

Humanoid chess-playing robotic arm – Step-by-step User Guide

I. Physical parts

The physical setup is composed of the following parts, each followed by an image of it:

- A wooden bar about 1.07 meter long and 1.3 centimeters thick



- An Arducam USB camera to which is attached a support system



- A 3d-printed support for a spot light



- A USB cable extension



- A 50x50cm chess board



- Two tripods, each having a 3d-printed support system attached to it



For the system to function, an additional spotlight supporting the attachment system provided by the spotlight support is required, and may or may not be provided with the other components. For the first internship of this project, the system was tested using the following spotlight, borrowed to Cyril DOCQUIN, the campus life manager of CESI Lyon.



II. Assembling

To assemble the system, please follow these steps.

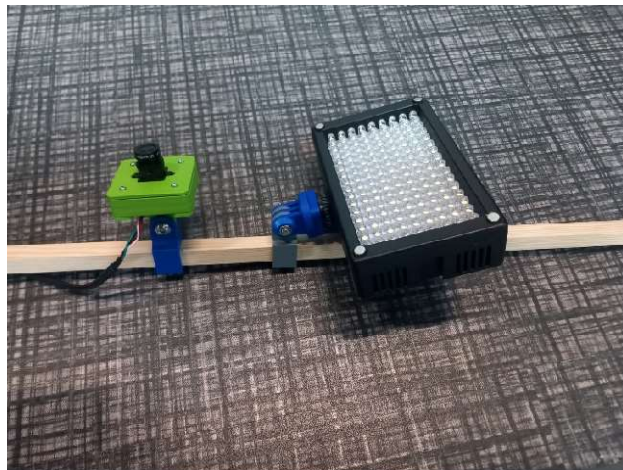
- 1) Attach the camera to the wooden bar as shown in the image below. Make sure that the camera is attached at the center of the wooden bar.



2) Attach the spotlight to the dedicated support as shown in the image below.



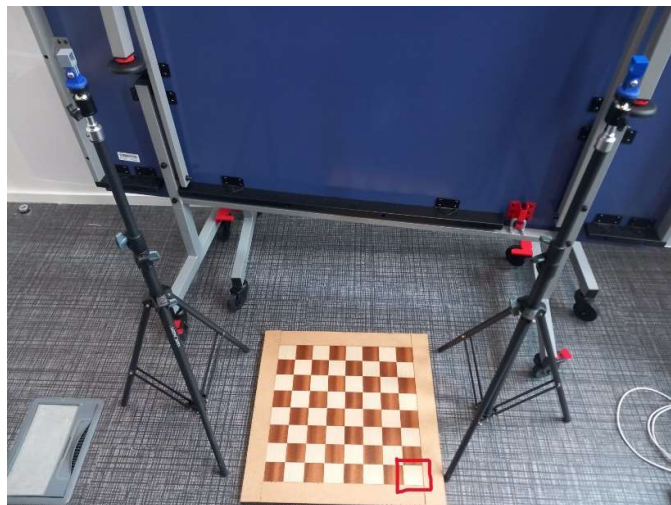
3) Attach this to the wooden bar next to the already present camera.



- 4) Place the two tripods as shown in the image below. Make sure that they are configured to the same height and that their support systems are aligned as in the image below.



- 5) Place the tripods next to the chessboard as shown below and make sure the bottom-right square of the board is a light square. The tripod's support systems must remain aligned in the same way as in step 4.



- 6) Mount the wooden bar system on the tripods using the latter's support systems to attach the pieces together. The result of this step should match what is shown below. Note the side on which the USB wire is connected to the camera. This is the side on which the white pieces will go.



- 7) Connect the USB cable extension to the camera's USB connector.



- 8) The physical setup is ready for use.

III. Using the standalone binary distribution

The binary distribution of the program is a windows executable that does not need Python to be installed on the host machine. It can be found on the GitHub page, under the “Releases” section. It is shipped as a zip archive containing the executable accompanied by the assets and dependencies it requires.

To run this distribution, simply unzip the archive and run the “hcpra.exe” file that is present inside.

For more information on the program’s runtime, see section *VI. Running the program*.

