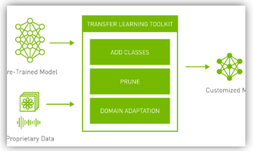
ACCELERATING ROBOTICS WITH NVIDIA ISAAC ROS

RAFFAELLO BONGHI, DEVELOPER RELATIONS MANAGER ROBOTICS

ISAAC- NVIDIA ROBOTICS PLATFORM

Accelerate the process with enhanced robotics development, simulation, and deployment



TRAIN

- AMP
- TAO Toolkit

SIMULATE

- OV SIM
- ISAAC ROS

DEVELOP

- ISAAC ROS
- RIVA
- DEEPSTREAM
- JETPACK
- PTM
- OSS

DATA

- OV SIM

Deploy

- ISAAC ROS
- RIVA
- DEEPSTREAM
- JETPACK
- TRITON
- FLEET COMMAND
- PTM


DGX/EGX


AGX

Ecosystem

SENSORS

SONY, Canon, Velodyne Lidar, BASLER

AI FRAMEWORKS

ONNX, mxnet, PYTORCH, TensorFlow

ROBOT COMPUTERS

AAEON, ADLINK, AVerMedia, ADVANTECH

ROBOTICS SW SOLUTIONS

kudan, ROS, SLAMCORE, concurrent REAL-TIME, MathWorks

REFERENCE PLATFORMS

DEVICE MGMT & CSP

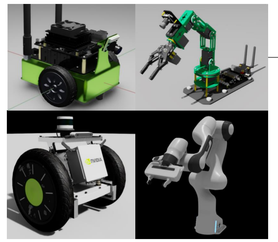
AWS Greengrass, balena, Azure, docker, MPI

NVIDIA

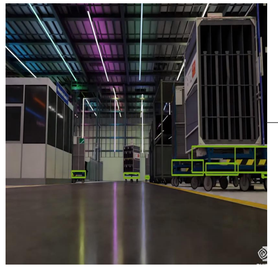
ISAAC SIM

Design, Train and Test Robots in Virtual World

ROBOT MODEL



WORLD MODEL



ISAAC SIM

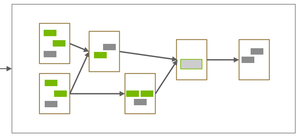
ROBOT IMPORTER	SENSOR MODELS	ROBOT UTILITIES	SYNTHETIC DATA	ROBOT INTERFACE
URDF CAD ONSHAPE SHAPENET	Contact sensors Stereo Camera RGBD Camera Lidar Ultrasonic	Map Generator RMP Controller Differential Base Joystick Control	2D/3D BBBOX Semantic Seg Instance Seg Point Cloud Randomization	ROS ROS2 ISAAC CUSTOM

OMNIVERSE

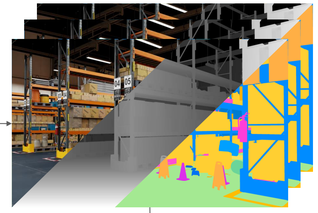
Nucleus	Connect	Kit	Simulation	RTX Renderer

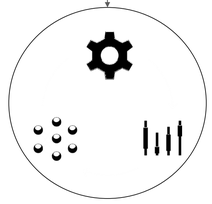
HIL/
SIL

ROBOT APPLICATION



SYNHETIC DATA





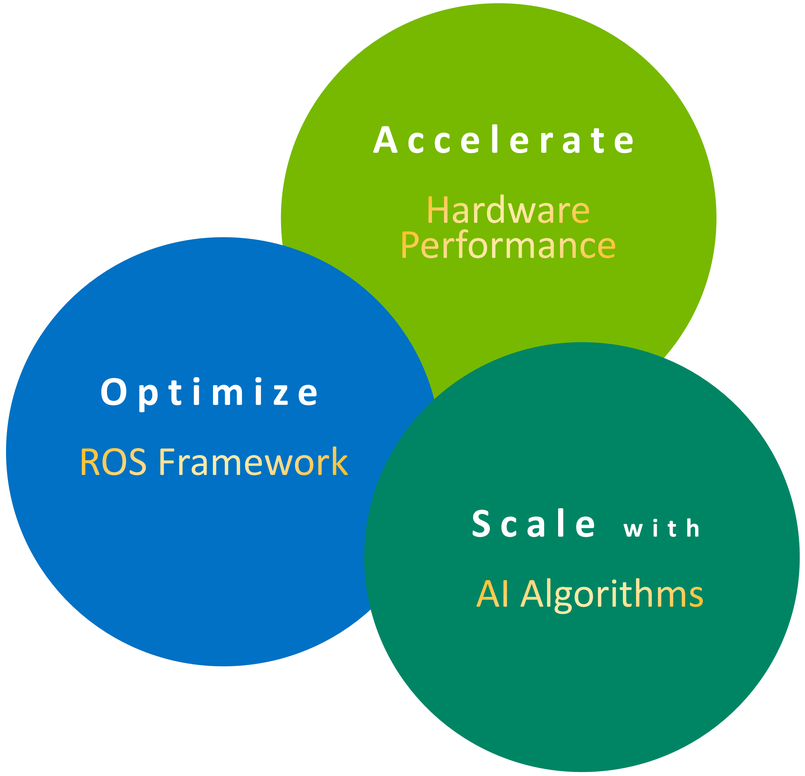
TAO

INTRODUCING ISAAC ROS

NVIDIA AI Perception to the ROS community

NVIDIA JETSON™

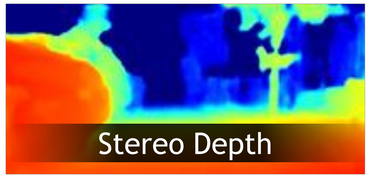
850,000+ Developers



2

750,000+ Downloads

AI modules that can plug into the ROS framework and get accelerated performance immediately



NVIDIA

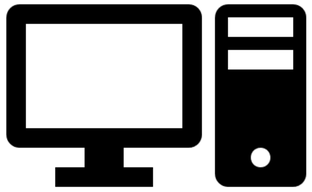
ISAAC ROS

HW accelerated GEMs as ROS2 packages

Isaac GEMs build on NVIDIA platforms to provide hardware accelerated robotics capabilities

Integrate GEMS as native nodes combined into ROS2 graphs with other ROS packages

Supports ROS2 Foxy on x86_64/dGPU (Ubuntu 20.04) and Jetson Xavier NX/AGX Xavier (JetPack 4.6)



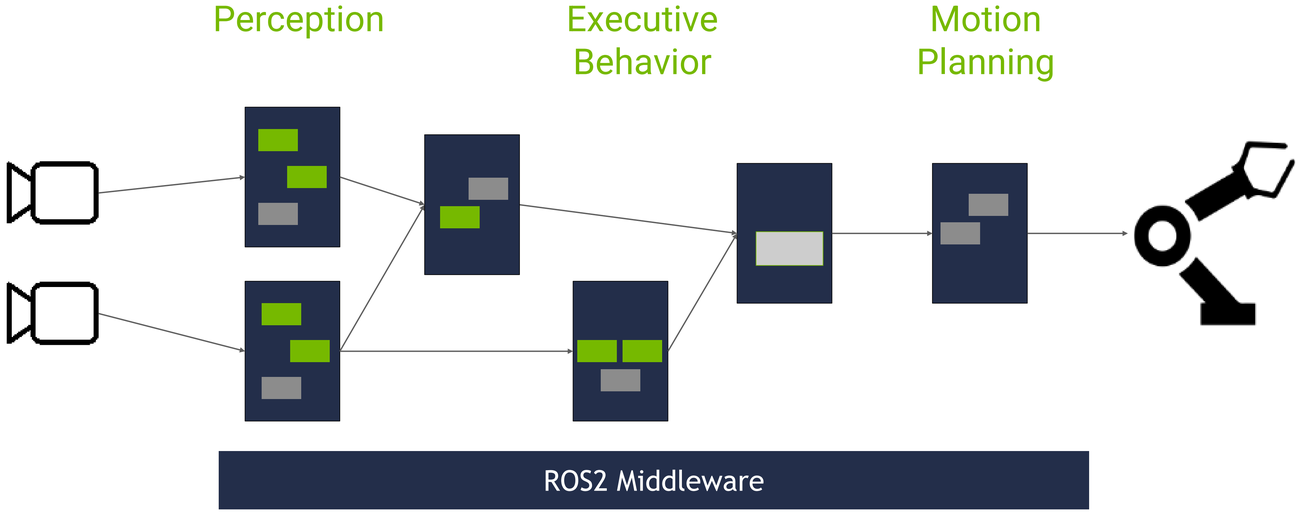
ROS

Robot Operating System

- ROS is a popular robotics autonomy framework with an active community

Started in 2007 at Willow Garage, ROS is now maintained by Open Robotics Foundation

- ROS2 is a redesign of ROS better suited for modern, commercial robotics
- ROS2 Foxy LTS until May 2023 (ROS1 Noetic ends life May 2025)
- Package repositories contains popular OSS frameworks (Nav2, RTabMap, MoveIt) and drivers



ISAAC ROS GEMs

Isaac_ros_image_pipeline

Isaac_ros_apriltag

Isaac_ros_dnn_inference

Isaac_ros_visual_odometry

Isaac_ros_argus_camera

NVIDIA ISAAC ROS GITHUB

https://github.com/NVIDIA-ISAAC-ROS

isaac_ros_apriltag Public

CUDA-accelerated Apriltag detection and pose estimation.

