Kinect

Kinect is a line of motion-sensing input device made by Microsoft. It is a depth-sensing camera featuring a RGB camera and Depth Sensor.

- Know the Kinect
- Install Drivers
- Use Kinect in TouchDesigner
- Use Kinect Anywhere

Know the Kinect

There is a Tech Demo from MicroSoft of what Kinect can do.

Kinect can do:

- Skeletal tracking in 3D space with human body joints
- IR image
- Depth image
- Spatial audio recording

Kinect can’t do:

- High precision motion capture
- Hand(gesture)/face(landmark) tracking (with only native Kinect SDK)

** Kinect offers native support on Windows only

** For macOS users, the alternative option is to use webcam with OpenCV, MediaPipe for tracking. LeapMotion is also an option, but limited to hand and gesture on a smaller scale.

There are two versions of Kinect available from the Kit Room, Kinect for Windows V2 and Azure Kinect. The functionality is mostly the same, with some difference in specs and formfactor. Kinect for Windows V2 is set up and installed in the Darklab.

Kinect for Windows V2

Azure Kinect

Install Drivers

Kinect for Windows and Azure require different drivers and SDK. Identify your Kinect and install the corresponding SDK from below.

Kinect for Windows V2

Azure Kinect

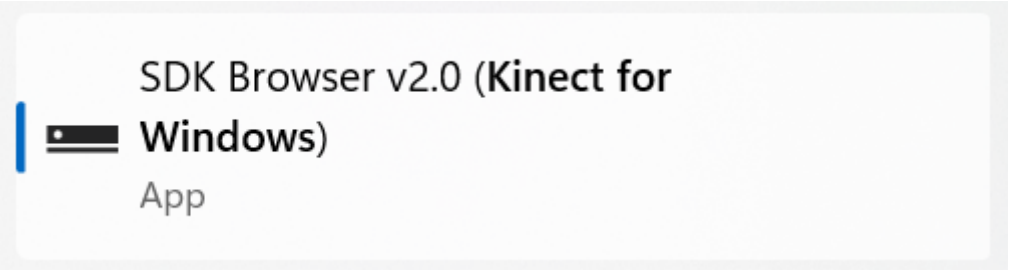
Kinect for Windows V2

Download and install the Kinect for Windows SDK 2.0

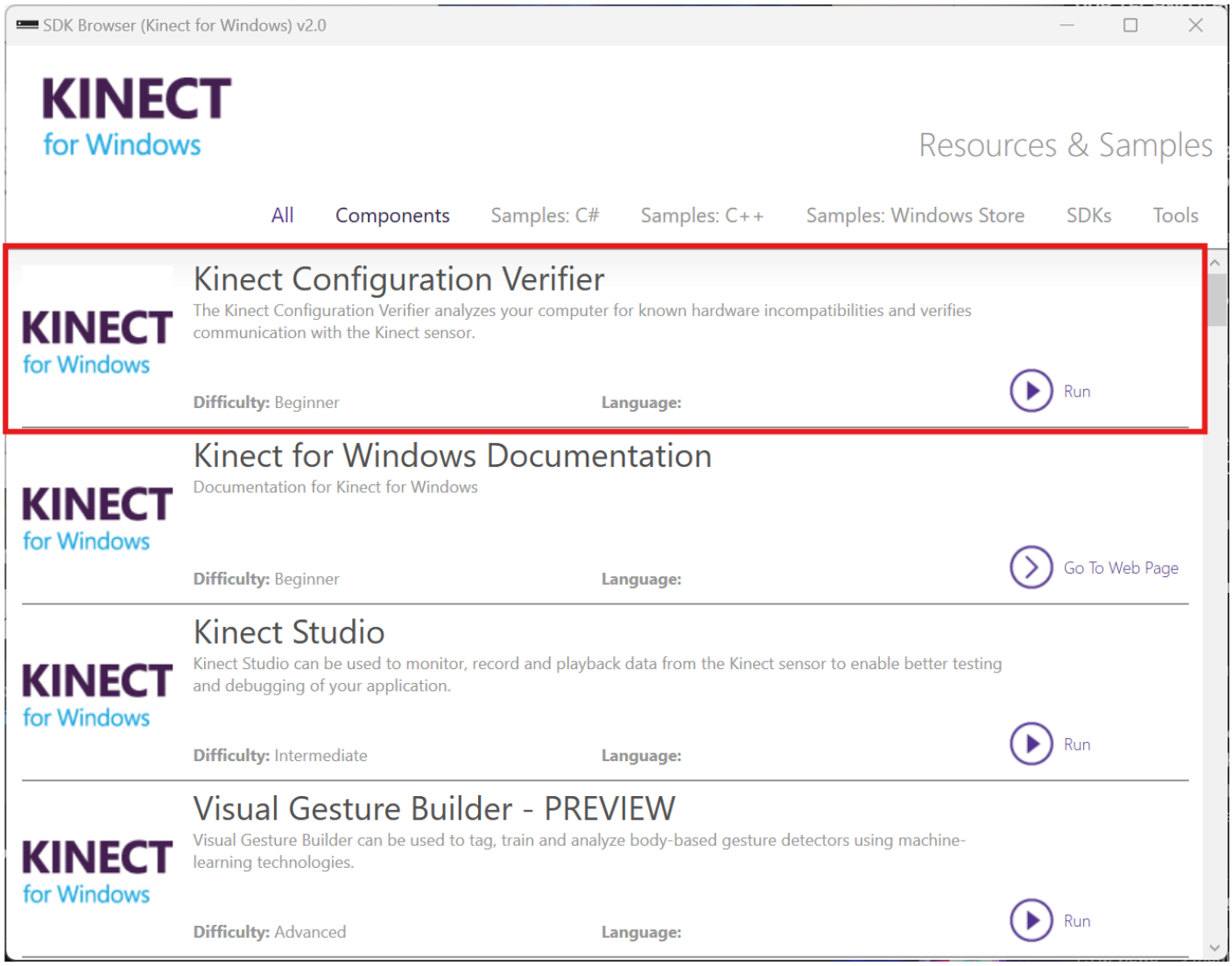
Download Kinect for Windows SDK 2.0 from Official Microsoft Download Center

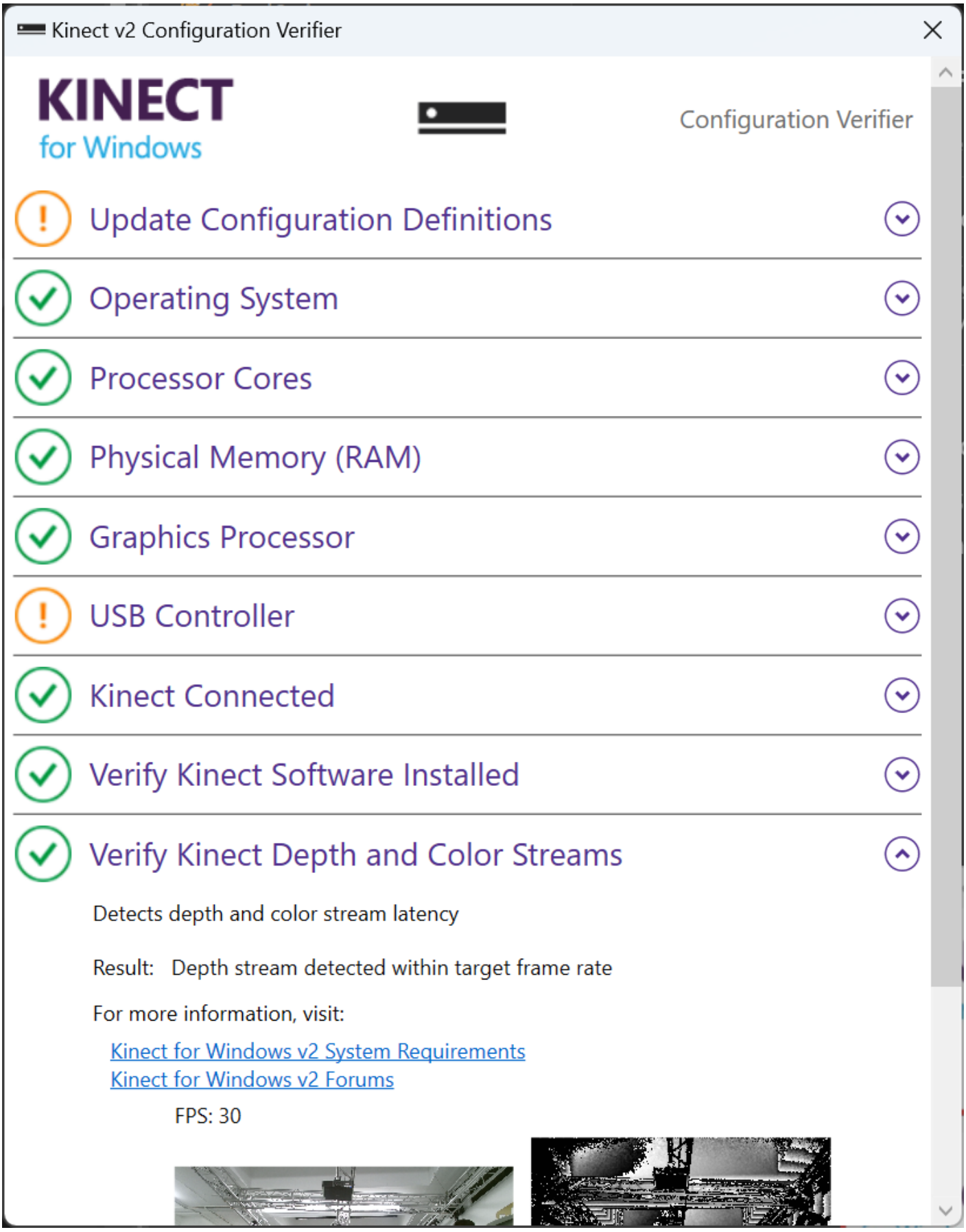
After you install the SDK, connect your Kinect to your computer, make sure the power supply is connected, the usb cable only transfer data.

