

# Simulating real robots in virtual environments using NVIDIA's Isaac SDK

Filipe Figueredo Monteiro  
Voxar Labs  
Universidade Federal de Pernambuco/CIn  
Recife, Brazil  
ffm@cin.ufpe.br

André Luiz Buarque Vieira-e-Silva  
Voxar Labs  
Universidade Federal de Pernambuco/CIn  
Recife, Brazil  
albvs@cin.ufpe.br

João Marcelo Xavier Natrio Teixeira  
Departamento de Eletrônica e Sistemas  
Universidade Federal de Pernambuco/CTG  
Recife, Brazil  
jmxnt@cin.ufpe.br

Veronica Teichrieb  
Voxar Labs  
Universidade Federal de Pernambuco/CIn  
Recife, Brazil  
vt@cin.ufpe.br

**Abstract**—Isaac SDK and Sim tools produced by NVIDIA have great potential in assisting, accelerating and even enabling, in some cases, the development of robotic applications that make use of some sort of artificial intelligence technique. This tutorial intends to show to its audience a quick theoretical background, the whole setting up process of such tools, how to integrate it to a powerful embedded system, the Jetson Nano, and a number of examples of current applications together with a good portion of possibilities enabled by those tools.

**Index Terms**—Robotics, AI, Isaac SDK, Isaac Sim, Jetson Nano

## I. TUTORIAL

The release of Isaac Sim and Isaac SDK by NVIDIA [1] brought new opportunities for developing robotics solutions as well as researching topics in this field [2]. By using these new tools, we can accelerate the development and deployment of AI-powered robots. Isaac provides high-performance robotics algorithms, hardware reference applications and a powerful simulation platform, accelerating robot development for manufacturers, researchers and startups by making it easier to add and simulate AI for perception and navigation into modern day robots. This tutorial aims at presenting:

- The main concepts and ideas of Isaac Sim and Isaac SDK
- How to perform the setup for using Isaac
- Isaac integration with the Jetson Nano embedded system
- A few use cases and examples of applications

In the next paragraphs, the above topics will be addressed coupled with visual representations for each one of them.

Firstly, the Isaac Sim is a virtual robotics laboratory and a high-fidelity 3D world simulator. It accelerates research, design, and development in robotics by providing state-of-the-art tools for testing and simulation. Developers can quickly and easily test their robots in detailed and realistic scenarios. Developers can also use virtual robots with several simulated sensors, including RGB, stereo, depth, segmentation, LIDAR and IMU. Once tested, applications can be deployed

to NVIDIA Jetson AGX Xavier, Jetson TX2 or Jetson Nano running on physical robots.

Robots in Isaac Sim are tightly coupled to the tools and frameworks in Isaac SDK, enabling easy transfer of algorithms and data between physical and virtual robots. By using synthetic data and controlling the randomization parameters in a scene, developers can create huge customized data sets. They can seamlessly randomize lighting, object materials, object colors, object poses and camera properties as seen in Figure 1. Isaac Sim also has an embedded URDF loader which allows you to load the URDF model of your robot and simulate its joints and movements. The URDF models for several mobile bases and manipulators have already been fully tested in Isaac Sim, which is an important functionality since URDF is the most common robot model. This allows an easy import of robot models to the simulator.



Fig. 1. Isaac Simulation.

Regarding the Isaac SDK, it is a software framework that allows the user to build modular robotics applications. It enables high-performance data processing and deep learning for intelligent robots. Robotics applications developed on Isaac SDK's robot engine can seamlessly run on edge devices like the NVIDIA Jetson AGX Xavier and NVIDIA Jetson Nano, as well as on workstations with modest NVIDIA GPUs. The robot engine allows developers to break down a complex robotics scenarios into smaller building blocks. It is also

possible to customize SDK features to fit their use cases by configuring pre-packaged components and even adding new features by developing custom components.

The SDK applications consist of multiple nodes and use inter-node communication to exchange data between them. Developers can also group nodes into subgraphs and effectively combine them into a robotics application. Other capabilities include experimental message recording and replay of sensor data or message passing over network sockets. The complete environment known as Isaac Sight is shown in Figure 2.

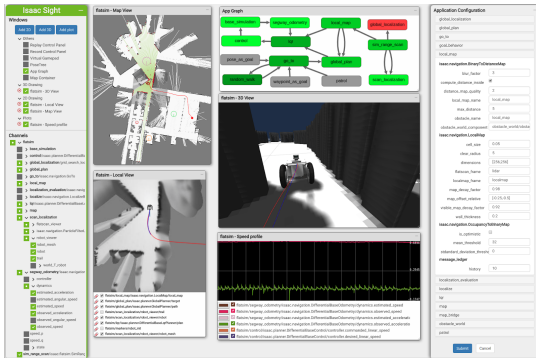


Fig. 2. Isaac Sight.

