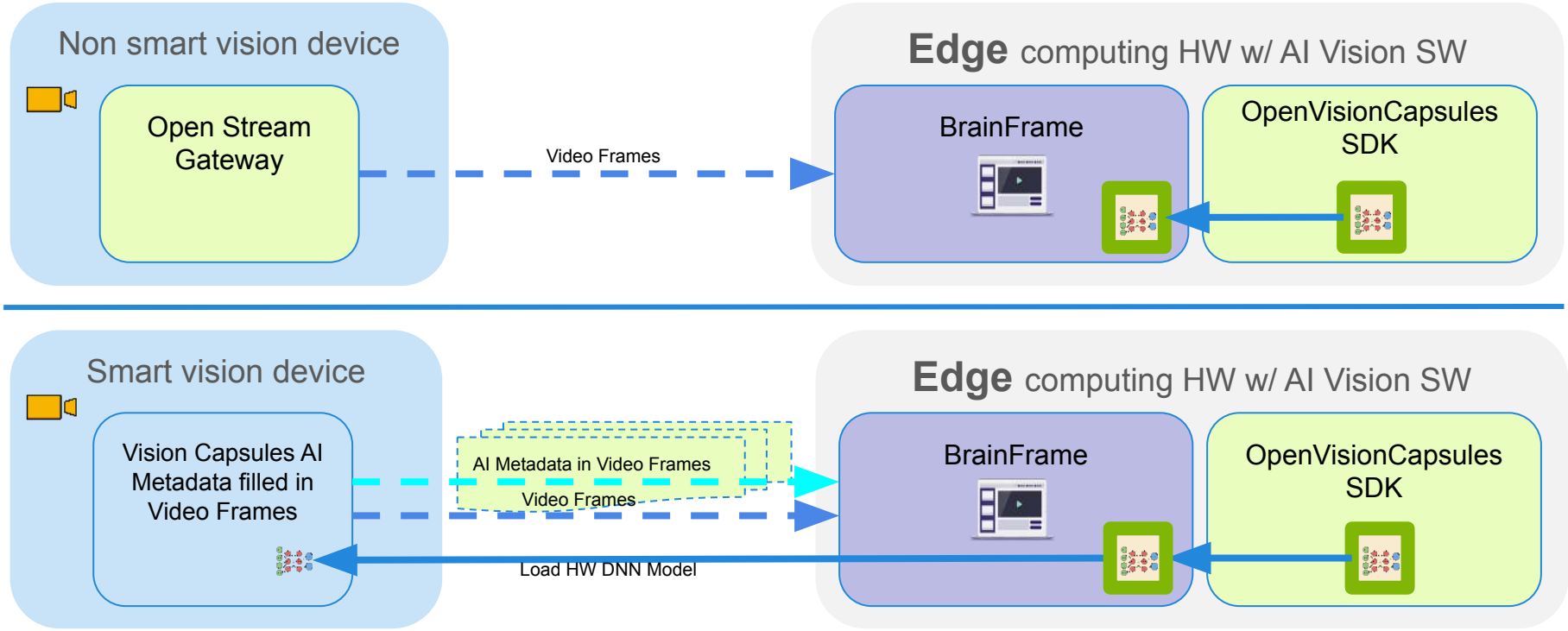
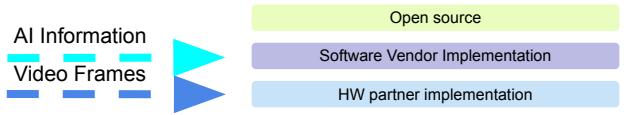
How to build an OpenVisionCapsules Compatible Hardware

