing (0 or 1).
-------------------------------	------------------------------------------------------------------

3.60.2.7 APEXCV_LIB_RESULT apexcv::Threshold::SetOutputValues (const uint8_t acTrueVal, const uint8_t acFalseVal)

Set Output Values.

This function allows to change the low and high output values.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	acTrueVal	true output value.
----	-----------	--------------------

Parameters

in	<i>acFalseVal</i>	false output value.
----	-------------------	---------------------

3.60.2.8 APEXCV_LIB_RESULT apexcv::Threshold::SetThreshold (const uint32_t *acThreshold*)

Set Threshold.

This function allows to change the value of the threshold.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>acThreshold</i>	threshold.
----	--------------------	------------

3.61 apexcv::ThresholdRange Class Reference

Threshold Range.

Public Member Functions

- APEXCV_LIB_RESULT [Initialize](#) (vsdk::SUMat &aSrc, const uint32_t acLowThreshold, const uint32_t acHighThreshold, vsdk::SUMat &aDst)
Initialize object (required).
- APEXCV_LIB_RESULT [ReconnectIO](#) (vsdk::SUMat &aSrc, vsdk::SUMat &aDst)
Reconnect IO (optional).
- APEXCV_LIB_RESULT [SetThresholds](#) (const uint32_t acLowThreshold, const uint32_t acHighThreshold)
Set Thresholds.
- APEXCV_LIB_RESULT [GetThresholds](#) (uint32_t &aLowThreshold, uint32_t &aHighThreshold)
Get Thresholds.
- APEXCV_LIB_RESULT [SetOutputValues](#) (const uint8_t acTrueVal, const uint8_t acFalseVal)
SetOutputValues.
- APEXCV_LIB_RESULT [GetOutputValues](#) (uint8_t &aTrueVal, uint8_t &aFalseVal)
Get Output Values.
- APEXCV_LIB_RESULT **SelectApexCore** (int aApexId)
Select the APEX Core.
- APEXCV_LIB_RESULT **Process** ()
Start processing and return when done.

3.61.1 Detailed Description

Threshold Range.

Object of this class thresholds the pixel value of *aSrc* with the following scheme pixel by pixel.

False when $aSrc(x,y) > acHighThreshold$, False when $aSrc(x,y) < acLowThreshold$, otherwise true.

Default output values are 255 when true and 0 when false.

Supported input type: VSDK_CV_8UC1

Supported width: 128 to 2048 pixels.

3.61.2 Member Function Documentation

3.61.2.1 APEXCV_LIB_RESULT apexcv::ThresholdRange::GetOutputValues (uint8_t & *aTrueVal*, uint8_t & *aFalseVal*)

Get Output Values.

This function allows to read the true and false output values.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	<i>aTrueVal</i>	true output value.
out	<i>aFalseVal</i>	false output value.

3.61.2.2 APEXCV_LIB_RESULT apexcv::ThresholdRange::GetThresholds (uint32_t & *aLowThreshold*, uint32_t & *aHighThreshold*)

Get Thresholds.

This function allows to read the value of the thresholds.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

out	<i>aLowThreshold</i>	low threshold.
out	<i>aHighThreshold</i>	high threshold.

3.61.2.3 APEXCV_LIB_RESULT apexcv::ThresholdRange::Initialize (vsdk::SUMat & *aSrc*, const uint32_t *acLowThreshold*, const uint32_t *acHighThreshold*, vsdk::SUMat & *aDst*)

Initialize object (required).

This function initializes the object. The function Process() can be called to execute the processing on the APEX core. To process another image buffer, use ReconnectIO(...).

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source memory buffer.
in	<i>acLowThreshold</i>	Low Threshold value.
in	<i>acHighThreshold</i>	High Threshold value.
in, out	<i>aDst</i>	Destination memory buffer.

3.61.2.4 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::Process () [inherited]

Start processing and return when done.

Execute code on selected APEX core (default is Apex core 0). This function is called after initialize() and is executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

3.61.2.5 APEXCV_LIB_RESULT apexcv::ThresholdRange::ReconnectIO (vsdk::SUMat & aSrc, vsdk::SUMat & aDst)

Reconnect IO (optional).

