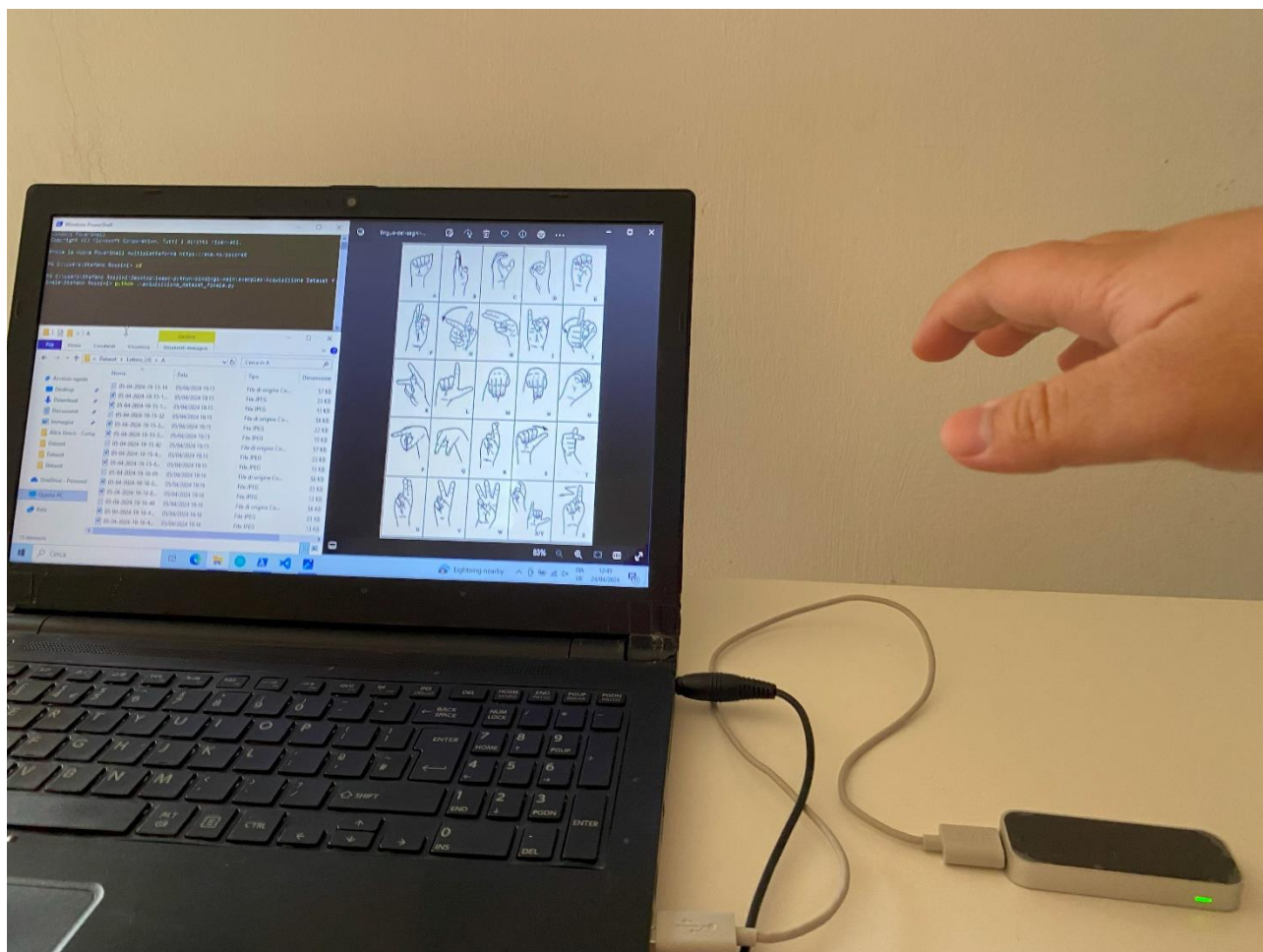


Acquisizione gesti con Leap Motion – Guida di installazione



Sommario

Prima di iniziare il tutto	3
Premessa	3
Obbiettivo	3
Requisiti fisici	3
Requisiti software	3
Altri tipi di requisiti:	3
Come installare il software	4
Gemini: Ultraleap	4
Leapc-python-bindings	8
Programma per salvare i dati delle mani	12
Preparativi – da fare solo una volta	12
Come eseguire il programma per salvare le informazioni delle mani – da ripetere per ogni candidato	16
Riconoscimento dei gesti offline	18
Obiettivo dell'esperimento	18
Dalla cattura all'elaborazione	18
Classification Learner	20
Conclusioni	22

