

# 3D Through - Unity Plugin Manual

---

**Version:** 1.0

**Author:** masterrey

**Contact:** masterrey@gmail.com

**License:** Apache License 2.0

**GitHub:** <https://github.com/masterrey/3DThrough>

**Open Source:** Yes

---

## Table of Contents

---

1. [Introduction](#)
  2. [System Requirements](#)
  3. [Installation](#)
  4. [Quick Start Guide](#)
  5. [Plugin Features](#)
  6. [User Interface](#)
  7. [Operating Modes](#)
  8. [Button Controls](#)
  9. [Script Reference](#)
  10. [Troubleshooting](#)
  11. [Support](#)
-

# 1. Introduction

---

**3D Through** is a Unity Editor plugin that provides seamless integration with 3DConnexion SpaceMouse devices. It allows you to control the Unity Scene View camera and manipulate selected objects using intuitive 6-degree-of-freedom (6DOF) input from your 3D mouse.

## Key Features

- Control Scene View camera with 6DOF input
  - Manipulate selected GameObjects directly
  - Customizable sensitivity settings
  - Direct access to raw sensor data for custom implementations
  - Multiple viewing modes and shortcuts
  - Robust error handling and connection management
- 

# 2. System Requirements

---

## Hardware Requirements

- 3DConnexion compatible device (SpaceMouse, Space Navigator, Space Pilot, etc.)
- Windows operating system (COM driver requirement)

## Software Requirements

- Unity Editor (2019.4 or later recommended)
- 3DConnexion driver installed and running
- TDxInput COM driver properly registered

## Supported 3D Mice

- SpaceMouse Compact
  - SpaceMouse Wireless
  - SpaceMouse Pro
  - SpaceMouse Enterprise
  - Space Navigator
  - And other 3DConnexion devices
- 

## 3. Installation

---

### Step 1: Install 3DConnexion Drivers

1. Download the latest stable 3DConnexion driver from: <https://www.3dconnexion.com/service/drivers.html>
2. Install the driver following the manufacturer's instructions
3. Connect your 3D mouse and verify it's working using the 3DConnexion control panel

### Step 2: Import the Plugin

1. Copy the `3DTrough` folder to your project's `Assets` directory
2. Unity will automatically import the plugin and compile the scripts
3. Verify no compilation errors appear in the Console window

### Step 3: Open the Plugin Window

1. In Unity, go to the menu: **Tools > 3D Through**
2. The 3D Through window will open
3. If your device is connected, you should see "3DConnexion device connected!" in the Console

## Verification

- The plugin window should appear without errors
  - Your 3D mouse should be recognized (green status indicator)
  - Moving the 3D mouse should move the Scene View camera
- 

## 4. Quick Start Guide

---

### First Time Setup

1. **Open the 3D Through window**
2. Navigate to **Tools > 3D Through** in the Unity menu bar
3. Keep this window open while working (it can be docked anywhere)
4. **Test Camera Control**
5. With the 3D Through window open, move your 3D mouse
6. The Scene View camera should respond to your movements
7. Push/pull for forward/backward movement
8. Twist for rotation
9. **Adjust Sensitivity**
10. Use the "Rot sens" slider to adjust rotation sensitivity (0.0 - 1.0)
11. Use the "Trans sens" slider to adjust translation sensitivity (0.0 - 1.0)
12. Settings are automatically saved between sessions
13. **Try Object Mode**
14. Select a GameObject in the Hierarchy
15. Click the "Object" toggle or press Button 2 on your 3D mouse
16. Now the device will control the selected object instead of the camera

---

## 5. Plugin Features

---

### Camera Control Mode (Default)

- Navigate the Scene View freely using your 3D mouse
- Smooth, intuitive 6DOF camera movement
- Independent translation and rotation sensitivity controls
- Ideal for scene inspection and level design

### Object Manipulation Mode

- Control selected GameObjects directly
- Position and rotate objects with precision
- Uses Undo system - all changes can be reverted
- Perfect for detailed object placement

### Brute Axis Mode

- Exposes raw sensor data through static properties
- Access translation and rotation values in your own scripts
- Useful for custom implementations and gameplay mechanics
- No Scene View manipulation when active

### Automatic Reconnection

- Plugin automatically attempts to reconnect if device disconnects
  - Clear status indicators show connection state
  - Manual retry button available in the UI
-

## 6. User Interface

---

The 3D Through window provides the following controls:

### Status Indicators

- **Green**: Device connected and working
- **Yellow/Orange**: Device not found or disconnected
- **Connection status message**: Shows current device state