Head to SDK Browser v2.0 to find the SDKs and Documentation. You don't need to touch the SDKs to use Kinect, but the installation is required for other application to run on it.

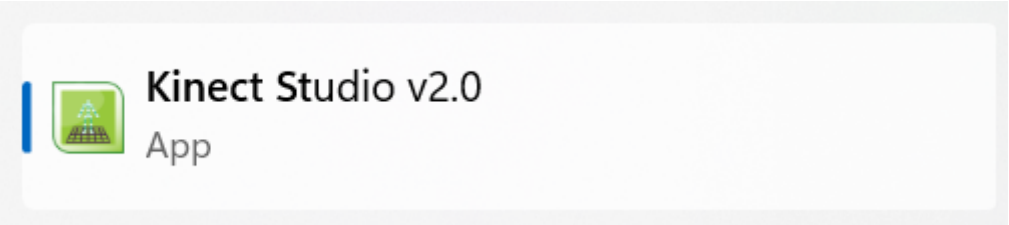


Use the Kinect Configuration Verifier to check if Kinect is working properly. It may take some time to run, if you can see the color and depth image in the last section, then everything is all set now.





You can view your live Kinect feed with Kinect Studio 2.0



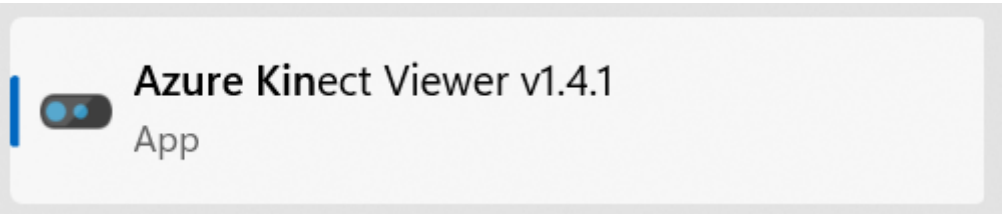
Azure Kinect

Azure Kinect SDK can be found on GitHub, follow the instruction to download and install the latest version.

GitHub - microsoft/Azure-Kinect-Sensor-SDK: A cross platform (Linux and Windows) user mode SDK to read data from your Azure Kinect device.

Connect Azure Kinect to your computer. Azure Kinect could be powered with a standalone power supply or directly from usb-c. Make sure you use the bundled usb-c cable or a quality cable that meets the power delivery and data speed requirement.