Prima di iniziare il tutto

Premessa

Questa guida è stata scritta in data 17/04/2024; se la leggete dopo questa data, non è garantito che il software funzioni perché gli sviluppatori di Ultraleap potrebbero cambiare delle specifiche.

Per informazioni dettagliate, porre quesiti agli sviluppatori e/o alla community di Ultraleap, scrivete immediatamente sul server Discord della software house <https://discord.gg/3VCndThqxS> : entro qualche giorno qualcuno risponderà alle vostre domande

Obbiettivo

Acquisire dei movimenti della mano usando il Leap Motion e salvarli in un file csv

Requisiti fisici

- Leap Motion, con relativo cavo usb 3.0
- Computer con OS Windows, MacOS o Linux (la guida prenderà in esempio Windows 10)
- Connessione ad Internet, per scaricare il software necessario

Requisiti software

Per ordine d'installazione:

- Ultraleap Hand tracking
- Python

Altri tipi di requisiti:

- Pazienza e tempo (non sono così scontati)

Come installare il software

Gemini: Ultraleap

Per scaricare la suite di software per interagire con il Leap Motion, andate su <https://leap2.ultraleap.com/gemini-downloads/>, selezionate “Desktop/Laptop Computers”, cliccate su “Download now” e selezionare la versione del sistema operativo del computer che andrete ad utilizzare.

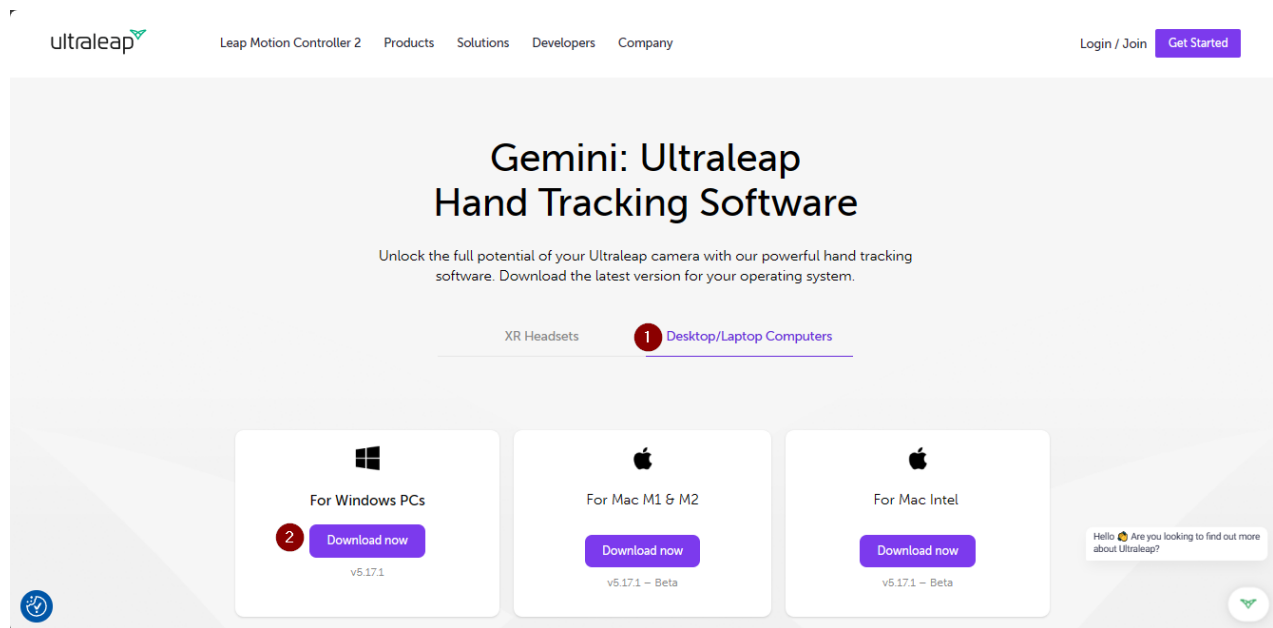


Figura 1 - Download page Gemini

Al momento del download, è necessario un account Ultraleap: se necessario, createvelo.

