

Huawei HiAI DDK Operator Specifications

1. Parameter Description

Parameter	Description
ni	Indicates the size of a batch.
ci	feature dimension, e.g. for RGB images ci=3 Indicates the number of channels.
hi/ho/	Indicates the height.
wi/wo	Indicates the width.
pad_h(pad_y)/ pad_w(pad_x)	Indicates the number of pixels to be padded. By default, the padding size is less than the kernel size.
stride_h(stride_y, shift_h)/ stride_w(stride_x, shift_w)	specifies the intervals at which to apply the filters to the input Indicates the step.
kernel_h(kernel_y)/ kernel_w(kernel_x)	specifies height and width of each filter Indicates the size of a convolution core.
window_h(window_y)/ window_w(window_x)	Indicates a window size.
zoom/shrink	Indicates the zoom-in or zoom-out multiple.

2. Operator Boundary

2.1 Caffe Operator Boundary

No.	Operator	Boundary
1	AvgPool/MaxPool	$stride_h \leq window_h; stride_w \leq window_w$ Note: better to make $window_h < =hi, window_w \leq wi$ $xo \times window_h + xo \leq 2000$ $wi + xo \leq 2000$ $xo = (wi + pad_w - window_w) / stride_w + 1;$ pad != 0: $pad_w < kw$ $pad_h < kh$ $(pad_w + wi) \times ci < 8000$

2	interp	$w_i \leq 10000$ $h_o \leq 10000$ $w_o \leq 2500$ $h_i \leq 12000$ $c_i \geq 256$: $4 \times w_o + w_i + h_o < 1000$ $C_i < 256$: $(4w_o + w_i + h_o) \times (c_i + 15) \leq 256000$
3	ABSVAL	<p>The result of $n_i \times c_i \times h_i \times w_i$ is affected by the memory size.</p> <p>The precision is limited. The value range is [-10, 10].</p>
4	Convolution	$kernel_h \leq 32$ $kernel_w \leq 32$ $n_i \leq 512$ $c_i \leq 3200$ $w_i \leq 5400000$ $h_i \leq 5400000$ $pad = 0$: <p>The result of $n_i \times h_i \times w_i \times (c_i + c_o)$ is affected by the memory size.</p> $pad \neq 0$: $pad_w < kw$ $pad_h < kh$ $(pad_w + w_i) \times c_i < 8000$ <p>The result of $2n_i \times (h_i + pad_h) \times (w_i + pad_w) \times (c_i + c_o)$ is affected by the memory size.</p> <p>When groups exist: $group \geq 2$ $(c_i / group) \% 16 = 0$ $(c_o / group) \% 16 = 0$ <p>The result of $n_i \times (h_i + pad_h) \times (w_i + pad_w) \times (c_i + c_o)$ is affected by the memory size.</p> </p>
5	RELU1	<p>The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.</p>

		The precision is limited. The value range is [-10, 10].
6	scale	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
7	greater	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
8	lessEqual	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
9	RELU6	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size. The precision is limited. The value range is [-10, 10].
10	exp	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
11	l2_pooling	$pool1_stride_h \leq pool1_window_h \leq h_i$ $pool1_stride_w \leq pool1_window_w \leq w_i$ $x_o \times window_h + x_o \leq 2000$ $w_i + x_o < 2000$ $x_o = (w_i + pad_w - window_w) / stride_w + 1$; $pad \neq 0$: $pad_w < k_w$ $pad_h < k_h$ $(pad_w + w_i) \times c_i < 8000$
12	concat_by_feature	$n_i < 128$ The input c value for image stitching must be the same. The value is a multiple of 16. A maximum of eight tensors can participate in image stitching at a time. $n_i \times h_i \times w_i < 4096 \times 4096$
13	deconvolution	$\max(stride_h, stride_w) \times n_o \leq 256$ $c_i < 256$ $co < 256$ Group parameters are not supported. $pad \neq 0$: $pad_w < k_w$ $pad_h < k_h$ $(pad_w + w_i) \times c_i < 8000$
14	SIGMOID	The result of $n_i \times c_i \times h_i \times w_i$ is

		restricted by the memory size. The precision is limited. The value range is [-10, 10].
15	add	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
16	mult	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
17	RELU	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size. The precision is limited. The value range is [-10, 10].
18	Sub	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
19	floor	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
20	select	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
21	svdf	$num_units < 256$
22	cast	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size. The following data types can be converted: CAST_FLOAT32_TO_UINT8 CAST_UINT8_TO_FLOAT32 CAST_INT8_TO_FLOAT16 CAST_FIX8_TO_FLOAT16 CAST_FLOAT16_TO_FIX8 CAST_FLOAT16_TO_FLOAT32 CAST_FLOAT32_TO_FLOAT16 CAST_INT16_TO_FLOAT16 CAST_FLOAT16_TO_INT16
23	conv_depthwise	$kernel_h \leq 32$ $kernel_w \leq 32$ $n_i \leq 512$ $w_i \leq 5350000$ $h_i \leq 5400000$ $c_i \times m \leq 640$ $n_i \times c_i \times h_i \times w_i \geq 8$ $c_i \times m \times k_h \times k_w \geq 8$ pad = 0:

