

CNN

Operating system	Linux
Image preprocessing	<ul style="list-style-type: none"> • Demosaicing • AOI • Resolution scaling • CNN preprocessing modes <ul style="list-style-type: none"> ○ zero255 ○ zero200 • CNN preprocessing modes from <code>keras.applications.imagenet_utils.preprocess_input</code> <ul style="list-style-type: none"> ○ tf ○ caffe ○ torch
Data postprocessing on CPU	<ul style="list-style-type: none"> • Softmax operation for classification • Box regression for object detection • User programmable processing on Dual Core ARM Cortex-A53

deep ocean core

Input data formats	<ul style="list-style-type: none"> • Up to 512 x 512 pixels • Tensor of shape (x, y, 3) for color • Tensor of shape (x, y, 1) for mono
Output data formats	<ul style="list-style-type: none"> • Float • Shape formatting can be defined in post-processing (depending on model and task)
Internal number format	<ul style="list-style-type: none"> • 16-bit fixed point • Integer and fractional bits individually optimized for each layer
Maximum number of layers in model	<ul style="list-style-type: none"> • 180 layers
Maximum model size	Any size fitting the RAM size (maximum 245 MB (1,6 MP mono), minimum 160 MB (6 MP color))
Supported operations/layers	<ul style="list-style-type: none"> • 2D convolution layer • Depthwise separable convolution layer • Average pooling layer • Max. pooling layer • Dense layer • Add layer • Concatenate layer

Subject to technical modifications (2021-10-27)

	<ul style="list-style-type: none"> • Squeeze-and-Excitation layer • ReLU activation • ReLU6 activation • Swish activation • Sigmoid activation • Batch normalization
Filter kernel/pooling parameters	<ul style="list-style-type: none"> • Kernel/pooling window: any rectangle up to 15 x 15 pixels • Kernel depth: any depth • Kernel/pooling stride: up to 15 pixels
Inference time	See Benchmark

Benchmark

Reference models from tensorflow.keras.applications

Architecture	Input format → output format	Single image inference time [ms]
MobileNet V1 =1.0	(224, 224, 3) → (1000)	66
MobileNet V1 =0.75	(224, 224, 3) → (1000)	50
MobileNet V1 =0.5	(224, 224, 3) → (1000)	34
MobileNet V1 =0.5	(128, 128, 3) → (1000)	16
MobileNet V1 =0.25	(224, 224, 3) → (1000)	28
MobileNet V1 =0.25	(128, 128, 3) → (1000)	12
MobileNet V2 =1.4	(224, 224, 3) → (1000)	102
MobileNet V2 =1.0	(224, 224, 3) → (1000)	71
MobileNet V2 =0.5	(224, 224, 3) → (1000)	47
MobileNet V3 large alpha 1.0	(224, 224, 3) → (1000)	73
MobileNet V3 large alpha 0.75	(224, 224, 3) → (1000)	65
MobileNet V3 small alpha 1.0	(224, 224, 3) → (1000)	33
ResNet50	(224, 224, 3) → (1000)	297
Xception	(224, 224, 3) → (1000)	596
MobileNet_V1_SSD	(300, 300, 3) → (scores: (3323, 81), boxes: (3323, 4), anchors: (3323, 4))	132

Subject to technical modifications (2021-10-27)

IDS NXT ferry 1.1.0

Supported model formats	tensorflow.keras .h5 models
Compatible Keras/Tensorflow versions	Tensorflow 2.3
Supported CNNs	<p>Tested CNNs from tensorflow.keras.applications</p> <ul style="list-style-type: none">• MobileNet• MobileNet V2• MobileNet V3• ResNet50 <p>The support of further published CNNs, like SqueezeNet or self-created CNNs is possible, as long as the restrictions listed under CNN are considered.</p> <p>"Single Shot Detector" architectures based on MobileNet are used for object detection.</p>