IV. Building from source

Here are a few steps to build the standalone binary distribution from source.

- 1) Make sure Python version 3.10 is installed on your machine.
- 2) Clone the GitHub repository
- 3) Run the “build.py” script with Python 3.10
- 4) The output should be present at “dist/hcpra”.

Note: Currently, only Windows is supported by this build script.

See *VI. Running the program* for how to use the program after having started it.

V. Preparing the environment

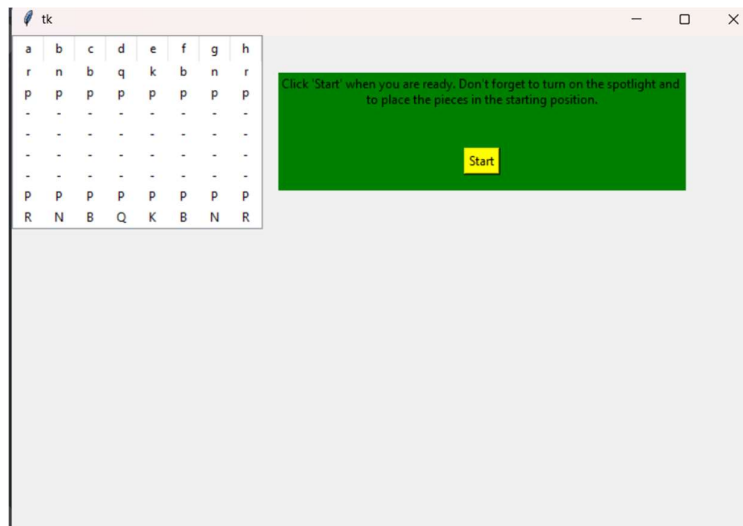
The following topic will provide you with an exhaustive way of preparing the software environment for running the program's source code directly.

- 1) Make sure you have Python version 3.10.X installed.
- 2) Make sure that the py launcher is installed on your system (see [4. Using Python on Windows — Python 3.13.3 documentation](#) for how to do this).
- 3) Make sure GIT is installed on your system.
- 4) Open the command prompt as administrator. Navigate to the directory where you want to download the program
- 5) Run the command "git clone <git-url>"
- 6) Navigate to the newly created directory, either from the windows explorer or from the currently opened command prompt.
- 7) Run "setup.bat" as administrator to install all the required packages.
- 8) To start the program, simply run the "run.bat" file. It is recommended not to run it as administrator.

VI. Running the program

- 1) Connect the USB camera to one of the USB ports of your machine.
- 2) Start the program. See sections III. or V. for details on how to do this.
- 3) Upon startup, the program take an image of the board and displays a window with that image. If identification of the chess board has succeeded, then the board's corners will be highlighted on that window. Otherwise, the program will crash and a ChessboardNotFoundError will be displayed to the console.

- 4) Close the window. Another window should open, with a menu and a 'start' button as shown below.



- 5) Install the pieces on the board in the starting position. Make sure the white pieces are placed on the side where the USB wire connects to the camera. The computer will be playing the black pieces.
- 6) Turn on the spot light.
- 7) Click the "Start" button.
- 8) Play your move.
- 9) Click "Turn". Make sure you are not obstructing the view of the camera.
- 10) The system will try to detect what move you have played.
- If it asks you to enter that move manually it means it has failed to detect your move.
 - Otherwise, you will be asked to confirm whether the move detected by the system is indeed the one you have played. If not, you will be asked to enter the move you have played manually.
- 11) The system chooses which move it wants to play, and asks you to perform that move. Once you have done so, click "Confirm". Make sure not to obstruct the view of the camera.
- 12) Go back to step 8 and repeat until the game is finished.

VII. Setting the ELO rating of the computer

Note: this topic requires some understanding of the Python programming language.

To tweak the ELO rating of the computer, first navigate to the source code folder. Then open the file “src/stockfish_implementation.py” for editing. There, the ELO_RATING variable can be modified. Refer to its inline documentation string for more information.

Note: modifying the ELO rating can only be done in the source code, and requires rebuilding the program from source for the changes to apply to the binary distribution.