C++ 14 1

isaac_ros_visual_odometry Public

Visual odometry package based on hardware-accelerated NVIDIA Elbrus library with world class quality and performance.

C++ 49 2

isaac_ros_dnn_inference Public

Hardware-accelerated DNN model inference ROS2 packages using NVIDIA Triton/TensorRT for both Jetson and x86_64 with CUDA-capable GPU.

C++ 25

isaac_ros_image_pipeline Public

Hardware-accelerated ROS2 packages for camera image processing.

Python 9 1

isaac_ros_common Public

Common utilities, packages, scripts, Dockerfiles, and testing infrastructure for Isaac ROS packages.

C++ 6

isaac_ros_argus_camera Public

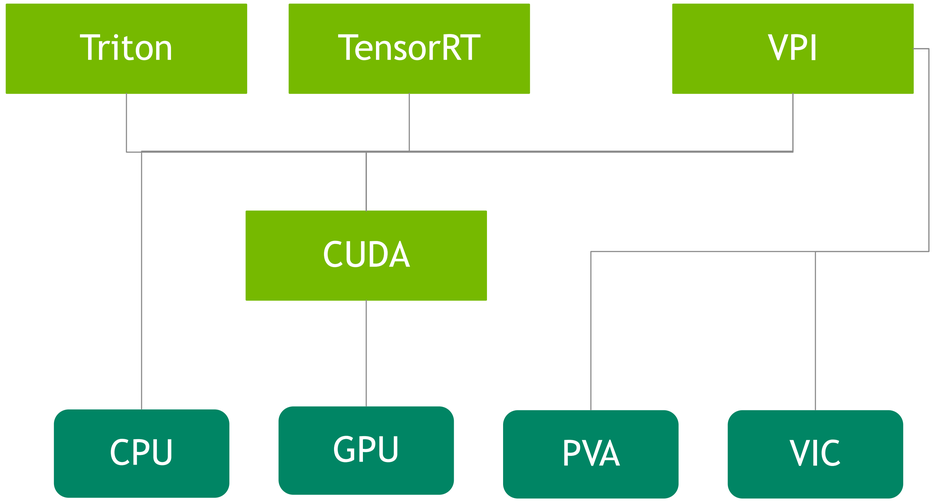
ROS2 packages based on NVIDIA libArgus library for hardware-accelerated CSI camera support.

C++ 10

NVIDIA ACCELERATED PERCEPTION

Unlock hardware performance

VPI (Vision Programming Interface)	Powerful computer vision (CV) and image processing (IP) algorithms implemented for different hardware backends such as CPU, GPU, PVA, and VIC.
CUDA SDK	Computing framework to implement accelerated vision algorithms on GPU.
TensorRT and Triton	High-performance deep learning inference for perception models on CPU/GPU.



- Isaac GEMs build on these and many other NVIDIA technologies to accelerate robotics capabilities
- **Isaac ROS** integrates Isaac GEMs into ROS

PVA = Programmable Vision Accelerator
VIC = Video and Image Compositor
Many others to be leveraged soon as GEMs

CAMERA IMAGE PROCESSING

Monocular Camera - isaac_ros_image_pipeline

- **Image Rectification**: After calibration, apply correction for lens distortion: radial, barrel, pincushion, tangential
- **Image Format Conversion:** Convert between images encoded as 8/16-bit mono, 8-bit RGB, BGR, with/without alpha
- **Image Resize:** Scale images with interpolation to target size

- All operations scale linearly (=poorly) with the size of the image being processed on CPU

Unrectified image with radial lens distortion

Rectified image with corrected distortion

https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_image_pipeline

ISAAC_ROS_IMAGE_PROC PACKAGE

isaac_ros_image_pipeline

- Hardware-accelerated monocular camera image processing through VPI (modeled on image_proc ROS package)

RectifyNode	Rectify/undistort raw camera images	***/image_raw*** ***/camera_info*** (camera intrinsics) ***/image_rect*** (rectified image)	CPU GPU VIC PVA
ImageFormatConverterNode	Convert between image formats supported by **cv_bridge** encoded in sensor_msgs.Image	***/image_raw*** ***/image*** (converted image)	CPU GPU VIC PVA
ResizeNode	Resize image with modified camera info	***/image_raw*** ***/camera_info*** ***/resized/image*** ***/resized/camera_info*** (resized)	CPU GPU VIC PVA

https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_image_pipeline

PVA = Programmable Vision Accelerator
VIC = Video and Image Compositor
Many others to be leveraged soon as GEMs

STEREO CAMERA DISPARITY

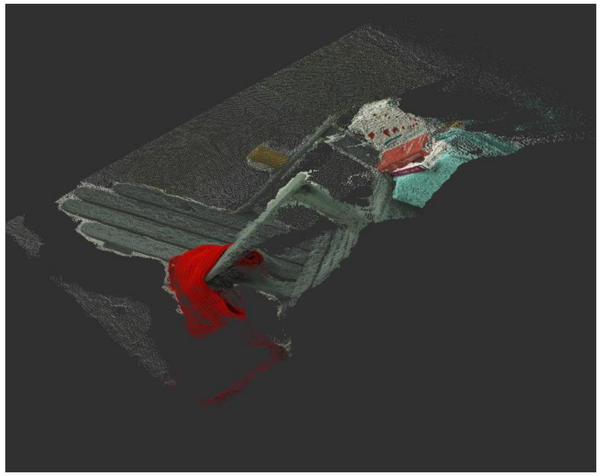
Depth from stereo image pairs - isaac_ros_image_pipeline

Stereo camera

Stereo image pair

Disparity image

Point cloud



- Processing stereo camera image pairs key to point cloud perception algorithms and 3D sensor fusion
- Disparity images often calculated from rectified images and baseline using semi-global matching algorithm (SGM)
- Point clouds derived from disparity with color image projected for XYZRGB
- Computationally intensive for CPU and scales poorly with the size of the images

https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_image_pipeline

ISAAC_ROS_STEREO_IMAGE_PROC PACKAGE

isaac_ros_image_pipeline

Hardware-accelerated stereo camera image processing using VPI(modeled on stereo_image_proc ROS package)

