Complete Manual: Miniconda + TensorFlow GPU Setup (Windows)

This manual explains how to: - Install Miniconda - Create a Python 3.10 environment - Install CUDA (via Conda) - Install TensorFlow GPU (2.10) - Connect environment to VS Code - Test GPU detection - Delete environment safely

---

STEP 1: Install Miniconda

1. Download Miniconda (Windows 64-bit installer).
2. Run the installer.
3. Choose:
4. Just Me
5. Add Miniconda to PATH (important)
6. Finish installation.

After installation, open **Anaconda Prompt** and verify:

```
conda --version
```

If version appears, installation is successful.

STEP 2: Create Python 3.10 Environment

In Anaconda Prompt:

```
conda create -n tf_gpu python=3.10
```

Press `y` to confirm.

Activate environment:

```
conda activate tf_gpu
```

You should now see `(tf_gpu)` in the terminal.

This environment is completely isolated and does NOT affect your system Python.

STEP 3: Install CUDA Toolkit & cuDNN (Inside Environment)

Install required versions for TensorFlow 2.10:

```
conda install -c conda-forge cudatoolkit=11.2 cudnn=8.1.0
```

Press `y` to confirm.

Note: CUDA is installed inside the environment. No manual NVIDIA CUDA installation required.

STEP 4: Install TensorFlow GPU Version

```
pip install tensorflow==2.10
```

Wait for installation to complete.

STEP 5: Test GPU Detection

Run Python:

```
python
```

Then execute:

```
import tensorflow as tf
print(tf.__version__)
print(tf.config.list_physical_devices('GPU'))
```

If working correctly, output should show:

```
[PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU')]
```

Exit Python:

```
exit()
```

STEP 6: Connect Environment to VS Code

1. Open VS Code.
2. Open your project folder.
3. Press:

Ctrl + Shift + P

1. Type:

Python: Select Interpreter

1. Select:

tf_gpu (Conda)

If not visible:

- Choose "Enter interpreter path"
- Browse to:

miniconda3 > envs > tf_gpu > python.exe

Select it.

Now VS Code uses this environment.

STEP 7: Verify Inside VS Code

Open VS Code terminal and run:

```
python
```

Then:

```
import tensorflow as tf
print(tf.config.list_physical_devices('GPU'))
```

If GPU appears, setup is complete.

STEP 8: Check NVIDIA Driver

In normal Command Prompt (not conda):

```
nvidia-smi
```

If GPU information appears, drivers are installed correctly.

HOW TO USE ENVIRONMENT DAILY

Before working:

```
conda activate tf_gpu
```

After finishing:

```
conda deactivate
```

HOW TO DELETE THE ENVIRONMENT

If you want to remove everything safely:

```
conda deactivate
conda remove -n tf_gpu --all
```

Press `y` to confirm.

This deletes the environment completely. System Python remains untouched.

WHERE ENVIRONMENTS ARE STORED

Usually located at:

```
C:\Users\YourName\miniconda3\envs
```

The folder `tf_gpu` contains your environment.

IMPORTANT NOTES

- This setup does NOT modify global Python.
- You can create multiple environments.
- You can delete and recreate anytime.
- Always activate the environment before running TensorFlow.

Setup Complete

You now have a clean, isolated TensorFlow GPU environment using Miniconda on Windows.