This function allows to change the Input and Output images without re-initializing.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>aSrc</i>	Source memory buffer.
in, out	<i>aDst</i>	Destination memory buffer.

3.61.2.6 APEXCV_LIB_RESULT apexcv::ApexcvHostBaseClass::SelectApexCore (int aApexId) [inherited]

Select the APEX Core.

Select which APEX core (0 or 1) to be selected to run the processing. This function has to be called after initialize() and can be executed on a per frame base.

Returns

APEXCV Error code (APEXCV_SUCCESS on success).

Parameters

<i>a</i> ↔ <i>ApexId</i>	ID of the APEX core used for performing the processing (0 or 1).
-----------------------------	------------------------------------------------------------------

3.61.2.7 APEXCV_LIB_RESULT apexcv::ThresholdRange::SetOutputValues (const uint8_t *acTrueVal*, const uint8_t *acFalseVal*)

SetOutputValues.

This function allows to change the true and false output values.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>acTrueVal</i>	true output value.
in	<i>acFalseVal</i>	false output value.

3.61.2.8 APEXCV_LIB_RESULT apexcv::ThresholdRange::SetThresholds (const uint32_t *acLowThreshold*, const uint32_t *acHighThreshold*)

Set Thresholds.

This function allows to change the value of the thresholds.

Returns

APEXCV_LIB_RESULT Error code.

Parameters

in	<i>acLowThreshold</i>	low threshold.
in	<i>acHighThreshold</i>	high threshold.

Bibliography

- [1] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846, Jan 1998. [19](#)

Index

apexcv::Abs, 6
 Initialize, 6
 Process, 7
 ReconnectIO, 7
 SelectApexCore, 7
 apexcv::AbsDiff, 8
 Initialize, 8
 Process, 8
 ReconnectIO, 9
 SelectApexCore, 9
 apexcv::Accumulate, 9
 Initialize, 10
 Process, 10
 ReconnectIO, 10
 SelectApexCore, 11
 apexcv::AccumulateSquared, 11
 GetScale, 12
 Initialize, 12
 Process, 12
 ReconnectIO, 12
 SelectApexCore, 13
 SetScale, 13
 apexcv::AccumulateWeighted, 13
 GetAlpha, 14
 Initialize, 14
 Process, 15
 ReconnectIO, 15
 SelectApexCore, 15
 SetAlpha, 15
 apexcv::Add, 16
 GetPolicy, 17
 Initialize, 17
 Process, 17
 ReconnectIO, 17
 SelectApexCore, 18
 SetPolicy, 18
 apexcv::BilateralFilter, 18
 Initialize, 19
 Process, 19
 ReconnectIO, 20
 SelectApexCore, 20
 SetSigmaColor, 20
 SetSigmaSpace, 21
 apexcv::BitwiseAND, 21
 Initialize, 22
 Process, 22
 ReconnectIO, 22
 SelectApexCore, 22
 apexcv::BitwiseNOT, 23
 Initialize, 23
 Process, 24
 ReconnectIO, 24
 SelectApexCore, 24
 apexcv::BitwiseOR, 25
 Initialize, 25
 Process, 25
 ReconnectIO, 26
 SelectApexCore, 26
 apexcv::BitwiseXOR, 26
 Initialize, 27
 Process, 27
 ReconnectIO, 28
 SelectApexCore, 28
 apexcv::BoxFilter, 28
 Initialize, 29
 Process, 29
 ReconnectIO, 29
 SelectApexCore, 30
 apexcv::BoxFilterHT, 30
 Initialize, 31
 Process, 31
 ReconnectIO, 31
 SelectApexCore, 31
 apexcv::CensusFilter, 32
 Initialize, 32
 Process, 33
 ReconnectIO, 33
 SelectApexCore, 33
 apexcv::Clz, 34
 Initialize, 34
 Process, 35
 ReconnectIO, 35
 SelectApexCore, 35
 apexcv::ColorConverter, 35
 ConversionType, 36
 eBGR888_TO_GREY, 36
 eGREY_TO_RGB888, 36
 eRGB565_TO_RGB888X, 36
 eRGB888_TO_GREY, 36
 eRGB888X_TO_RGB565, 36