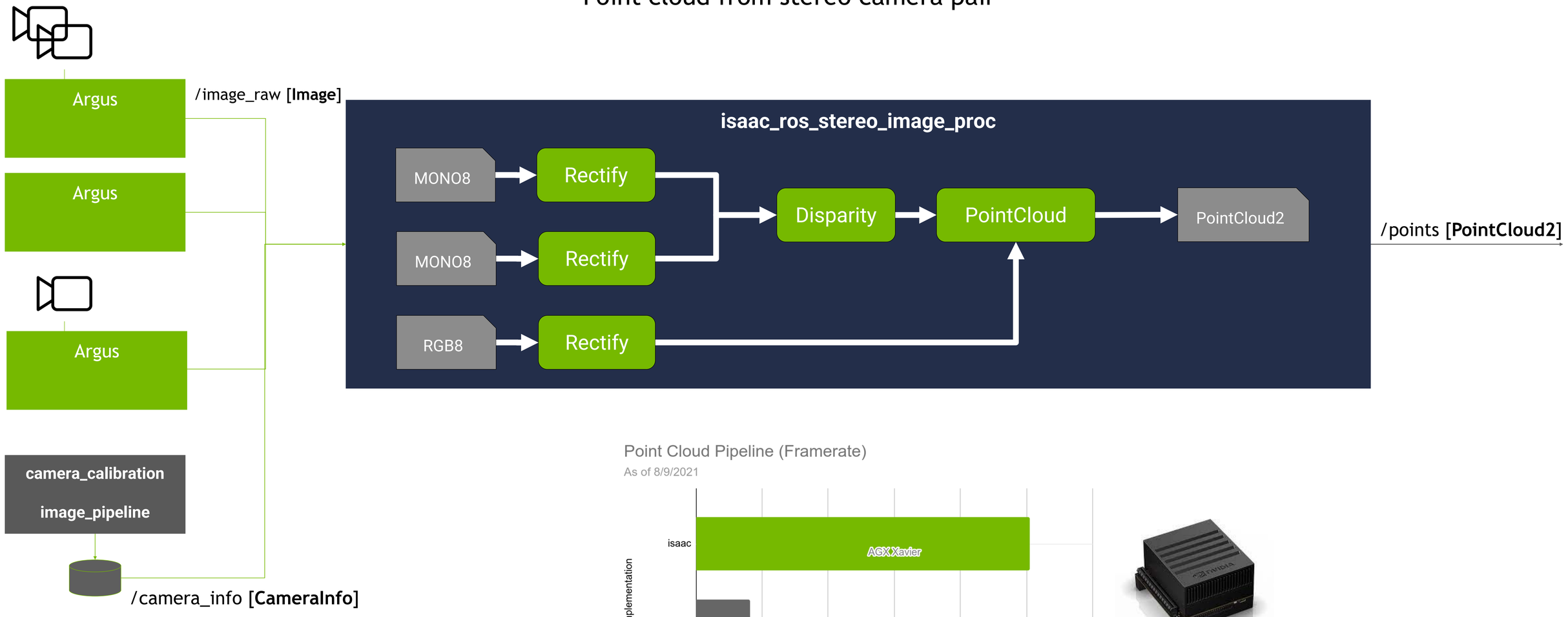
DisparityNode	Calculate disparity image between camera in a stereo pair using SGM	***/left, right/image_rect*** ***/left, right/camera_info***	CPU GPU VIC PVA
		/disparity (disparity image)	
PointCloudNode	Derive colorized point cloud from disparity image	***/disparity*** ***/left/image_rect_color*** *(color image)* ***/left, right camera_info***	CPU GPU VIC PVA
		/points2 (point cloud xyzrgb)	

https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_image_pipeline

PVA = Programmable Vision Accelerator
VIC = Video and Image Compositor
Many others to be leveraged soon as GEMs

STEREO CAMERA POINT CLOUD PIPELINE

Point cloud from stereo camera pair



https://developer.nvidia.com/isaac-ros-gems
https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_image_pipeline

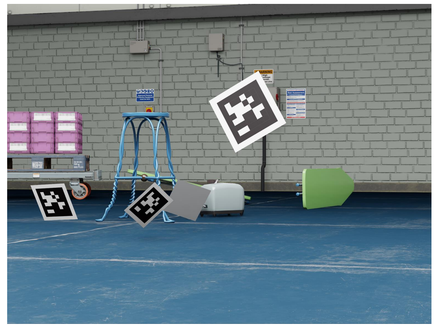
APRILTAG DETECTION

Fiducials for pose estimation with monocular cameras

- AprilTags are a popular fiducial with applications across robotics perception
- Tags from different families can be printed out and placed in an environment
- With an image from a monocular camera, intrinsic parameters from calibration, tag size, and tag family provided, an AprilTag detector can determine
  - identity of the tags in the frame
  - 2D bounding boxes for each tag
  - 6DOF poses for each tag relative to the camera

Camera Tag

- Enables localization and object tracking for closed-loop manipulation to camera extrinsic calibration routines
- On CPU, the detection/pose estimation workload scales with the image size



From Isaac SIM

From Camera

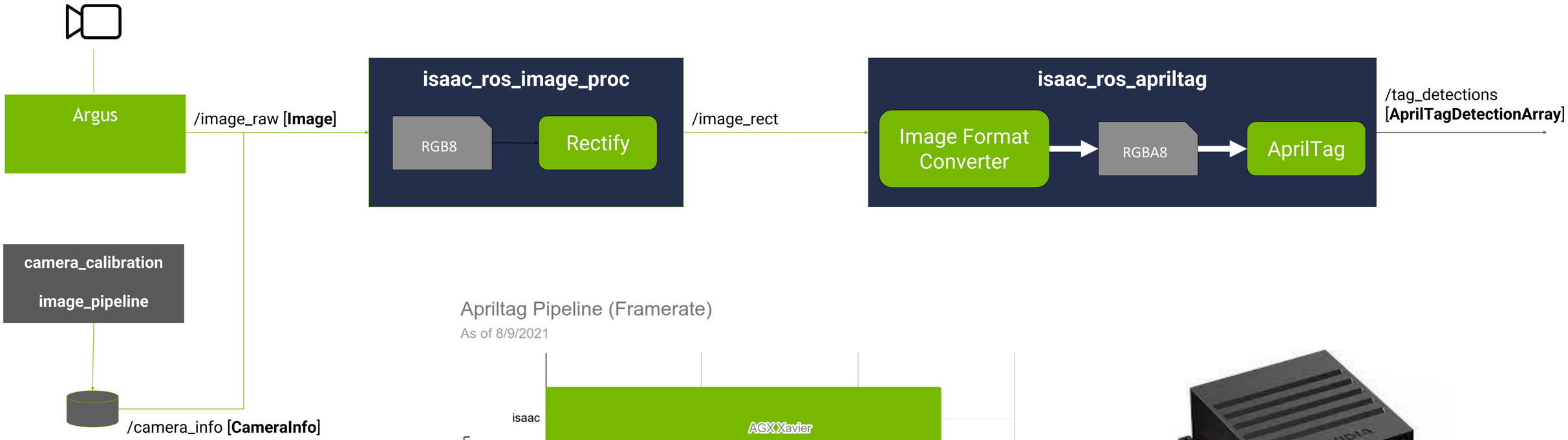
AprilTagNode	Detect 6DOF pose of Apriltag in family 36h11 in monocular camera frame	***/image_rect*** *(image in rgba8)* ***/camera_info*** (camera intrinsics)	CPU **GPU** VIC PVA
		/tag_detections (rectified image) ***/tf*** (camera_pose_tag transforms)	

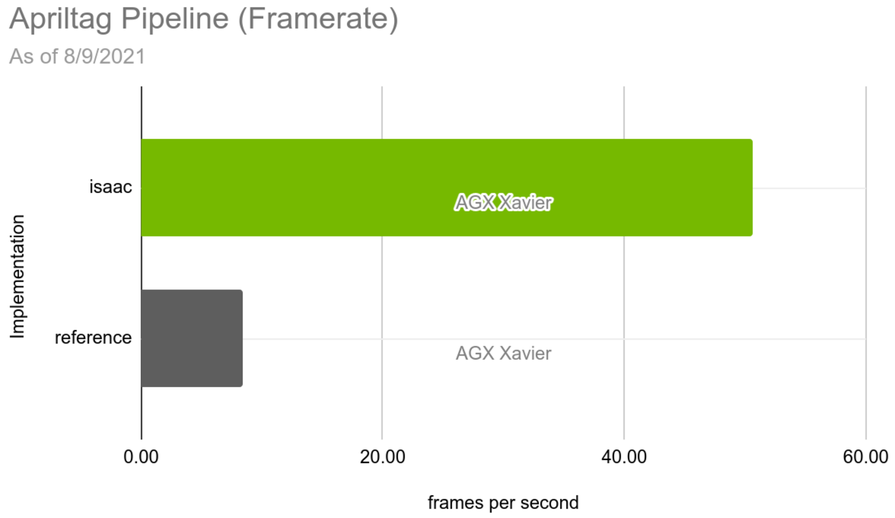
https://developer.nvidia.com/isaac-ros-gems

PVA = Programmable Vision Accelerator
VIC = Video and Image Compositor
Many others to be leveraged soon as GEMs