Developers can add obstacle detection, stereo depth estimation, or human speech recognition to enrich their robot use cases by using GEMS. They are modular capabilities for sensing, planning, or actuation that can be easily plugged into a robotics application. Similarly, developers can use the Isaac navigation stack, a collection of modules to build a robotics navigation module, to their robot. Regarding perception, Isaac SDK offers Stereo Depth, Stereo Visual Odometry, Object/AprilTag detection, Path Segmentation DNN, CUDA Orb and Superpixels GEMS. In Navigation, Lidar-based Global Localization and LQR Path Planning are the most useful GEMS, and they provide support for non-circular robot bases. GEMS are also used to simulate sensors, and Isaac provides Stereo camera, Structured Light Depth Camera, Lidar, IMU and Audio Keyword Detection sensor GEMS. By combining GEMS, developers and researchers can focus on other challenges, building their solutions on top of it.

About the NVIDIA Jetson Nano (Figure 4), it is an AI computer for makers, learners, and developers that brings the power of modern artificial intelligence to a low-power, easy to-use platform. It is supported by the comprehensive NVIDIA JetPack SDK, and has the performance and capabilities needed to run modern AI workloads. Nano comes with full desktop Linux with NVIDIA drivers, developer tools and AI and computer vision libraries and APIs, including: TensorRT and cuDNN for high-performance deep learning applications; CUDA for GPU-accelerated applications across multiple domains; and VisionWorks and OpenCV for visual computing applications.

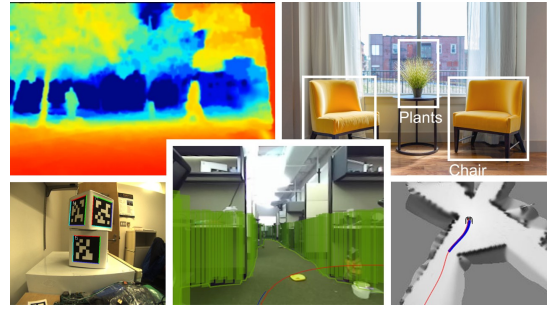


Fig. 3. Outputs of some Isaac GEMS.

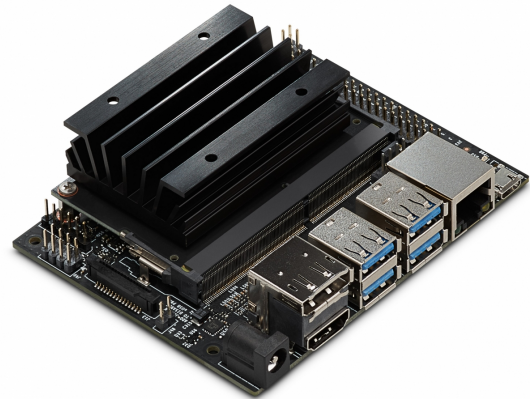


Fig. 4. Jetson Nano board.

Three of the case scenarios of NVIDIA Isaac that will be presented during the tutorial are shown below:

- 1) **Visual recognition neural network**  
One of the many capabilities of the Isaac SDK is the ability to easily deploy Neural Networks. In this use case, we'll explore how to test and run image and object recognition DNNs, such as ResNet and GoogLeNet, and discuss how to create your own DNN. We will also discuss some Deep Learning techniques and frameworks.
- 2) **Basic Follow-Me routine**  
A navigation demo. In this use case, we will test and construct a basic follow me routine, using a robot model from Isaac Sim. We will discuss the challenges of this task, and how we can minimize them.
- 3) **Path planning and cartography applications**  
Expanding on navigation, path planning and cartography are two applications that are in high demand in the robotics field. In this use case, we will build and test, by using Isaac SDK's GEMS, a full path planning and cartography stack. We will discuss challenges and possibilities that building this stack provides.

#### REFERENCES

- [1] NVIDIA Corporation. NVIDIA Isaac SDK: Accelerate your creation of autonomous machines, 2018.
- [2] Taro Narahara and Yoshihiro Kobayashi. Personalizing homemade bots with plug & play ai for steam education. In *SIGGRAPH Asia Technical Briefs*, pages 9–1, 2018.