eRGB888X_TO_YUV, 36
 eRGB888X_TO_Y, 36
 Initialize, 37
 Process, 37
 ReconnectIO, 38
 SelectApexCore, 38
 SetFactors, 38
 apexcv::ColorConverterHT, 39
 ConversionType, 39
 eHT_RGB888X_TO_Y, 39
 Initialize, 40
 Process, 40
 ReconnectIO, 40
 SelectApexCore, 41
 SetFactors, 41
 apexcv::ConvertDepth, 41
 GetPolicyType, 42
 GetShift, 42
 Initialize, 42
 Process, 43
 ReconnectIO, 43
 SelectApexCore, 43
 SetPolicyType, 44
 SetShift, 44
 apexcv::ConvolveFilter, 44
 Initialize, 45
 Process, 46
 ReconnectIO, 46
 SelectApexCore, 46
 SetFilterCoeff, 47
 SetFilterScale, 47
 apexcv::ConvolveFilterHT, 48
 Initialize, 49
 Process, 49
 ReconnectIO, 49
 SelectApexCore, 50
 SetFilterCoeff, 50
 SetFilterScale, 51
 apexcv::DerivativeXFilterHT, 51
 Initialize, 52
 Process, 52
 ReconnectIO, 52
 SelectApexCore, 53
 SetK0, 53
 SetK1, 53
 SetK2, 53
 apexcv::DerivativeYFilterHT, 54
 Initialize, 54
 Process, 55
 ReconnectIO, 55
 SelectApexCore, 55
 SetK0, 56
 SetK1, 56
 SetK2, 56
 apexcv::DilateFilter, 56
 Initialize, 57
 Process, 57
 ReconnectIO, 57
 SelectApexCore, 58
 apexcv::ErodeFilter, 58
 Initialize, 59
 Process, 59
 ReconnectIO, 59
 SelectApexCore, 59
 apexcv::ExtractChannel, 60
 Initialize, 60
 Process, 61
 ReconnectIO, 61
 SelectApexCore, 61
 apexcv::GaussianFilter, 62
 Initialize, 62
 Process, 62
 ReconnectIO, 63
 SelectApexCore, 63
 apexcv::Histogram, 63
 Initialize, 64
 Process, 64
 ReconnectIO, 64
 SelectApexCore, 65
 apexcv::InsertChannel, 65
 Initialize, 66
 Process, 66
 ReconnectIO, 66
 SelectApexCore, 66
 apexcv::IntegrallImage, 67
 Initialize, 67
 Process, 68
 ReconnectIO, 68
 SelectApexCore, 68
 apexcv::InterpolationBicubicGrayscale, 69
 Initialize, 69
 Process, 69
 ReconnectIO, 70
 SelectApexCore, 70
 apexcv::InterpolationBilinearGrayscale, 70
 Initialize, 71
 Process, 71
 ReconnectIO, 72
 SelectApexCore, 72
 apexcv::InterpolationLinearGrayscale, 72
 Initialize, 73
 Process, 73
 ReconnectIO, 73
 SelectApexCore, 74
 apexcv::Magnitude, 74
 Initialize, 75
 Process, 75
 ReconnectIO, 75