APRILTAG PERCEPTION PIPELINE

Detect tag and 6DOF pose in monocular camera frame





https://developer.nvidia.com/isaac-ros-gems

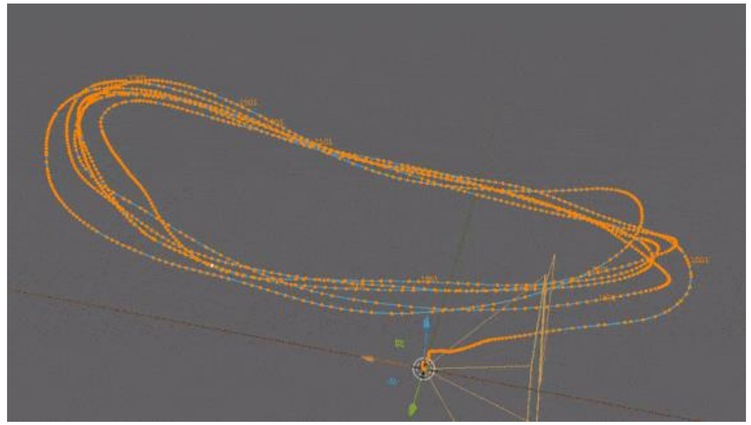
VISUAL NAVIGATION ACCELERATION

ROS2 nodes to provide acceleration for ego motion determination

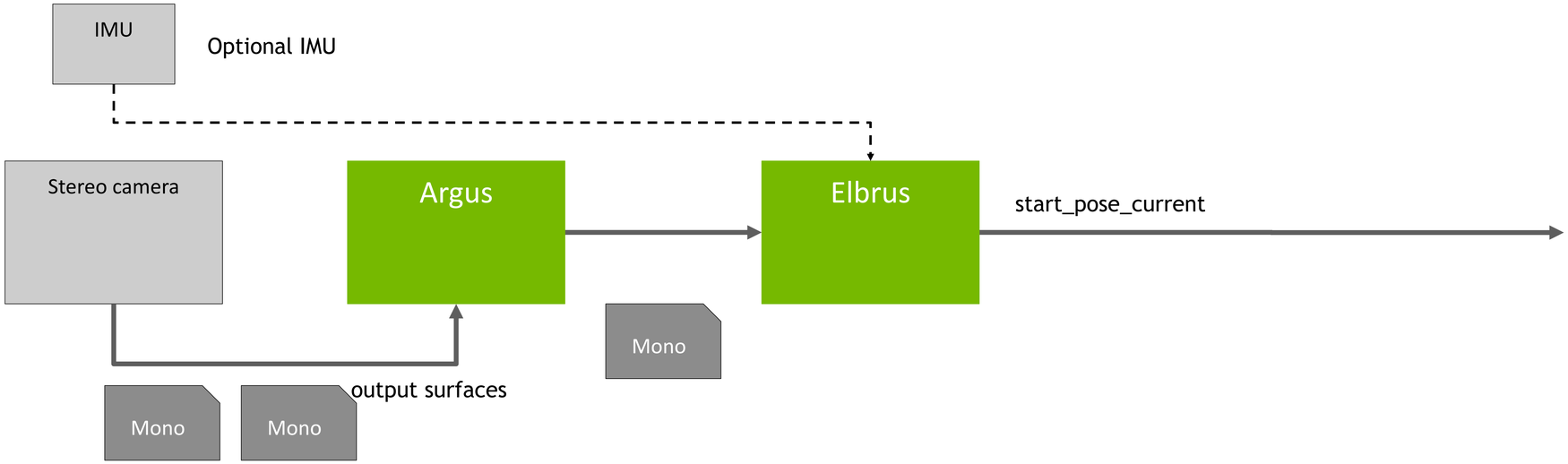
Visual Odometry is a method for estimating a camera position relative to its start position. This method has an iterative nature: at each iteration it considers two consequential input frames (stereo pairs). On both frames, it finds a set of keypoints. Matching keypoints in these two sets gives the ability to estimate the transition and relative rotation of the camera between frames.

- This package provides ROS2 node to estimates stereo visual inertial odometry using the Isac Elbrus

It is now possible to run SLAM on HD resolution (1280x720) in real-time (>60fps) on a Jetson Xavier AGX



Visual odometry pipeline

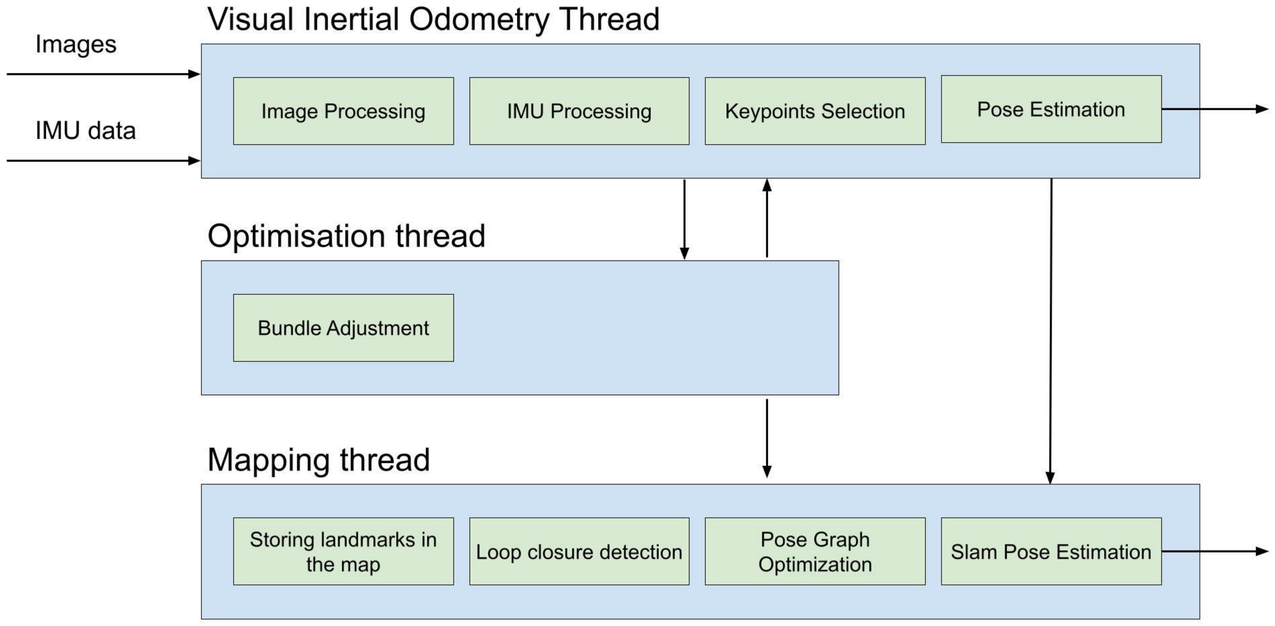


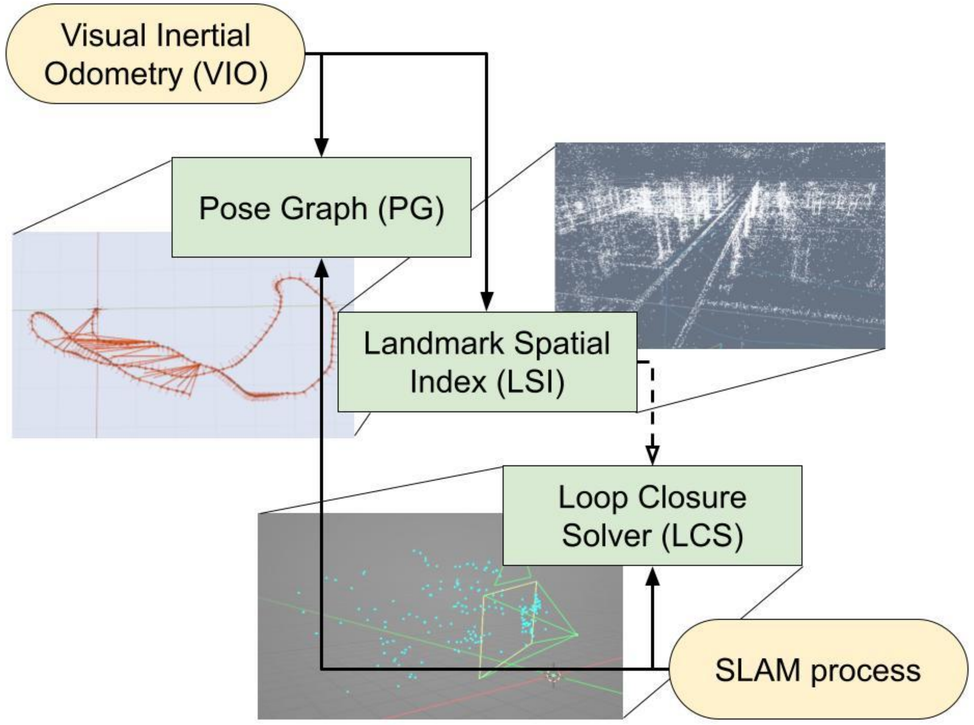
http://www.cvlibs.net/datasets/kitti/eval_odometry.php
https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_visual_odometry

VISUAL ODOMETRY

Isaac Elbrus

Elbrus implements a Stereo VIO and SLAM architecture based on keyframes, which is a two-tiered system: a minor subset of all input frames are used as key frames and processed by additional algorithms, while the other frames are solved quickly by 2D tracking of already selected observations. The end-to-end tracking pipeline contains two major components: 2D and 3D.