Verify the connection and view the live feed from Azure Kinect Viewer.



Troubleshooting

Kinect don’t show up as a device/ Couldn’t connect to the Kinect

- Check your usb connection
- Check if Kinect is connected to power
- Try a different usb cable that is known good for data and power

** The light on Kinect only turns on when there’s application actively using the device

Kinect for Windows connects, but looses connection/reboot every couple minutes

Go to your system sound settings and find Microphone Array Xbox NUI Sensor. Make sure this device is allowed for audio. If not allowed, Kinect won’t initialize properly and try to reboot every minute.

System › Sound › Properties

Microphone Array

3- Xbox NUI Sensor

Rename

Provider	(Generic USB Audio)
Driver date	16/04/2025
Driver version	10.0.22621.5262

Check for driver updates

General

Audio
Allow apps and Windows to use this device for audio

Don't allow

Use Kinect in TouchDesigner

TOP and Chop

There are generally two different ways of utilizing Kinect in TouchDesigner, with Kinect TOP and Kinect Chop. `Kinect TOP` offers all image sensor data in a TOP. `Kinect CHOP` provides skeletal tracking data through different channels in a CHOP.

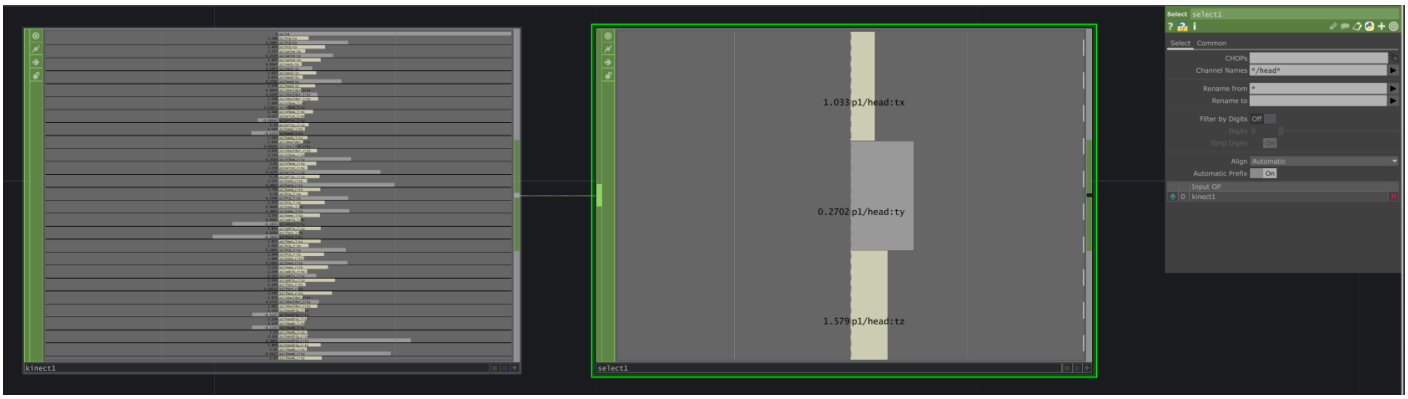
TOP

The image data can be accessed through `Kinect TOP` or `Kinect Azure TOP` respectively.

Below is an example of using creating a colored point cloud from the depth and color sensing image from Kinect. The project file for this example could be found here.

CHOP

`Kinect CHOP` offers the skeletal tracking data for each joint and its x/y/z position in different coordinate space through different channels. Use `Select CHOP` to select the channel you need, Pattern Matching is helpful in filtering and selecting multiple channels with conditions.