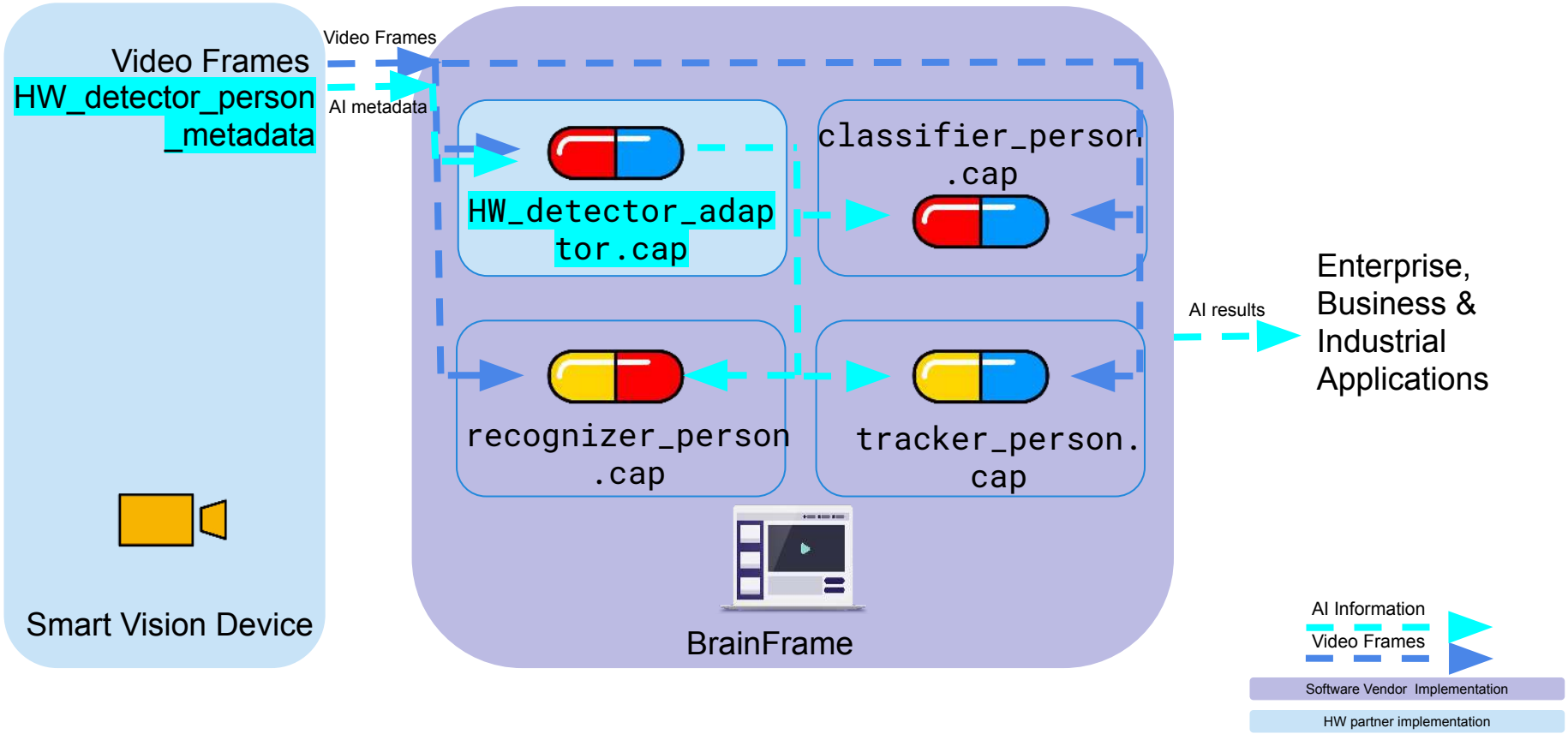
HW VisionCapsules System Architecture



BrainFrame can be download from https://dililil abs.com/docs/downloads/. It is offered by Dilili Labs as an AI Software Vendor.