SelectApexCore, 76	SelectApexCore, 97
apexcv::Max, 76	apexcv::SaturateFilterHT, 97
Initialize, 76	Initialize, 98
Process, 77	Process, 98
ReconnectIO, 77	ReconnectIO, 98
SelectApexCore, 77	SelectApexCore, 99
apexcv::Mean, 78	apexcv::ScharrFilter, 99
Initialize, 78	Initialize, 99
Process, 78, 79	Process, 100
ReconnectIO, 79	ReconnectIO, 100
SelectApexCore, 79	SelectApexCore, 100
apexcv::MeanStddev, 80	apexcv::ScharrXFilter, 101
Initialize, 80	Initialize, 101
Process, 80, 81	Process, 102
ReconnectIO, 81	ReconnectIO, 102
SelectApexCore, 81	SelectApexCore, 102
apexcv::MedianFilter, 81	apexcv::ScharrXYFilter, 102
Initialize, 82	Initialize, 103
Process, 82	Process, 103
ReconnectIO, 82	ReconnectIO, 104
SelectApexCore, 83	SelectApexCore, 104
apexcv::MergeChannel, 83	apexcv::ScharrYFilter, 104
Initialize, 84, 85	Initialize, 105
Process, 85	Process, 105
ReconnectIO, 85, 86	ReconnectIO, 105
SelectApexCore, 86	SelectApexCore, 106
apexcv::Min, 87	apexcv::SeparableFilterHT, 106
Initialize, 87	Initialize, 107
Process, 88	Process, 108
ReconnectIO, 88	ReconnectIO, 108
SelectApexCore, 88	SelectApexCore, 108
apexcv::Mul, 89	SetFilterCol, 109
GetPolicy, 89	SetFilterRow, 109
GetScale, 90	apexcv::SobelFilter, 109
Initialize, 90	Initialize, 110
Process, 90	Process, 110
ReconnectIO, 91	ReconnectIO, 110
SelectApexCore, 91	SelectApexCore, 111
SetPolicy, 91	apexcv::SobelFilterHT, 111
SetScale, 92	Initialize, 112
apexcv::Phase, 92	Process, 112
Initialize, 92	ReconnectIO, 112
Process, 93	SelectApexCore, 112
ReconnectIO, 93	apexcv::SobelXFilter, 113
SelectApexCore, 93	Initialize, 113
apexcv::PrewittXFilter, 94	Process, 114
Initialize, 94	ReconnectIO, 114
Process, 95	SelectApexCore, 114
ReconnectIO, 95	apexcv::SobelXFilterHT, 115
SelectApexCore, 95	Initialize, 115
apexcv::PrewittYFilter, 95	Process, 116
Initialize, 96	ReconnectIO, 116
Process, 96	SelectApexCore, 116
ReconnectIO, 96	apexcv::SobelXYFilter, 116

Initialize, 117
 Process, 117
 ReconnectIO, 118
 SelectApexCore, 118
 apexcv::SobelYFilter, 118
 Initialize, 119
 Process, 119
 ReconnectIO, 119
 SelectApexCore, 120
 apexcv::SobelYFilterHT, 120
 Initialize, 121
 Process, 121
 ReconnectIO, 121
 SelectApexCore, 121
 apexcv::SplitChannel, 122
 Initialize, 123
 Process, 124
 ReconnectIO, 124, 125
 SelectApexCore, 125
 apexcv::Subtract, 125
 GetPolicy, 126
 Initialize, 126
 Process, 127
 ReconnectIO, 127
 SelectApexCore, 127
 SetPolicy, 129
 apexcv::TableLookup, 129
 Initialize, 129
 Process, 130
 ReconnectIO, 130
 SelectApexCore, 130
 apexcv::Threshold, 131
 GetOutputValues, 131
 GetThreshold, 132
 Initialize, 132
 Process, 132
 ReconnectIO, 133
 SelectApexCore, 133
 SetOutputValues, 133
 SetThreshold, 134
 apexcv::ThresholdRange, 134
 GetOutputValues, 135
 GetThresholds, 135
 Initialize, 135
 Process, 136
 ReconnectIO, 136
 SelectApexCore, 136
 SetOutputValues, 137
 SetThresholds, 137

 ConversionType
 apexcv::ColorConverter, 36
 apexcv::ColorConverterHT, 39

 eBGR888_TO_GREY
 apexcv::ColorConverter, 36
 eGREY_TO_RGB888
 apexcv::ColorConverter, 36
 eHT_RGB888X_TO_Y
 apexcv::ColorConverterHT, 39
 eRGB565_TO_RGB888X
 apexcv::ColorConverter, 36
 eRGB888_TO_GREY
 apexcv::ColorConverter, 36
 eRGB888X_TO_RGB565
 apexcv::ColorConverter, 36
 eRGB888X_TO_YUV
 apexcv::ColorConverter, 36
 eRGB888X_TO_Y
 apexcv::ColorConverter, 36