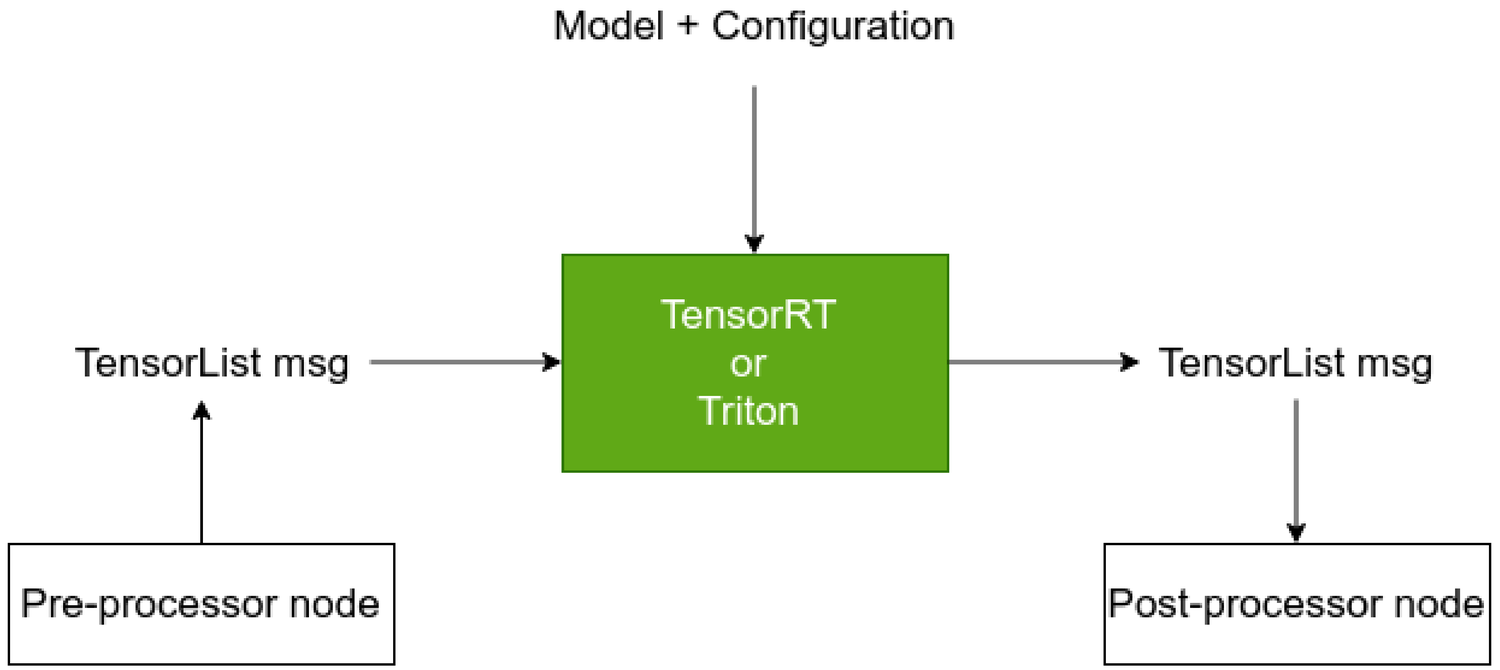


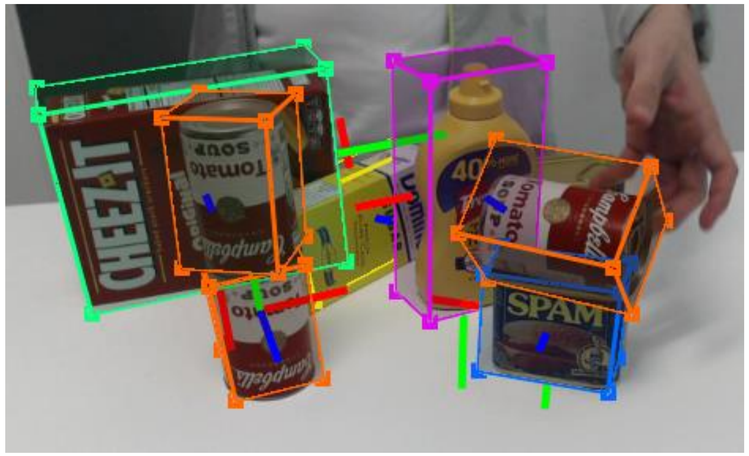


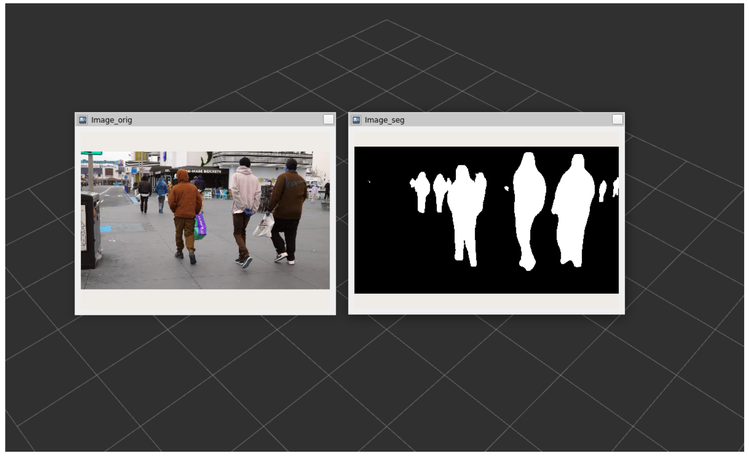
https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_visual_odometry

DNN INFERENCE

Integrate TensorRT and Triton SDK in a powerful ROS2 package







DNN model inference for custom and pre-trained DNNs 2 with included examples for DOPE 3 3D pose estimation & U-NET semantic image segmentation (pre-trained PeopleSemSegNet 25fps at 544p)

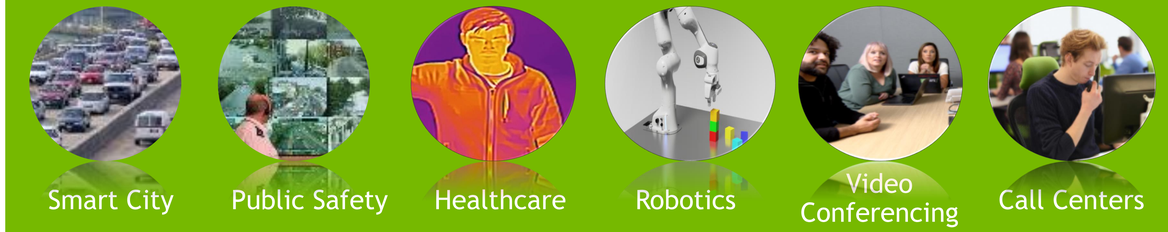
https://developer.nvidia.com/tensorrt
https://developer.nvidia.com/nvidia-triton-inference-server
https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_dnn_inference