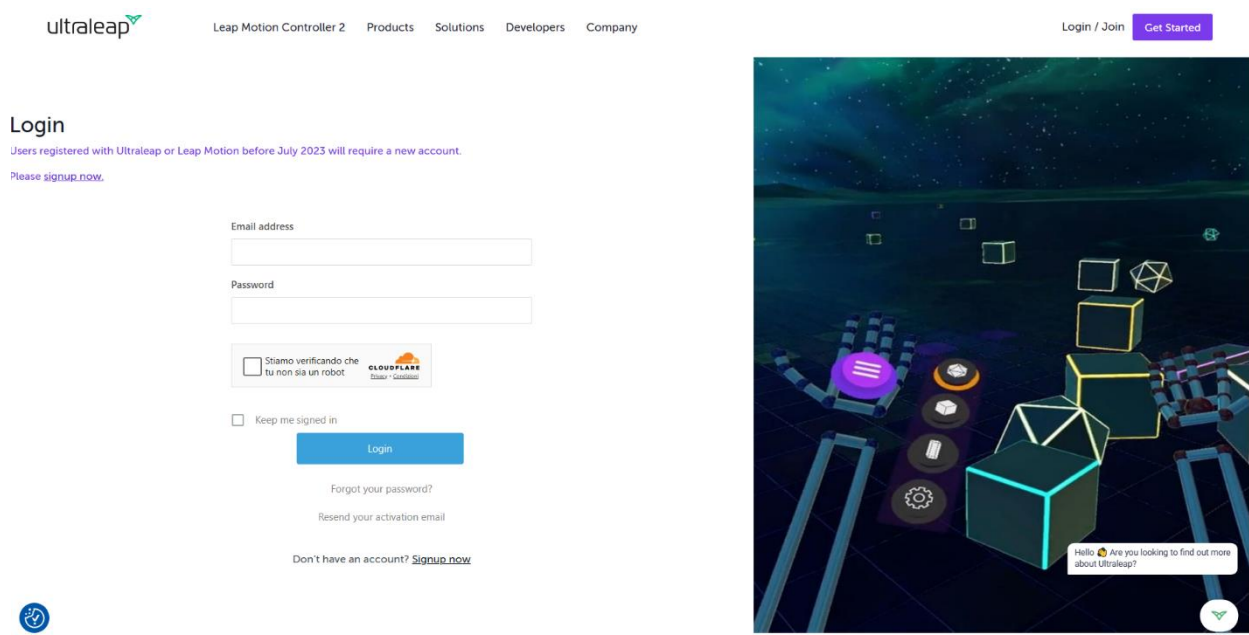


Figura 2 - Login page Ultraleap

Una volta scaricato, eseguite l'installazione del programma.

Al momento dell'installazione, non cambiate la directory d'installazione: se la cambiate annotatela perché poi la dovreste cambiare nell'installazione della libreria Python.