		<p>The result of $n_i \times h_i \times w_i \times (c_i + c_o)$ is affected by the memory size.</p> <p>pad != 0: $pad_w < kw$ $pad_h < kh$ $(pad_w + w_i) \times c_i < 8000$</p> <p>The result of $2n_i \times (h_i + pad_h) \times (w_i + pad_w) \times (c_i + c_o)$ is affected by the memory size.</p>
24	LRN	$c_i \leq 256$
25	Elu	<p>The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.</p> <p>The precision is limited. The value range is $[-10, 10]$.</p>
26	Exp	<p>The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.</p> <p>The precision is limited. The value range is $[-10, 10]$.</p>
27	Log	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
28	Lstm_unit	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
29	normalize	$kernel_h < = 32$ $kernel_w < = 32$ $n_i < = 512$ $c_i < = 3200$ $w_i < = 5400000$ $h_i < = 5400000$ pad = 0: <p>The result of $n_i \times h_i \times w_i \times (c_i + c_o)$ is affected by the memory size.</p> <p>pad != 0: $pad_w < kw$ $pad_h < kh$ $(pad_w + w_i) \times c_i < 8000$</p> <p>The result of $2n_i \times (h_i + pad_h) \times (w_i + pad_w) \times (c_i + c_o)$ is affected by the memory size.</p>
30	power	<p>The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.</p> <p>$y = power(x)$ is implemented by $e^{(y \times \ln x)}$. The precision is affected by exp</p>

		and ln.
31	greptensor	no+shift_n <=ni ho+shift_h <=hi wo+shift_w <=wi wo x fo <=12000
32	InnerProduct	if (ni>1&&(hi>1 wi>1) Same as convolution else; Restricted by the memory size
33	batchnorm	The result of ni x ci x hi x wi is restricted by the memory size.
34	Eltwise	The result of ni x ci x hi x wi is restricted by the memory size.
35	Sqrt	The result of ni x ci x hi x wi is restricted by the memory size.
36	tanh	The result of ni x ci x hi x wi is restricted by the memory size.
37	Slice	The result of ni x ci x hi x wi is restricted by the memory size.
38	Silence	The result of ni x ci x hi x wi is restricted by the memory size.

2.2 Tensorflow Operator Boundary

No.	Python API	C++ API	Boundary
1	tf.nn.avg_pool	AvgPool	stride_h <= window_h; stride_w <= window_w Note: better to make window_h <=hi,window_w <=wi xo x window_h+xo <=2000 wi+xo <=2000 xo=(wi+pad_w-window_w)/stride_w+1; pad != 0: pad_w < kw pad_h < kh (pad_w+wi) x ci < 8000
2	tf.nn.max_pool	MaxPool	stride_h <= window_h; stride_w <=

			<p>window_w</p> <p>Note:</p> <p>better to make window_h <=hi, window_w <=wi</p> <p>$xo \times window_h + xo \leq 2000$</p> <p>$wi + xo \leq 2000$</p> <p>$xo = (wi + pad_w - window_w) / stride_w + 1$;</p> <p>pad != 0:</p> <p>pad_w < kw</p> <p>pad_h < kh</p> <p>$(pad_w + wi) \times ci < 8000$</p>
3	tf.image.resize_images (ResizeMethod.BILINEAR)	ResizeBilinear	<p>$wi \leq 10000$</p> <p>$ho \leq 10000$</p> <p>$wo \leq 2500$</p> <p>$hi \leq 12000$</p> <p>ci>=256:</p> <p>$4 \times wo + wi + ho < 1000$</p> <p>ci < 256:</p> <p>$(4wo + wi + ho) \times (ci + 15) \leq 256000$</p>
4	tf.image.resize_images (ResizeMethod.NEAREST_NEIGHBOR)	ResizeNearestNeighbor	<p>$wi \leq 10000$</p> <p>$ho \leq 10000$</p> <p>$wo \leq 2500$</p> <p>$hi \leq 12000$</p> <p>ci>=256:</p> <p>$4 \times wo + wi + ho < 1000$</p> <p>ci < 256:</p> <p>$(4wo + wi + ho) \times (ci + 15) \leq 256000$</p>
5	tf.abs	Abs	<p>The result of ni x ci x hi x wi is restricted by the memory size.</p> <p>The precision is limited. The value range is [-10, 10].</p>
6	tf.nn.conv2d	Conv2D	<p>kernel_h <=32</p> <p>kernel_w <=32</p> <p>ni <=512</p> <p>ci <=3200</p> <p>wi <=5400000</p> <p>hi <=5400000</p> <p>pad = 0:</p> <p>The result of ni x hi x wi x (ci+co) is affected by the memory size.</p> <p>pad != 0:</p> <p>pad_w < kw</p>