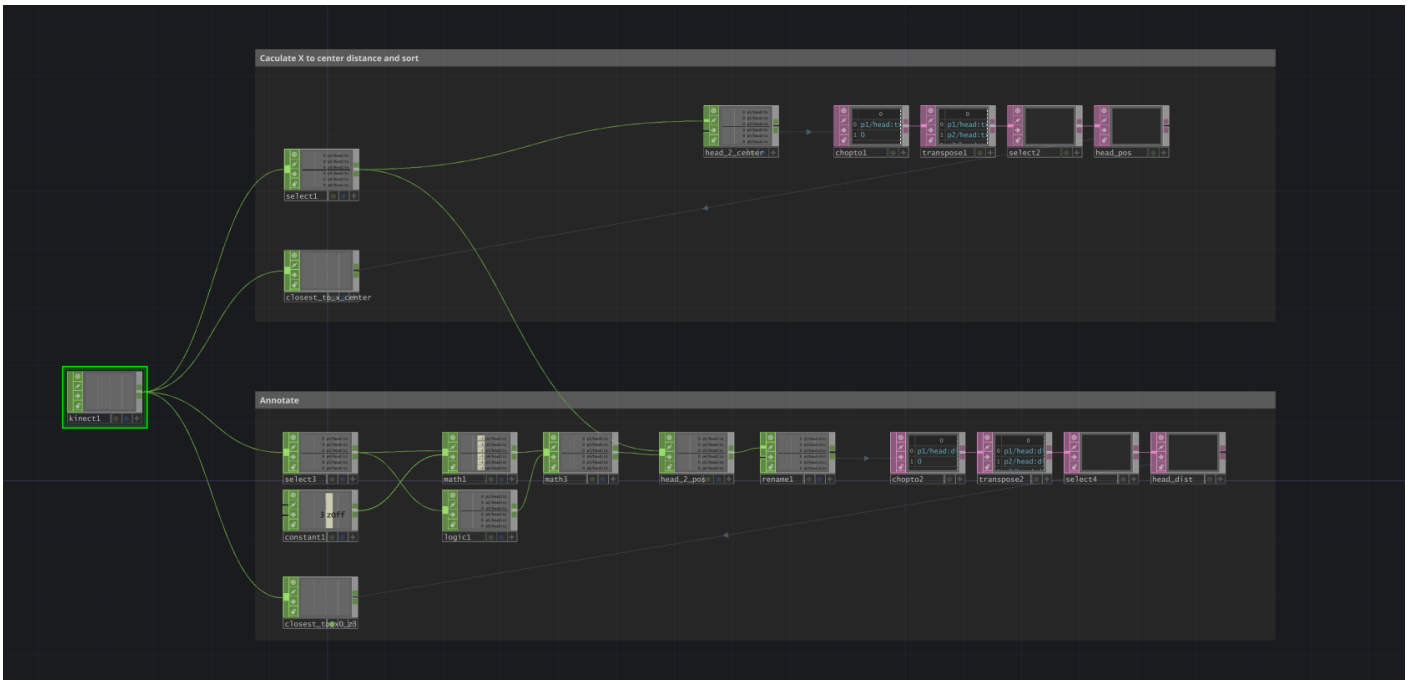


Multi-player Tracking

A common problem of using Kinect skeletal tracking in an installation is about finding the right person to track. When you need to track only one person, setting `Max Player` to 1 will pass the handling of player selection to Kinect, and most of the time it will lock on to the closest person and there is no way to manually switch player. When there are more people in the Kinect tracking space in an installation, this could be an problem.

A good approach is to keep the `Max Player` at Max, and create cuntom logic to filter and select the player you wanna track. Every time Kinect detects a new player in the frame, they will be assigned a `p*/id`. You can use id to keep tracking locked on the same person, no matter the position and how many players are in frame. For each player, you can use the x/y/z positions from the head joint (or any other joint) `p*/head:*` to caculate its reletive distance to any point in space. And use math to draw a boundary or create sorting logic, you can map a physical point in real-world to a point in Kinect coordinate space, so Kinect only use the tracking from the person standing right on the point.

Below is an example of selecting a player based on the realtive position to the vertical center of the frame `x = 0` and a point `x = 0, z = 3`. The project file could be found here



Use Kinect Anywhere

Stream Data

Althrough only a limited set of software, for example TouchDesigner, offers native support for Kinect over the Kinect SDK, you can stream Kinect data from one of these applications to almost anywhere, including in different environment like Python, Node.js or even Web Apps including p5.js.

Within a local network, you can effectively setup one Windows computer as the Kinect Host and stream data to the same machine or different machines, utilizing Kienct on Raspberry Pi, MacOS or even Arduino.

