
Intelligent Control Systems

Programming Environment Setup

Shingo Kagami

Graduate School of Information Sciences,

Tohoku University

swk(at)ic.is.tohoku.ac.jp

<http://www.ic.is.tohoku.ac.jp/ja/swk/>

Course Schedule

In the next four weeks, we cover the basics of visual processing, which constitute an important building block for vision-based machine control.

July 8: Image Processing (1)

July 15: Image Processing (2)

July 22: Visual Tracking (1)

July 29: Visual Tracking (2)

- Sample codes in Python are provided (and in the final assignment, you will be requested to develop your own image processing project and submit a report about it, although you can choose another programming language if you like to).
- Because setting up the programming environment consumes some time and generate a large network traffic, it is recommended to finish the setup before the lecture starts.
- If you are not familiar with Python, quickly skimming through the following page may help:

https://web.tohoku.ac.jp/kc_kyomu/computer_seminar1/py/textbook_en/tutorial.html


https://web.tohoku.ac.jp/kc_kyomu/computer_seminar1/py/textbook_ja/tutorial.html

Recommended Step (1): for Windows 64-bit

Install WinPython (storage usage: ~ 80 MB)

- Open <https://github.com/winpython/winpython/releases>
- Download Winpython64-3.10.4.0dot.exe (26 MB)
- Execute the downloaded exe file
- Move the generated folder to **C:¥ic2022¥WPy64-31040**

arbitrary folder of your choice;
avoid including white spaces



Test running Python

- Open the folder **C:¥ic2022¥WPy64-31040**
- Run WinPython Command Prompt.exe
- A command prompt will open; run the following command there:

```
python -V
```

If you don't use Windows or don't want to use WinPython for some reasons, see Notes (2) in the last page.

Recommended Step (2)

Install Python packages (storage usage: ~ 400 MB)

- Run the following commands in the WinPython Command Prompt:

```
python -m pip install --upgrade pip
pip install opencv-python opencv-contrib-python
pip install ipython numba scipy matplotlib
```

Recommended Step (3)

Download samples codes and data (storage usage: ~ 20 MB)

- Open <https://github.com/shingo-kagami/ic.git>
- Click the green button “Code” and click “Download Zip”
- Extract the zip file
- Move the generated ic-master folder to `C:¥ic2022¥sample`

If you are a Git user, you can simply run the followings instead of the above:

```
cd C:¥ic2022
git clone https://github.com/shingo-kagami/ic.git sample
```

Test a sample code

- Run the following commands in the WinPython Command Prompt:

```
cd C:¥ic2022¥sample
python ic00_minimal.py
```

Optional Step (1)

This step installs Spyder, a Python development environment.

<https://www.spyder-ide.org/>

You can skip this if you already have another editor (e.g. Visual Studio Code, Vim, Emacs, etc.) to edit Python codes.

Install the Spyder environment (~ 500 MB)

- Run the following commands in the WinPython Command Prompt:

```
pip install spyder
```

Test running Spyder

- Run the following commands in the WinPython Command Prompt:

```
spyder
```

Optional Step (2)

This step installs PyTorch, a deep learning framework for Python. This is needed if you want to run a demonstration of object detection by YOLOv5.

Install PyTorch (storage usage: ~ 800 MB)

- Run the following commands in the WinPython Command Prompt:

```
pip install torch torchvision pandas pyyaml tqdm seaborn  
pip install "protobuf<=3.20.1"
```

Optional Step (3)

This step installs Dlib, a C++ library for various algorithms and its wrapper interface for Python. This is needed if you want to run one of face detection demonstrations. Note that installation of dlib requires a C++ compiler.

Install Visual Studio Community Edition and CMake (* see next page)

- <https://visualstudio.microsoft.com/> (~ 3 GB)
- <https://cmake.org/> (~ 100 MB)

Install Dlib (~ 10 MB)

- Run the following commands in the WinPython Command Prompt:

```
pip install dlib
```

Download the pretrained parameter files (~ 70 MB)

- Open <http://dlib.net/files/>
- Download the following files:
 - shape_predictor_5_face_landmarks.dat.bz2
 - shape_predictor_68_face_landmarks.dat.bz2
- Uncompress them and move those dat files to **C:\¥ic2022¥sample**

Notes (1): Installing Visual Studio and CMake

A standard way to install Visual Studio (Community Edition) and CMake is to download their installers and follow the instructions.

Another easy way to install them is to use Chocolatey:

- Execute Windows PowerShell as administrator
- Run the following commands:

```
Set-ExecutionPolicy Bypass -Scope Process -Force  
iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))  
choco install -y visualstudio2019community  
choco install -y cmake
```

Notes (2): Installing Python

If you are not a Windows user or you don't want to install WinPython for some reasons, another choice is to install miniconda. See:

<https://conda.io/projects/conda/en/latest/user-guide/install/index.html>

You may want to use yet another Python environment you are using every day. Of course it'll be possible and it's just up to you. But it is highly recommended that you create a separate environment for this course (e.g. by `conda create` or `python -m venv`).

Our sample codes have been tested under Python 3.9 and 3.10.