### Control Toggles

- **Object**: Toggle between Camera mode and Object mode
- **Brute Axis**: Enable raw data mode for custom scripting

### Sensitivity Sliders

- **Rot sens**: Rotation sensitivity (0.0 - 1.0)
  - Lower values = slower, more precise rotation
  - Higher values = faster, more responsive rotation
- **Trans sens**: Translation sensitivity (0.0 - 1.0)
  - Lower values = slower, more precise movement
  - Higher values = faster, more responsive movement

### Action Buttons

- **Retry Connection**: Attempt to reconnect to the 3D mouse
  - Appears when device is not connected

# 7. Operating Modes

---

## 7.1 Camera Mode (Default)

**How to activate:** Default mode when window opens, or toggle off "Object" mode

**Behavior:** - 3D mouse controls the Scene View camera - Translation moves the camera through 3D space - Rotation changes the viewing angle - Works in both Perspective and Orthographic views

**Best for:** - Navigating scenes - Inspecting level design - Reviewing 3D models - General scene exploration

## 7.2 Object Mode

**How to activate:** - Click the "Object" toggle in the plugin window, OR - Press Button 2 on your 3D mouse

**Behavior:** - 3D mouse controls the selected GameObject - Translation moves the object's position - Rotation changes the object's orientation - Changes are recorded in Unity's Undo system - Automatically switches back to Camera mode if nothing is selected

**Best for:** - Precise object positioning - Rotating objects to exact angles - Fine-tuning scene layouts - Placing decorative elements

## 7.3 Brute Axis Mode

**How to activate:** - Click the "Brute Axis" toggle in the plugin window, OR - Press Button 5 on your 3D mouse

**Behavior:** - Disables automatic camera/object control - Exposes raw sensor values through `Mouse3D.E3dThrough` class - Values update every frame - Scene View control is disabled in this mode

**Best for:** - Custom implementations - Runtime gameplay mechanics - Creating your own movement scripts - Debugging sensor values

### **Accessing Values:**

```
using Mouse3D;

// Get translation (movement) values
Vector3 movement = E3dThrough.BruteTranslation;

// Get rotation values
Vector3 rotation = E3dThrough.BruteRotation;
```

---

## **8. Button Controls**

---

The plugin supports various button shortcuts on your 3D mouse:

<b>Button</b>	<b>Function</b>	<b>Description</b>
2	Toggle Object/Camera Mode	Switches between controlling the camera or selected object
3	Helicopter Mode	Changes rotation origin to camera position
4	Toggle Perspective/Ortho	Switches Scene View between perspective and orthographic
5	Toggle Brute Axis Mode	Enables/disables raw data mode
7	Top View	Changes to top-down view of selected object
8	Left View	Changes to left side view of selected object
9	Right View	Changes to right side view of selected object
10	Front View	Changes to front view of selected object
11	Fit View	Frames the selected object (configurable in mouse settings)

**Note:** Button availability depends on your specific 3D mouse model. Some compact models may have fewer buttons.

---

# 9. Script Reference

---

## 9.1 Namespace

All plugin scripts are contained in the `Mouse3D` namespace:

```
using Mouse3D;
```

## 9.2 E3dThrough Class

Static class that provides access to raw 3D mouse sensor data when Brute Axis mode is enabled.

### Properties

**BruteTranslation** (static Vector3) - Raw translation values from the 3D mouse - Represents movement on X, Y, and Z axes - Values are not scaled or processed - Only updates when Brute Axis mode is active

```
Vector3 translation = E3dThrough.BruteTranslation;
```

**BruteRotation** (static Vector3) - Raw rotation values from the 3D mouse with sensitivity applied - Represents rotation on X, Y, and Z axes - Values are multiplied by the configured sensitivity - Only updates when Brute Axis mode is active

```
Vector3 rotation = E3dThrough.BruteRotation;
```

## 9.3 Example Scripts

### Example 1: Moving an Object with Brute Axis

```
using UnityEngine;
using Mouse3D;

public class MoveWithMouse3D : MonoBehaviour
{
    void Update()
    {
        // Move object based on 3D mouse translation
        transform.position += E3dThrough.BruteTranslation * Time.deltaTime;
    }
}
```

**Usage:** 1. Enable Brute Axis mode in the 3D Through window 2. Attach this script to any GameObject 3. The object will move according to your 3D mouse input