Depending on the use case, you can stream the raw camera data as a video feed, or stream the skeletal tracking data as a live data feed. TouchDesigner is good as the host to do both.

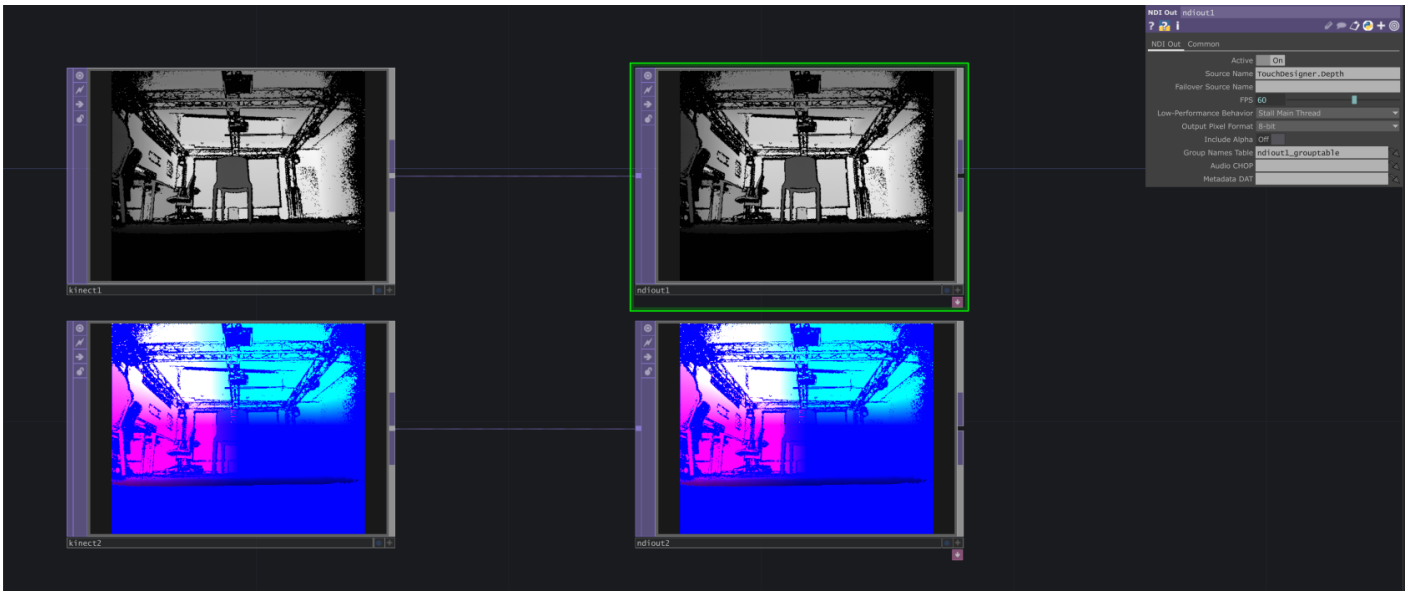
Stream Image Feed via NDI

NDI (Network Device Interface) is a real-time video-over-IP protocol developed by NewTek. It's designed for sending high-quality, low-latency video and audio over a local network (LAN), with minimal setup and high performance. Read NDI Documentation for more infomation.

You can use NDI to stream Kinect video feeds (color, depth, IR) from TouchDesigner to:

- Another TouchDesigner instance (on same or different machine)
- OBS Studio for recording or streaming
- Unreal Engine, Unity, Max/MSP, etc.
- Custom apps using NDI SDK or NDI-compatible plugins

Setup NDI stream in TouchDesigner with `NDI OUT CHOP`, you can create different streams for different image source (TOP) with different names.