			<p>$pad_h < kh$ $(pad_w+wi) \times ci < 8000$ The result of $2ni \times (hi+pad_h) \times (wi+pad_w) \times (ci+co)$ is affected by the memory size.</p>
7	tf.greater	Greater	The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.
8	tf.less_equal	LessEqual	The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.
9	tf.nn.relu	Relu	<p>The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.</p> <p>The precision is limited. The value range is [-10, 10].</p>
10	tf.nn.relu6	Relu6	<p>The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.</p> <p>The precision is limited. The value range is [-10, 10].</p>
11	tf.contrib.keras.layers.LeakyReLU	/	<p>The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.</p> <p>The precision is limited. The value range is [-10, 10].</p>
12	tf.exp	Exp	The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.
13	tf.concat	Concat	<p>1. Input restrictions: $ni < 128$, $ni \times hi \times wi < 4096 \times 4096$</p> <p>2. $values[i].dim(axis)==16 \times$ (integer multiple of 16)</p>
14	tf.nn.conv2d_backprop_input	Conv2DBackpropInput	$max(stride_h, stride_w) \times num_output \leq 256$
15	tf.sigmoid	Sigmoid	<p>The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.</p> <p>The precision is limited. The value range is [-10, 10].</p>
16	tf.add	Add	The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.
17	tf.add_n	AddN	The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.
18	tf.multiply	Multiply	The result of $ni \times ci \times hi \times wi$ is restricted by the memory size.

19	tf.subtract	Subtract	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
20	tf.matmul	MatMul	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
21	tf.nn.bias_add	BiasAdd	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
s22	tf.nn.fused_batch_norm	FusedBatchNorm	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
23	tf.nn.lrn	LRN	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
24	tf.where	Select	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
25	tf.summary.merge	Merge	Not restricted
26	tf.nn.elu	Elu	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size. The precision is limited. The value range is [-10, 10].
27	tf.rsqrt	Rsqrt	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size. The precision is limited. The value range is [-10, 10].
28	tf.exp	Exp	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
29	tf.log	Log	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
30	tf.tanh	Tanh	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
31	tf.slice	Slice	1. Input restrictions: $n_i < 128$, $n_i \times h_i \times w_i < 4096 \times 4096$ 2. Parameter restrictions: $\text{begin}[i] == 16x$, $\text{size}[i] == 16x$
32	tf.contrib.layers.flatten	Flatten	If IPU operators exist before and after the flatten API, the operators that either directly or indirectly depend on the output of the flatten API are considered as activation operators (such as relu, sigmoid, and tanh) or MatMul operators.
33	tf.split	Split	Input restrictions: $n_i < 128$, $n_i \times h_i \times w_i < 4096 \times 4096$ Parameter restrictions:

			<p>2.1. Currently, num_or_size_splits can be only an integer, instead of an array.</p> <p>2.2. value.dim(axis) ==num_or_size_splits x 16x</p>
34	tf.nn.depthwise_conv2d	DepthwiseConv2dNative	<p>kernel_h <=32 kernel_w <=32 ni <=512 wi <=5350000 hi <=5400000 ci x m <=640 ni x ci x hi x wi >=8 ci x m x kh x kw >=8</p> <p>pad = 0: The result of ni x hi x wi x (ci+co) is affected by the memory size.</p> <p>pad != 0: pad_w < kw pad_h < kh (pad_w+wi) x ci < 8000 The result of 2ni x (hi+pad_h) x (wi+pad_w) x (ci+co) is affected by the memory size.</p>
35	tf.cast	Cast	<p>The result of ni x ci x hi x wi is restricted by the memory size.</p> <p>The following data types can be converted:</p> <p>CAST_FLOAT32_TO_UINT8 CAST_UINT8_TO_FLOAT32 CAST_INT8_TO_FLOAT16 CAST_FIX8_TO_FLOAT16 CAST_FLOAT16_TO_FIX8 CAST_FLOAT16_TO_FLOAT32 CAST_FLOAT32_TO_FLOAT16 CAST_INT16_TO_FLOAT16 CAST_FLOAT16_TO_INT16</p>
36	tf.floor	Floor	The result of ni x ci x hi x wi is restricted by the memory size.
37	tf.contrib.keras.backend.switch	Switch	Not restricted
38	tf.identity	Identity	Not restricted
39	tf.nn.softplus	Softplus	The result of ni x ci x hi x wi is restricted

			by the memory size.
40	tf.nn.softsign	Softsign	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
41	tf.pad	Pad, PadV2	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
42	tf.contrib.rnn.LSTMCell	/	Parameter restrictions: $num_units == 16x$ && $num_proj == 16x$
43	tf.contrib.rnn.GRUCell	/	Parameter restriction: $num_units == 16x$
44	tf.contrib.rnn.GRUCell	GRUCell	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.
45	tf.contrib.rnn.LSTMCell	LSTMCell	The result of $n_i \times c_i \times h_i \times w_i$ is restricted by the memory size.