### Example 2: Rotating an Object with Brute Axis

```
using UnityEngine;
using Mouse3D;

public class RotateWithMouse3D : MonoBehaviour
{
    public float rotationSpeed = 10f;

    void Update()
    {
        // Rotate object based on 3D mouse rotation
        Vector3 rotation = E3dThrough.BruteRotation * rotationSpeed * Time.deltaTime;
        transform.Rotate(rotation);
    }
}
```

### Example 3: Combined Movement and Rotation

```
using UnityEngine;
using Mouse3D;

public class FullControl3D : MonoBehaviour
{
    public float moveSpeed = 1f;
    public float rotateSpeed = 10f;

    void Update()
    {
        // Move the object
        Vector3 movement = E3dThrough.BruteTranslation * moveSpeed * Time.deltaTime;
        transform.position += movement;

        // Rotate the object
        Vector3 rotation = E3dThrough.BruteRotation * rotateSpeed * Time.deltaTime;
        transform.Rotate(rotation);
    }
}
```

## Example 4: Conditional Control

```
using UnityEngine;
using Mouse3D;

public class ConditionalControl : MonoBehaviour
{
    public KeyCode activationKey = KeyCode.Space;
    public float sensitivity = 1f;

    void Update()
    {
        // Only respond to 3D mouse when key is held
        if (Input.GetKey(activationKey))
        {
            transform.position += E3dThrough.BruteTranslation * sensitivity;
        }
    }
}
```

## 9.4 Best Practices

1. **Always enable Brute Axis mode** when using `E3dThrough` in your scripts
2. **Multiply by Time.deltaTime** for frame-rate independent movement
3. **Add sensitivity multipliers** to fine-tune control feel
4. **Check for zero values** before applying transformations to avoid unnecessary updates
5. **Keep the 3D Through window open** for the plugin to function

# 10. Troubleshooting

---

## Device Not Connecting

**Symptom:** Yellow warning in plugin window, "Device could not be initialized" message

**Solutions:** 1. Verify 3DConnexion drivers are installed and running 2. Check device is connected and powered on 3. Test device in 3DConnexion control panel 4. Restart Unity Editor 5. Click "Retry Connection" button in plugin window 6. Restart 3DConnexion driver service

## Plugin Window Not Responding

**Symptom:** Moving 3D mouse has no effect

**Solutions:** 1. Ensure 3D Through window is open (**Tools > 3D Through**) 2. Check connection status in plugin window 3. Verify Scene View is visible and active 4. Try closing and reopening the plugin window 5. Restart Unity if problem persists

## COM Exception Errors

**Symptom:** "COMException" or "Class not registered" errors in Console

**Solutions:** 1. Install latest stable 3DConnexion drivers 2. Re-register TDxInput COM component: - Run Command Prompt as Administrator - Navigate to driver installation folder - Run: `regsvr32 TDxInput.dll` 3. Restart computer 4. Reinstall 3DConnexion drivers if problem persists

## Object Mode Not Working

**Symptom:** Enabling Object mode has no effect

**Solutions:** 1. Ensure a GameObject is selected in the Hierarchy 2. Check the object is not locked in the Inspector 3. Verify the object's Transform is not driven by other scripts 4. Try selecting a different object

## Sensitivity Too High or Low

**Symptom:** Movement is too fast/slow or imprecise

**Solutions:** 1. Adjust "Rot sens" and "Trans sens" sliders in plugin window 2. Recommended starting values: 0.1 - 0.3 3. Settings are saved automatically 4. For Brute Axis mode, add sensitivity multipliers in your scripts

## Performance Issues

**Symptom:** Unity becomes slow or unresponsive

**Solutions:** 1. Reduce sensitivity values 2. Close other 3DConnexion applications 3. Update to latest drivers 4. Disable Brute Axis mode if not needed

---

# 11. Support

---

## Getting Help

If you encounter issues not covered in this manual:

1. Check Unity Console for error messages
2. Review this Troubleshooting section
3. Verify your setup meets System Requirements
4. Test your 3D mouse in other applications

## Contact

For questions, bug reports, or feature requests:

**Email:** masterrey@gmail.com

When contacting support, please include: - Unity version - 3D mouse model - Driver version - Error messages from Console - Steps to reproduce the issue

## Updates

To receive notifications about plugin updates and new features:

- Watch the GitHub repository: <https://github.com/masterrey/3DThrough>
- Contact the author via email

## Open Source

This plugin is now open source! You can:

- View the source code on GitHub
- Report issues and bugs
- Contribute improvements
- Fork for your own modifications

**Repository:** <https://github.com/masterrey/3DThrough>

---

## License

---

This plugin is licensed under the Apache License 2.0.

Copyright 2024 masterrey

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

---

**Thank you for using 3D Through!**