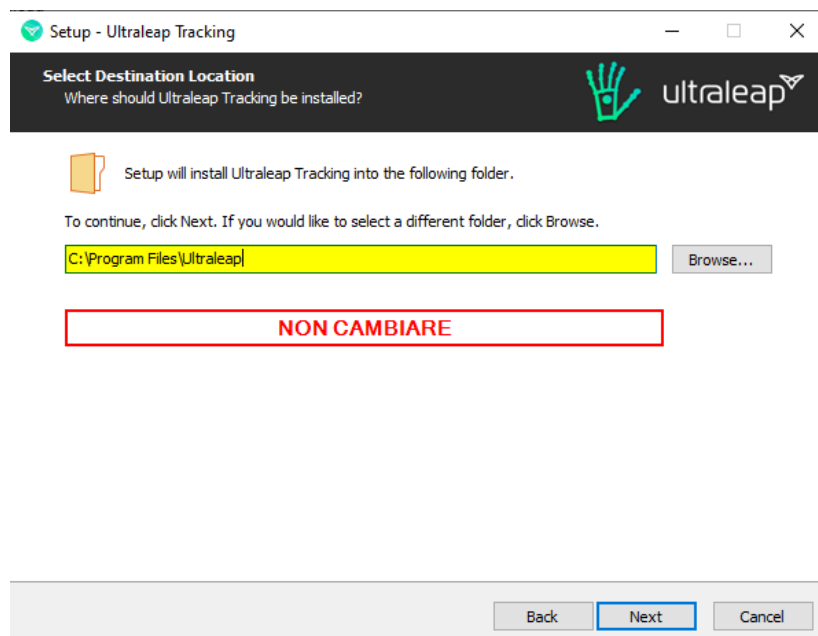


Figura 3 - Installazione Ultraleap

Scaricato il bundle Ultraleap Tracking, riavviare il computer.

Collegate il Leap Motion al pc usando il cavo usb 3.0 e la relativa porta; quindi, aprite il programma “Ultraleap Control Panel”.

Se non vi compare la schermata sottostante, verificate che i driver siano stati installati e/o aggiornate il firmware del Leap Motion.

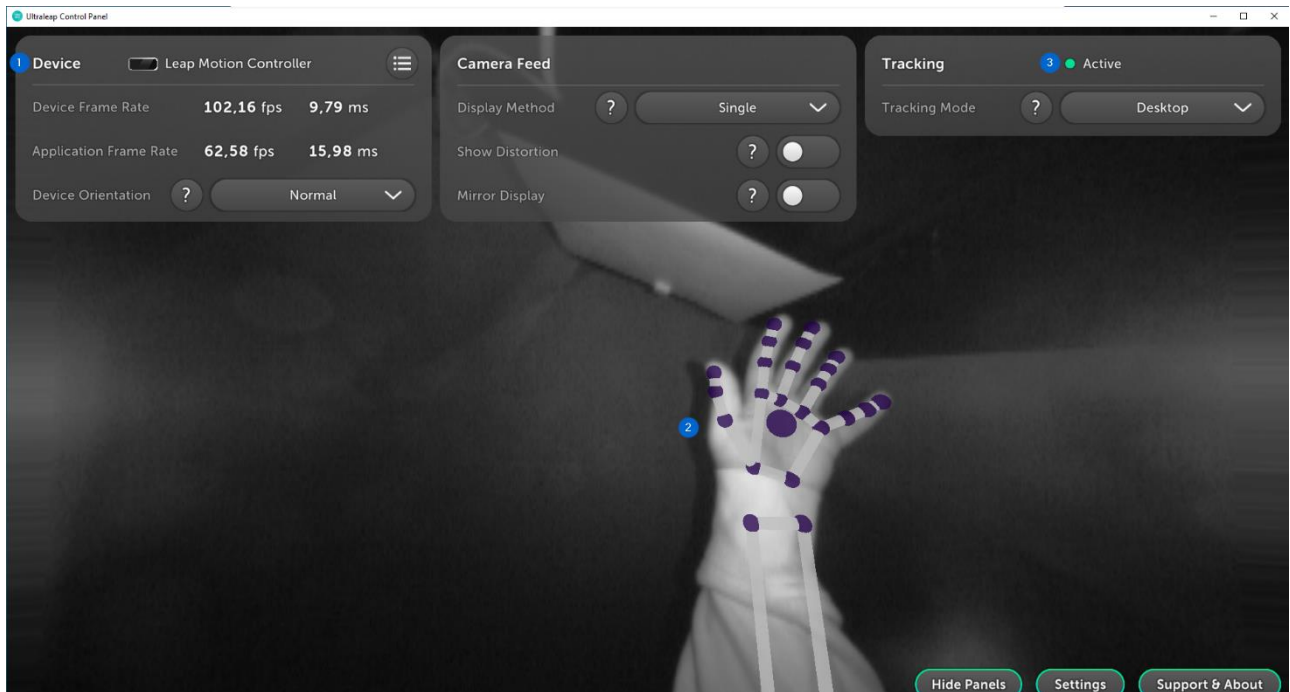


Figura 4 - Schermata Ultraleap Control Panel funzionante

Se vi comparisse il messaggio indicato in basso, il Leap Motion potrebbe essere rallentato e/o presentare problemi nell’acquisizione: cambiate pc o continuate con quello che avete sapendo di questo inconveniente.