 GetAlpha
 apexcv::AccumulateWeighted, 14
 GetOutputValues
 apexcv::Threshold, 131
 apexcv::ThresholdRange, 135
 GetPolicy
 apexcv::Add, 17
 apexcv::Mul, 89
 apexcv::Subtract, 126
 GetPolicyType
 apexcv::ConvertDepth, 42
 GetScale
 apexcv::AccumulateSquared, 12
 apexcv::Mul, 90
 GetShift
 apexcv::ConvertDepth, 42
 GetThreshold
 apexcv::Threshold, 132
 GetThresholds
 apexcv::ThresholdRange, 135

 Initialize
 apexcv::Abs, 6
 apexcv::AbsDiff, 8
 apexcv::Accumulate, 10
 apexcv::AccumulateSquared, 12
 apexcv::AccumulateWeighted, 14
 apexcv::Add, 17
 apexcv::BilateralFilter, 19
 apexcv::BitwiseAND, 22
 apexcv::BitwiseNOT, 23
 apexcv::BitwiseOR, 25
 apexcv::BitwiseXOR, 27
 apexcv::BoxFilter, 29
 apexcv::BoxFilterHT, 31
 apexcv::CensusFilter, 32
 apexcv::Clz, 34
 apexcv::ColorConverter, 37
 apexcv::ColorConverterHT, 40

apexcv::ConvertDepth, 42
 apexcv::ConvolveFilter, 45
 apexcv::ConvolveFilterHT, 49
 apexcv::DerivativeXFilterHT, 52
 apexcv::DerivativeYFilterHT, 54
 apexcv::DilateFilter, 57
 apexcv::ErodeFilter, 59
 apexcv::ExtractChannel, 60
 apexcv::GaussianFilter, 62
 apexcv::Histogram, 64
 apexcv::InsertChannel, 66
 apexcv::IntegrallImage, 67
 apexcv::InterpolationBicubicGrayscale, 69
 apexcv::InterpolationBilinearGrayscale, 71
 apexcv::InterpolationLinearGrayscale, 73
 apexcv::Magnitude, 75
 apexcv::Max, 76
 apexcv::Mean, 78
 apexcv::MeanStddev, 80
 apexcv::MedianFilter, 82
 apexcv::MergeChannel, 84, 85
 apexcv::Min, 87
 apexcv::Mul, 90
 apexcv::Phase, 92
 apexcv::PrewittXFilter, 94
 apexcv::PrewittYFilter, 96
 apexcv::SaturateFilterHT, 98
 apexcv::ScharrFilter, 99
 apexcv::ScharrXFilter, 101
 apexcv::ScharrXYFilter, 103
 apexcv::ScharrYFilter, 105
 apexcv::SeparableFilterHT, 107
 apexcv::SobelFilter, 110
 apexcv::SobelFilterHT, 112
 apexcv::SobelXFilter, 113
 apexcv::SobelXFilterHT, 115
 apexcv::SobelXYFilter, 117
 apexcv::SobelYFilter, 119
 apexcv::SobelYFilterHT, 121
 apexcv::SplitChannel, 123
 apexcv::Subtract, 126
 apexcv::TableLookup, 129
 apexcv::Threshold, 132
 apexcv::ThresholdRange, 135

Process

apexcv::Abs, 7
 apexcv::AbsDiff, 8
 apexcv::Accumulate, 10
 apexcv::AccumulateSquared, 12
 apexcv::AccumulateWeighted, 15
 apexcv::Add, 17
 apexcv::BilateralFilter, 19
 apexcv::BitwiseAND, 22