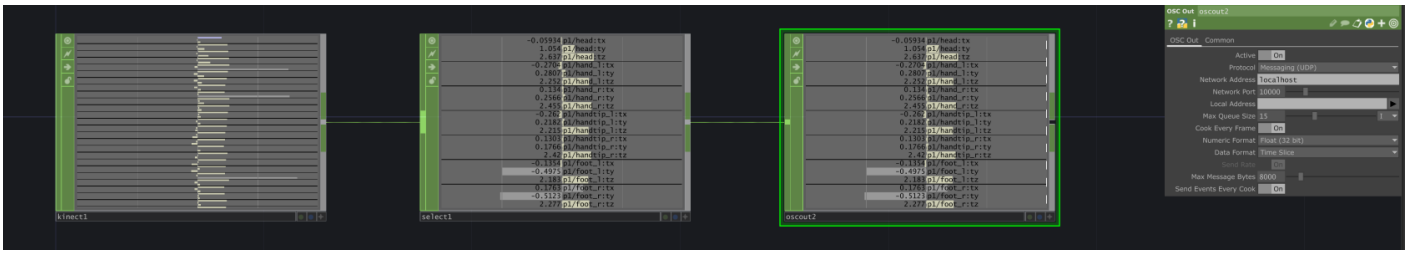
Stream Body Tracking Data via OSC

OpenSoundControl (OSC) is a data transport specification (an encoding) for realtime message communication among applications and hardware over network, typically TCP or UDP. OSC is originally created for highly accurate, low latency, lightweight, and flexible method of communication for use in realtime musical performance and is widely used in music, art, motion tracking, lighting, robotics, and more.

You can use OSC to stream body tracking data from Kinect in TouchDesigner to other software (or vice versa), such as:

- Unity, Unreal Engine
- Processing, openFrameworks
- Max/MSP, Pure Data, Isadora
- Web apps (via websocket bridge)
- Python, Node.js apps
- Other TouchDesigner instance

Send OSC data from TouchDesigner with `OSC Out CHOP` or `OSC Out DAT`. Use `CHOP` when sending multiple channels straight from a chop; Use `DAT` with custom python script to further manipulate and format the data before sending.



Stream with Kinectron for Web

OSC communication is typically implemented with UDP, which is fast and easy for native application to send and receive over the same local network. However web application in the browser runs in an isolated sandbox and does not have the access to local UDP port. To get data into your web application, you need a bridge for communicating with your web app through WebSocket or WebRTC.

Kinectron enables real-time stream of Azure Kinect Data to web browser. Visit Kinctron Release page to download the latest version of the server side application and client side library for using the data.

** Notice that Kinectron V1.0.0 only support Azure Kinect. Support for Kinect Windows V2 could be found on the older version 0. Find more about Kinectron V0 and usage examples.

Receiving data

OSC is widely supported in a range of applications and programming languages. Find the package or library to receive OSC data or you can bind a socket to the UDP port then listen and parse any OSC message.

Unity

Script and examples for receiving OSC message

https://t-o-f.info/UnityOSC/

Unreal

Unreal has built-in OSC Plugin, include the plugin from plug-in manager and start with a blutprint. Find the documentation

OSC Plug-in Overview

Processing

Use oscP5 library for processsing

oscP5

openFrameworks

openFrameworks has add-on to support OSC natively, find the documentation

ofxOsc Documentation

MaxMSP and Pure Data

Use `udpsend` and `udpreceive` objects to send and receive osc message

Ableton

Use Connection Kit in Max4Live to send and receive OSC data in Ableton. More info in

Connection Kit

Python

There are multiple packages available for osc in python.

One example is `python-osc`. Install with `pip install python-osc` and find the documentation python-osc

Node.js

There are multiple osc packages in node as well.

One example is `osc`, install with `npm intall osc` and find the documentation osc.js

With `osc.js`, you can also create a WebSocket bridge that forward osc messages to browser application.

Browser /Web Application

To use Kinect data in the browser, there are two options

- Use Kinectron to stream data and receive with Kinectron client-side script, full support for video stream and body tracking data
- Use `osc.js` in Node.js to create WebSocket bridge and forward selective tracking data through websocket. Use JavaScript native WebSocket Client API to receive the data

Use Kinect Directly in Unreal and Unity

In game engines, you should be able to use the Kinect SDK directly, which still envolves some lower level programing and a lot of experience. There are some plugins developed for Kinect, but some of them are paid and some haven't been updated for years. Look for the right plugin for your need, depending on whether you want to get Skeletal Tracking or Depth Image.

Unreal

Neo Kinect (paid)

Azure Kinect for Unreal Engine

Unity

Azure Kinect and Femto Bolt Examples for Unity (paid)

Unity_Kinect (free)