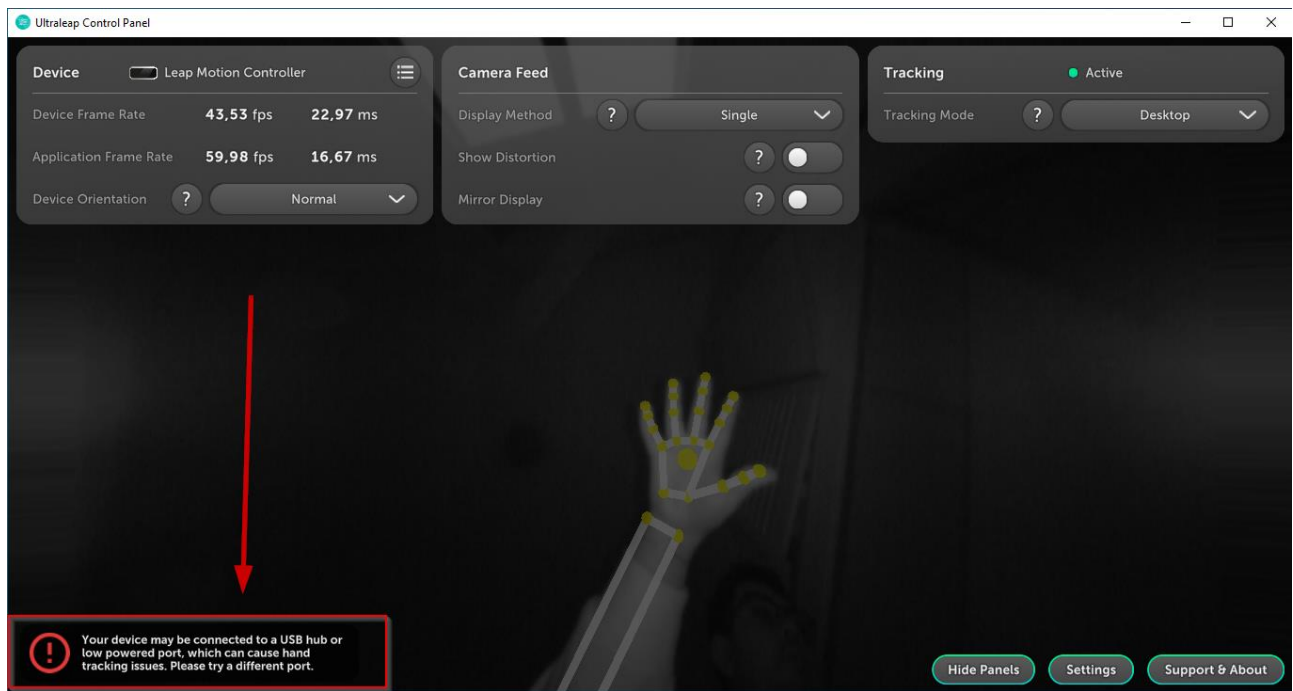


Figura 5 - Screenshot Leap Motion con poca batteria

Leapc-python-bindings

Ora che Ultraleap è scaricato, è necessario scaricare la libreria da <https://github.com/ultraleap/leapc-python-bindings> ed estrarre lo zip.