apexcv::BitwiseNOT, 24
 apexcv::BitwiseOR, 25
 apexcv::BitwiseXOR, 27
 apexcv::BoxFilter, 29
 apexcv::BoxFilterHT, 31
 apexcv::CensusFilter, 33
 apexcv::Clz, 35
 apexcv::ColorConverter, 37
 apexcv::ColorConverterHT, 40
 apexcv::ConvertDepth, 43
 apexcv::ConvolveFilter, 46
 apexcv::ConvolveFilterHT, 49
 apexcv::DerivativeXFilterHT, 52
 apexcv::DerivativeYFilterHT, 55
 apexcv::DilateFilter, 57
 apexcv::ErodeFilter, 59
 apexcv::ExtractChannel, 61
 apexcv::GaussianFilter, 62
 apexcv::Histogram, 64
 apexcv::InsertChannel, 66
 apexcv::IntegrallImage, 68
 apexcv::InterpolationBicubicGrayscale, 69
 apexcv::InterpolationBilinearGrayscale, 71
 apexcv::InterpolationLinearGrayscale, 73
 apexcv::Magnitude, 75
 apexcv::Max, 77
 apexcv::Mean, 78, 79
 apexcv::MeanStddev, 80, 81
 apexcv::MedianFilter, 82
 apexcv::MergeChannel, 85
 apexcv::Min, 88
 apexcv::Mul, 90
 apexcv::Phase, 93
 apexcv::PrewittXFilter, 95
 apexcv::PrewittYFilter, 96
 apexcv::SaturateFilterHT, 98
 apexcv::ScharrFilter, 100
 apexcv::ScharrXFilter, 102
 apexcv::ScharrXYFilter, 103
 apexcv::ScharrYFilter, 105
 apexcv::SeparableFilterHT, 108
 apexcv::SobelFilter, 110
 apexcv::SobelFilterHT, 112
 apexcv::SobelXFilter, 114
 apexcv::SobelXFilterHT, 116
 apexcv::SobelXYFilter, 117
 apexcv::SobelYFilter, 119
 apexcv::SobelYFilterHT, 121
 apexcv::SplitChannel, 124
 apexcv::Subtract, 127
 apexcv::TableLookup, 130
 apexcv::Threshold, 132
 apexcv::ThresholdRange, 136

ReconnectIO

apexcv::Abs, 7
 apexcv::AbsDiff, 9
 apexcv::Accumulate, 10
 apexcv::AccumulateSquared, 12
 apexcv::AccumulateWeighted, 15
 apexcv::Add, 17
 apexcv::BilateralFilter, 20
 apexcv::BitwiseAND, 22
 apexcv::BitwiseNOT, 24
 apexcv::BitwiseOR, 26
 apexcv::BitwiseXOR, 28
 apexcv::BoxFilter, 29
 apexcv::BoxFilterHT, 31
 apexcv::CensusFilter, 33
 apexcv::Clz, 35
 apexcv::ColorConverter, 38
 apexcv::ColorConverterHT, 40
 apexcv::ConvertDepth, 43
 apexcv::ConvolveFilter, 46
 apexcv::ConvolveFilterHT, 49
 apexcv::DerivativeXFilterHT, 52
 apexcv::DerivativeYFilterHT, 55
 apexcv::DilateFilter, 57
 apexcv::ErodeFilter, 59
 apexcv::ExtractChannel, 61
 apexcv::GaussianFilter, 63
 apexcv::Histogram, 64
 apexcv::InsertChannel, 66
 apexcv::IntegrallImage, 68
 apexcv::InterpolationBicubicGrayscale, 70
 apexcv::InterpolationBilinearGrayscale, 72
 apexcv::InterpolationLinearGrayscale, 73
 apexcv::Magnitude, 75
 apexcv::Max, 77
 apexcv::Mean, 79
 apexcv::MeanStddev, 81
 apexcv::MedianFilter, 82
 apexcv::MergeChannel, 85, 86
 apexcv::Min, 88
 apexcv::Mul, 91
 apexcv::Phase, 93
 apexcv::PrewittXFilter, 95
 apexcv::PrewittYFilter, 96
 apexcv::SaturateFilterHT, 98
 apexcv::ScharrFilter, 100
 apexcv::ScharrXFilter, 102
 apexcv::ScharrXYFilter, 104
 apexcv::ScharrYFilter, 105
 apexcv::SeparableFilterHT, 108
 apexcv::SobelFilter, 110
 apexcv::SobelFilterHT, 112
 apexcv::SobelXFilter, 114
 apexcv::SobelXFilterHT, 116

apexcv::SobelXYFilter, 118
 apexcv::SobelYFilter, 119
 apexcv::SobelYFilterHT, 121
 apexcv::SplitChannel, 124, 125
 apexcv::Subtract, 127
 apexcv::TableLookup, 130
 apexcv::Threshold, 133
 apexcv::ThresholdRange, 136