PRE-TRAINED MODELS

Multi-Purpose Pretrained Models

Reduce development time by 10x | Highly Accurate | Optimized for performance

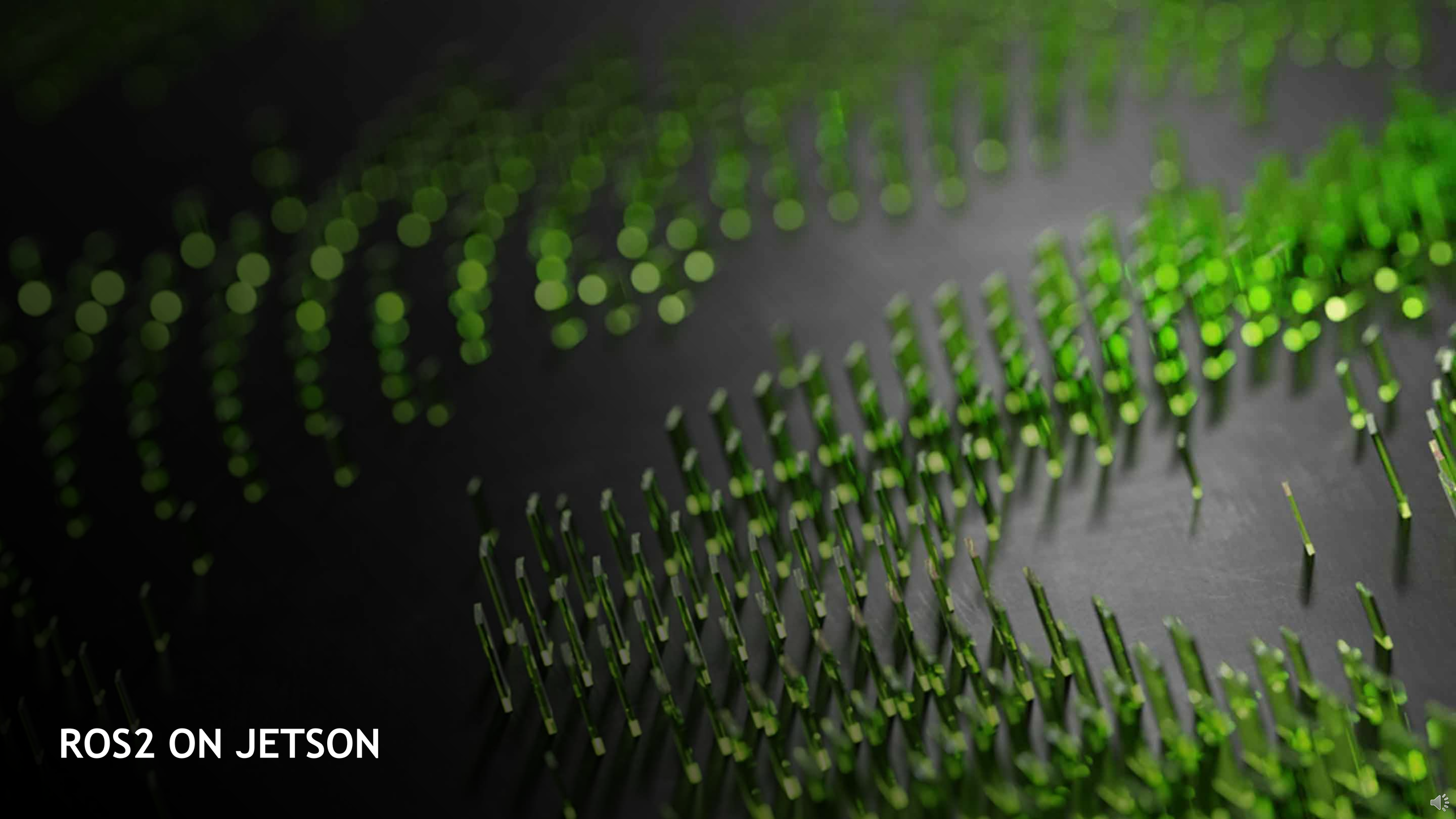
NVIDIA Optimized domain specific models



Flexible, train 100+ permutations of model arch & backbones | Supports YoloV3/V4, EfficientNet, FasterRCNN | State-of-the-art accuracy on public

General Purpose Models

TAO toolkit


ROS2 ON JETSON

ROS2 ON JETSON

Docker containers to develop and run Isaac ROS packages

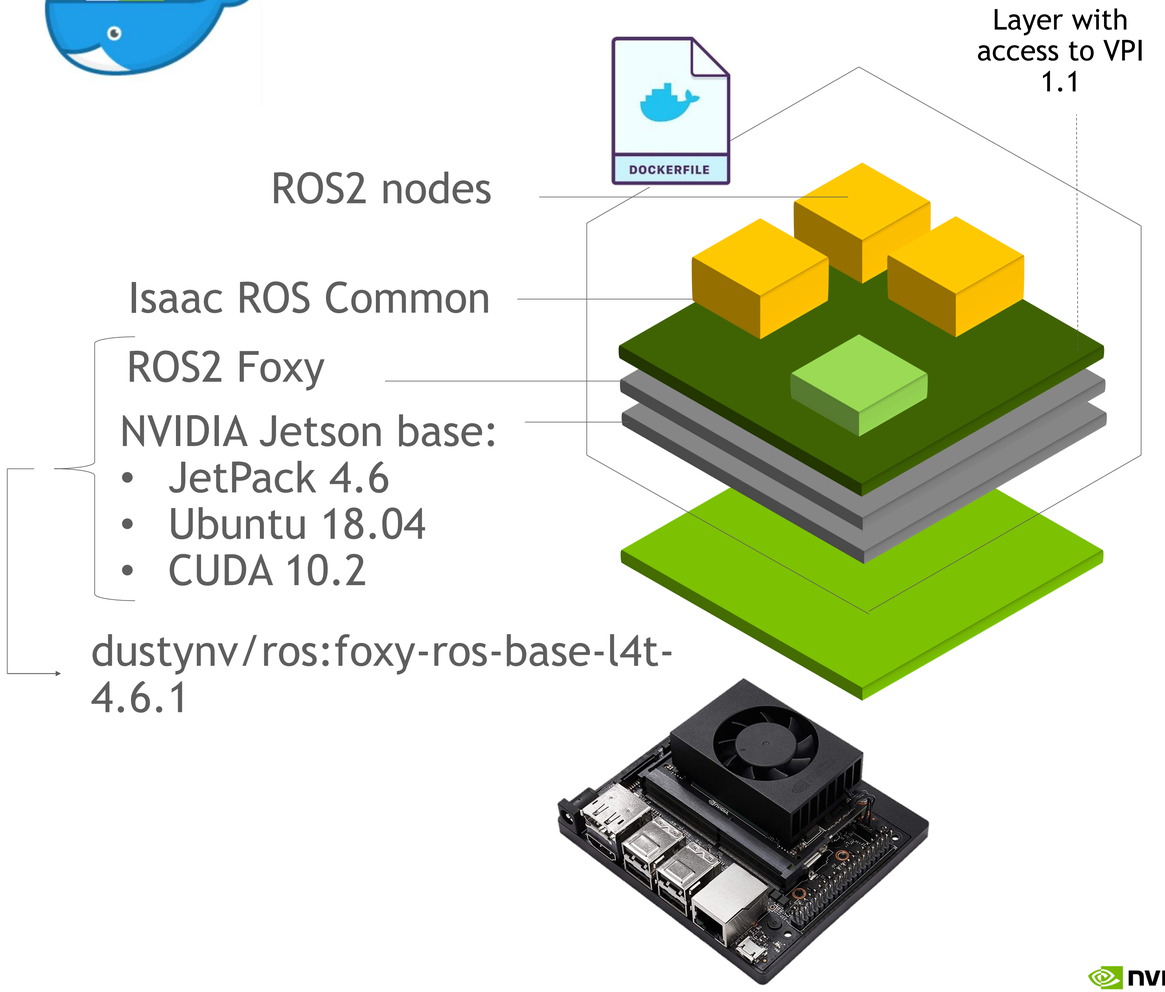
ROS2 Foxy supported on Ubuntu 20.04

However, JetPack 4.6 L4T based on Ubuntu 18.04

- **Solution 1**: Compile ROS2 Foxy from source yourself on JetPack 4.6 and install dependencies manually

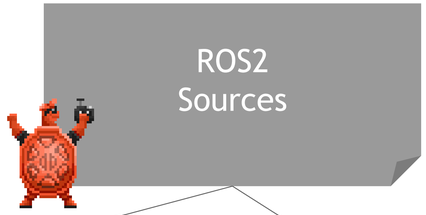
- **Solution 2**: Docker image with ROS2 Foxy and useful dependencies compiled from source on L4T

Run container with NVIDIA Container Runtime to access CUDA-X layers on Jetson



ROS ON NVIDIA JETSON

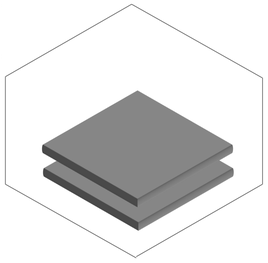
How we made ROS2 for NVIDIA Jetson



ROS2
Sources

NVIDIA Jetpack X
X = 4.5, 4.6.1

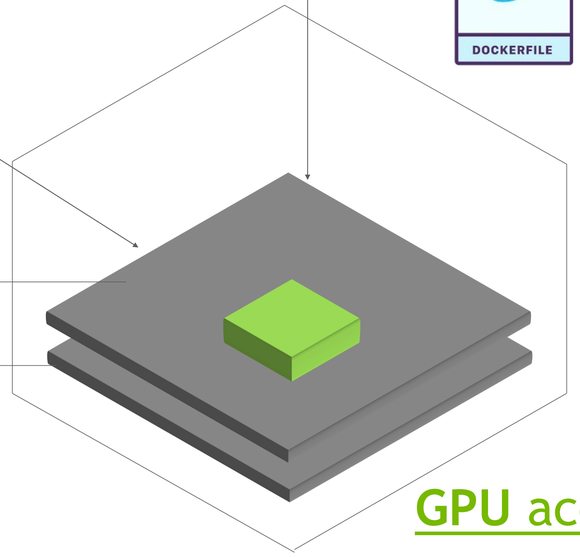
ROS Docker



ROS2

NVIDIA Jetson base:

- JetPack 4.X
- Ubuntu 18.04
- CUDA 10.X

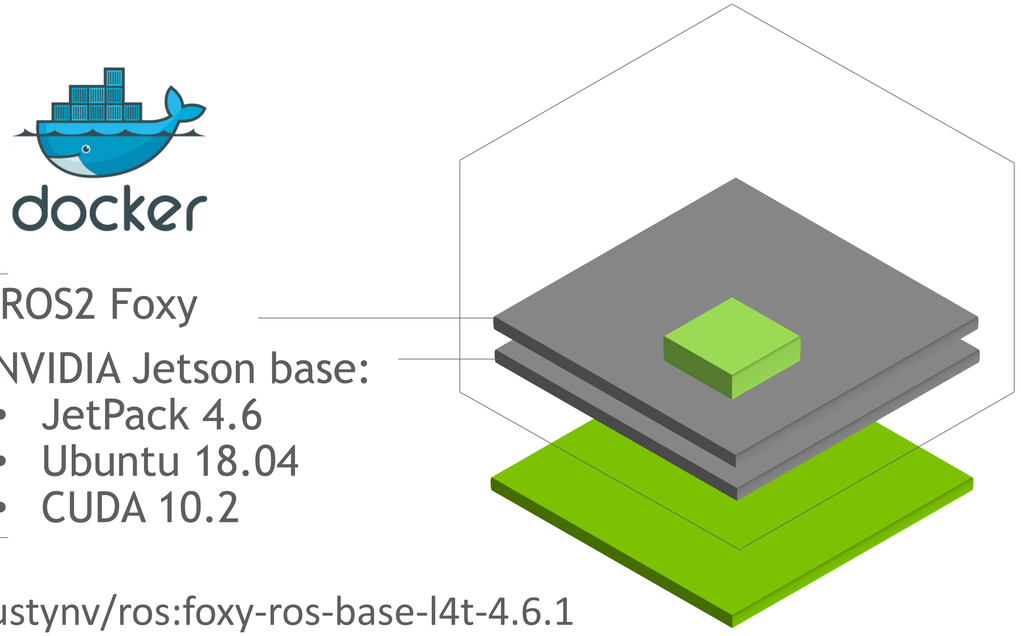


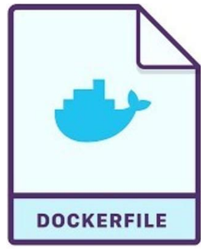
GPU accelerated

Both docker distributions runs on NVIDIA Jetson

ISAAC GEMS FOR ROS

New packages for ROS2





Isaac GEMs for ROS specifications:
- ROS2 – Foxy
- Jetpack 4.6
- CUDA 10.2
- VPI 1.1

Repository to build Docker Isaac ROS image
https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_common

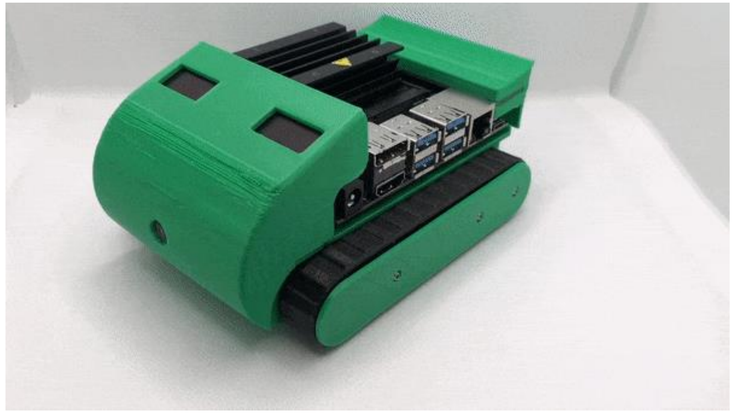
- Grant access to VPI 1.1 inside the Docker container
- VPI wrapper package to ROS2 called Isaac_ros_common

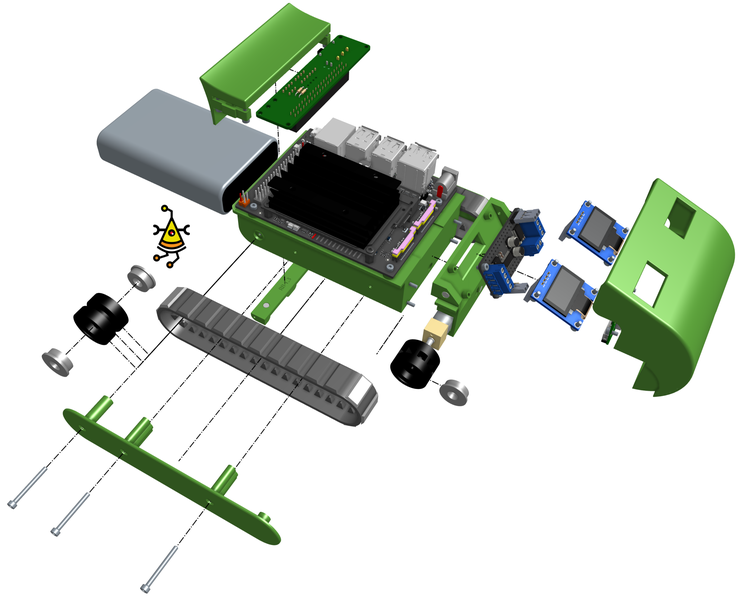
nanosaur

- The smallest NVIDIA Jetson robot
- Fully 3D printable
- Open-source
- ROS2 foxy based
- https://nanosaur.ai

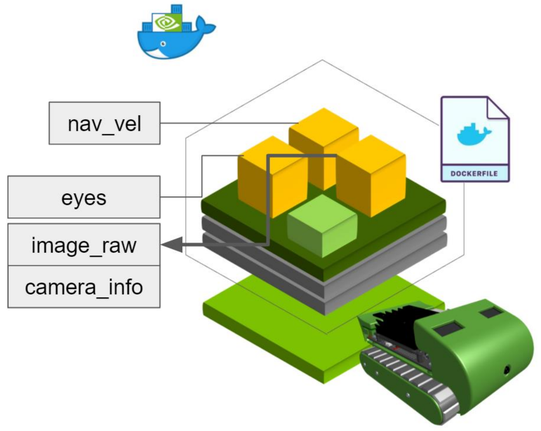
NANOSAUR

Open-source NVIDIA Jetson robot





- https://nanosaur.ai
- nanosaur is a simple open-source robot based on NVIDIA Jetson.
- The robot is fully 3D printable, able to wander on your desk autonomously
- ROS2 based (Available for ROS2 🦊Foxy and 🐢Galactic)
- uses a simple camera and two OLEDs — these act as a pair of eyes.
- It measures a compact 10x12x6cm and it weighs only 500g.
- Designed on ROS2 Jetson Docker
  - https://github.com/dusty-nv/jetson-containers



NANOSAUR

Architecture design

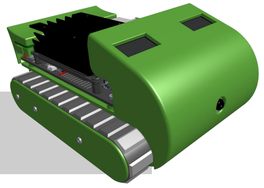
joy_vel

key_vel

nav_vel

twist_mux

twist_control (nanosaur_control)