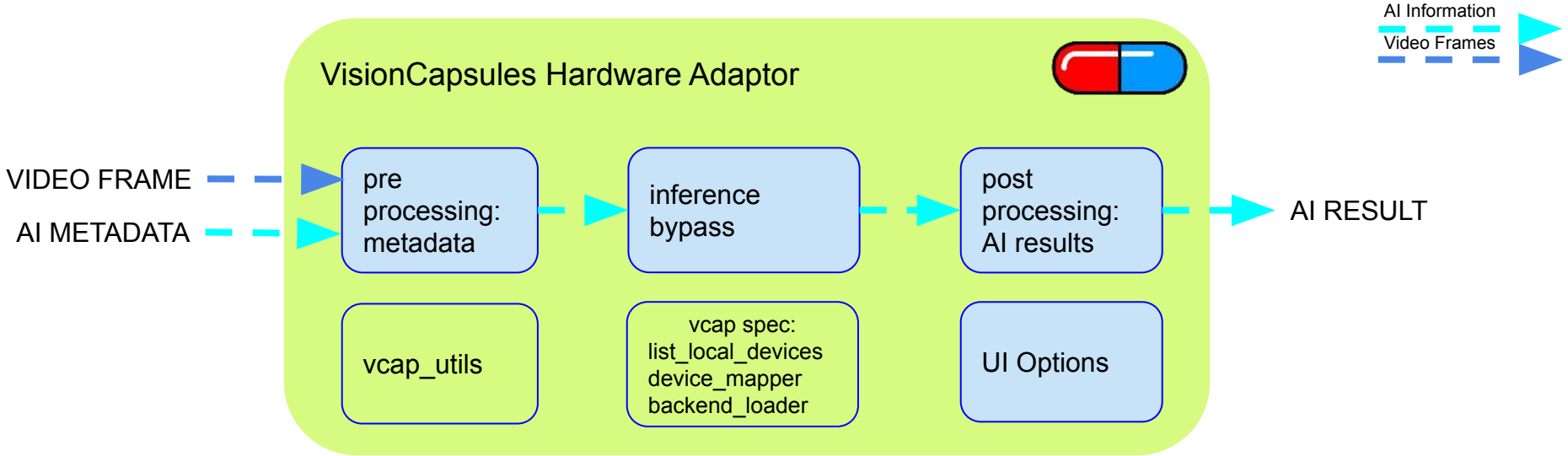
HW VisionCapsules Adaptor Architecture



VisionCapsules Hardware Adaptor Implementation

OpenVisionCapsules Opensource

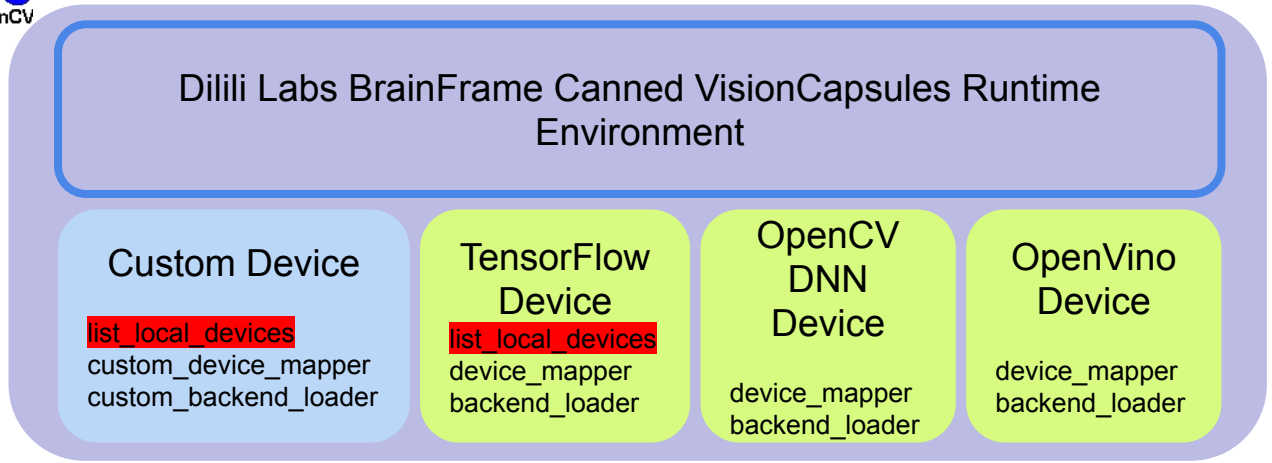
HW partner Implementation



VisionCapsules Source code can be downloaded from github.com/opencv/open_vision_capsules. HW partner need to implement VisionCapsules HW adaptor. Follow the steps below based on a VisionCapsules sample implementation,

1. HW VisionCapsules receives METADATA, and pack it into proper AI results.
2. Inference should be bypass, as it is done on hardware
3. If the device stack is not supported by TensorFlow, OpenCV DNN, or OpenVino, then a custom device needs to be created. Check “Add a Custom Device”

HW VisionCapsules - Add a Custom Device to BrainFrame



BrainFrame out-of-the-box supports TensorFlow, OpenCV DNN, or OpenVino. If a HW partner implementation is not supported by TensorFlow, OpenCV DNN, or OpenVino, then a custom device needs to be added for BrainFrame to load HW accelerated DNN model to a HW device. Follow the steps below to add a Custom device to BrainFrame,

1. Adding Device Name to the system: BrainFrame Canned VisionCapsules Runtime Environment uses TensorFlow Stack list_local_devices to find all available devices. HW partner may contact Software Vendor for the BrainFrame to support a Custom Device.
2. Alternatively, HW partner can hack BrainFrame Implementation, replace TensorFlow Device with a Custom Device, so BrainFrame will route the request to the list_local_devices of the Custom Device.
3. User can always ignore 1) & 2) and manually load a DNN model with an external tool. BrainFrame will not be able to control the HW in this case.

OpenVisionCapsules Format

A portable format for all others, e.g. Tensorflow, Torch, ONNX, etc.

Items		Existing Model Formats: **Tensorflow** - .pb/.meta/.pbtxt, .tflite **Keras** - .h5, .keras **Caffe** - .caffemodel, .prototxt **Torch** - .pt, .pth, t7 **ONNX** -.onnx	**OpenVisionCapsules Format** .cap (Portable format for all others) (Self-contained, Executable)
	Design goal	DNN training	DNN deployment
DNN model Computing	AI computing target support: CPU, GPU, or other AI chip	✔	✔
	Batch-deploy DNN models onto multiple computing targets	X	✔
	Separately batchable inference from parallelizable operations	X	✔
Self-contained DNN model & processing information	DNN weights & model in a single file	X / ✔ Separate files in most formats	✔
	Executable logic for pre/post processing	X	✔
	Expose standardized APIs for runtime algorithm configuration	X	✔
DNN Model Interconnect, Compatibility	Describes the interconnection to other DNN models	X	✔
	Compatible with other formats	X	✔