SelectApexCore

apexcv::Abs, 7
 apexcv::AbsDiff, 9
 apexcv::Accumulate, 11
 apexcv::AccumulateSquared, 13
 apexcv::AccumulateWeighted, 15
 apexcv::Add, 18
 apexcv::BilateralFilter, 20
 apexcv::BitwiseAND, 22
 apexcv::BitwiseNOT, 24
 apexcv::BitwiseOR, 26
 apexcv::BitwiseXOR, 28
 apexcv::BoxFilter, 30
 apexcv::BoxFilterHT, 31
 apexcv::CensusFilter, 33
 apexcv::Clz, 35
 apexcv::ColorConverter, 38
 apexcv::ColorConverterHT, 41
 apexcv::ConvertDepth, 43
 apexcv::ConvolveFilter, 46
 apexcv::ConvolveFilterHT, 50
 apexcv::DerivativeXFilterHT, 53
 apexcv::DerivativeYFilterHT, 55
 apexcv::DilateFilter, 58
 apexcv::ErodeFilter, 59
 apexcv::ExtractChannel, 61
 apexcv::GaussianFilter, 63
 apexcv::Histogram, 65
 apexcv::InsertChannel, 66
 apexcv::IntegrallImage, 68
 apexcv::InterpolationBicubicGrayscale, 70
 apexcv::InterpolationBilinearGrayscale, 72
 apexcv::InterpolationLinearGrayscale, 74
 apexcv::Magnitude, 76
 apexcv::Max, 77
 apexcv::Mean, 79
 apexcv::MeanStddev, 81
 apexcv::MedianFilter, 83
 apexcv::MergeChannel, 86
 apexcv::Min, 88
 apexcv::Mul, 91
 apexcv::Phase, 93
 apexcv::PrewittXFilter, 95
 apexcv::PrewittYFilter, 97
 apexcv::SaturateFilterHT, 99

apexcv::ScharrFilter, 100
 apexcv::ScharrXFilter, 102
 apexcv::ScharrXYFilter, 104
 apexcv::ScharrYFilter, 106
 apexcv::SeparableFilterHT, 108
 apexcv::SobelFilter, 111
 apexcv::SobelFilterHT, 112
 apexcv::SobelXFilter, 114
 apexcv::SobelXFilterHT, 116
 apexcv::SobelXYFilter, 118
 apexcv::SobelYFilter, 120
 apexcv::SobelYFilterHT, 121
 apexcv::SplitChannel, 125
 apexcv::Subtract, 127
 apexcv::TableLookup, 130
 apexcv::Threshold, 133
 apexcv::ThresholdRange, 136
 SetAlpha
 apexcv::AccumulateWeighted, 15
 SetFactors
 apexcv::ColorConverter, 38
 apexcv::ColorConverterHT, 41
 SetFilterCoeff
 apexcv::ConvolveFilter, 47
 apexcv::ConvolveFilterHT, 50
 SetFilterCol
 apexcv::SeparableFilterHT, 109
 SetFilterRow
 apexcv::SeparableFilterHT, 109
 SetFilterScale
 apexcv::ConvolveFilter, 47
 apexcv::ConvolveFilterHT, 51
 SetK0
 apexcv::DerivativeXFilterHT, 53
 apexcv::DerivativeYFilterHT, 56
 SetK1
 apexcv::DerivativeXFilterHT, 53
 apexcv::DerivativeYFilterHT, 56
 SetK2
 apexcv::DerivativeXFilterHT, 53
 apexcv::DerivativeYFilterHT, 56
 SetOutputValues
 apexcv::Threshold, 133
 apexcv::ThresholdRange, 137
 SetPolicy
 apexcv::Add, 18
 apexcv::Mul, 91
 apexcv::Subtract, 129
 SetPolicyType
 apexcv::ConvertDepth, 44
 SetScale
 apexcv::AccumulateSquared, 13
 apexcv::Mul, 92
 SetShift
 apexcv::ConvertDepth, 44
 SetSigmaColor
 apexcv::BilateralFilter, 20
 SetSigmaSpace
 apexcv::BilateralFilter, 21
 SetThreshold
 apexcv::Threshold, 134
 SetThresholds
 apexcv::ThresholdRange, 137