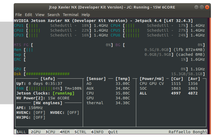


nanosaur_base

eyes

diagnostics

ros2_jetson_stats



nanosaur_camera

image_raw

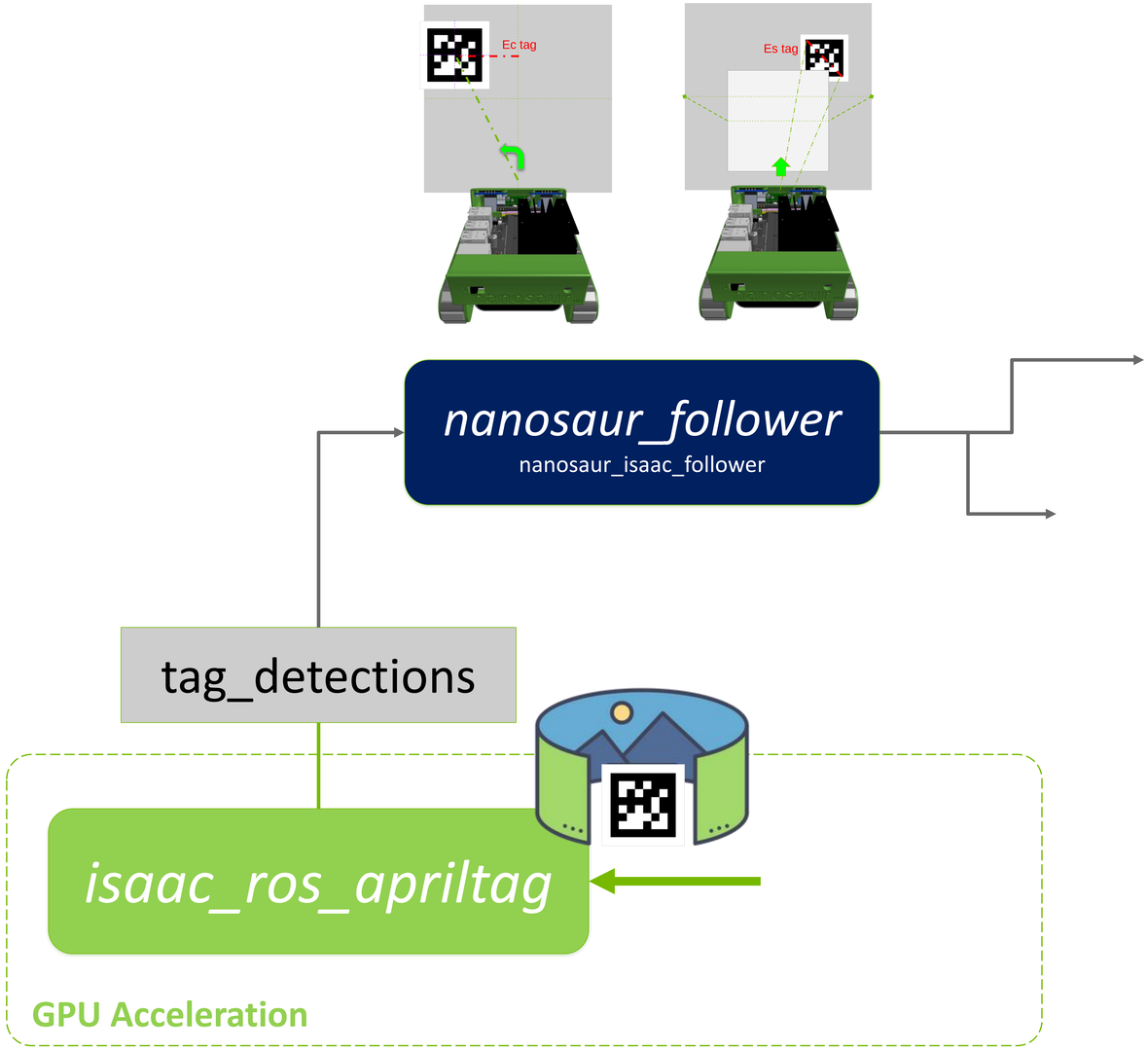
camera_info

GPU Acceleration

- Nanosaur such can be driven from
  - joystick, keyboard, or other input simultaneously
- ros2_jetson_stats update the NVIDIA Jetson status sending diagnostics messages
- You can also control the eyes to emulate a real dinosaur face

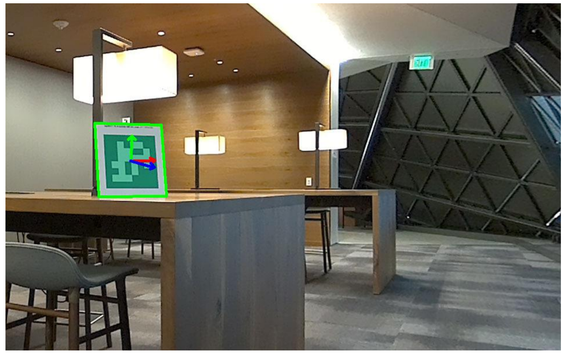
NANOSAUR

Architecture design



- Isaa_ros_apriltag detect all apriltag from a frame and publish a **tag_detections** message. The message contain:
  - Tag centre
  - Tag corners
  - Pose and orientation from camera frame

- nanosaur_follower select a configurable apriltag and generate a command to drive nanosaur
  1. the first controller (A) reduces to zero the error from the centre AprilTag corner to the centre vertical line
  2. second error coming from camera distance drives the robot speed (B)



NANOSAUR

Architecture design

joy_vel

key_vel

nav_vel

twist_mux

twist_control (nanosaur_control)

nanosaur_follower

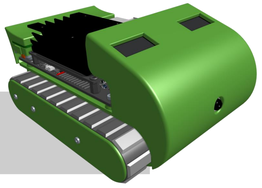
nanosaur_isaac_follower

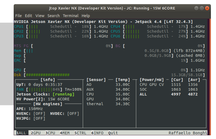
eyes

diagnostics

nanosaur_base

ros2_jetson_stats





tag_detections

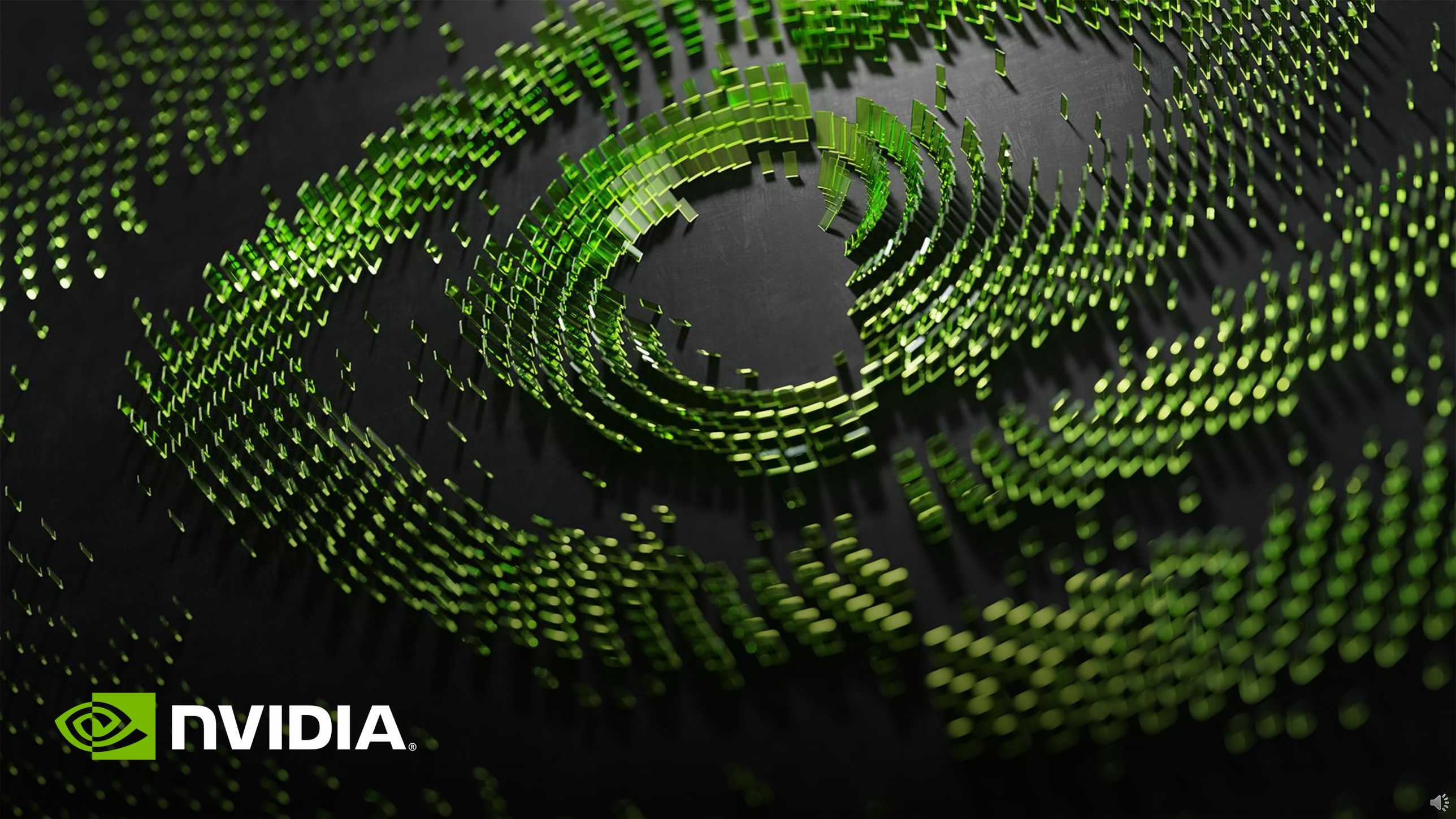
isaac_ros_apriltag

image_raw

camera_info

nanosaur_camera

GPU Acceleration


NVIDIA