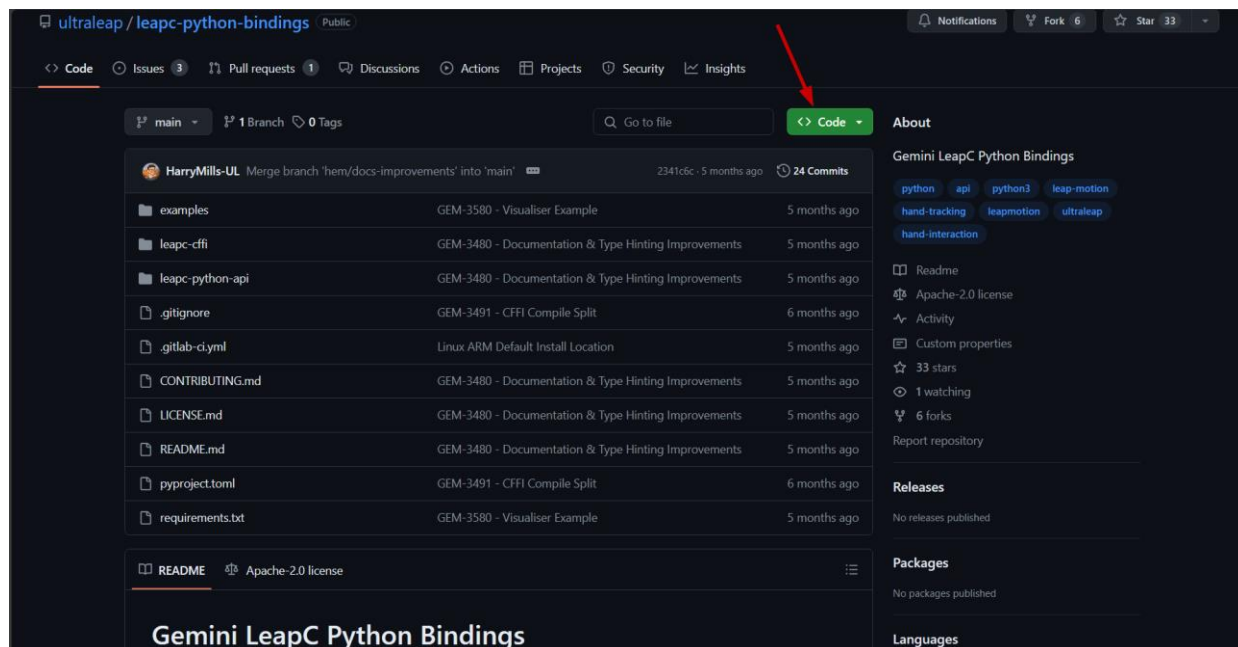


Figura 6 - Schermata Github della libreria

Leggete attentamente il README.md della libreria <https://github.com/ultraleap/leapc-python-bindings/blob/main/README.md> per tutti i dettagli tecnici.

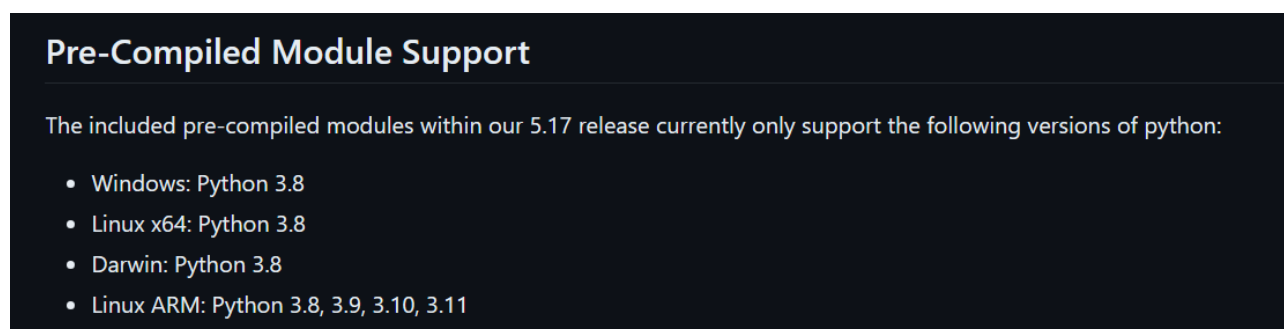


Figura 7 - Schermata dal Readme della libreria

In questa guida, useremo il “Pre-compiled module”, quindi installeremo la versione più recente di Python 3.8 dal sito ufficiale <https://www.python.org/downloads/release/python-3810/>.

Non scaricate la versione di Python dal Microsoft Store per problemi di incompatibilità.

Files					
Version	Operating System	Description	MD5 Sum	File Size	PGP
Gzipped source tarball	Source release		83d71c304acab6c678e86e239b42fa7e	23.6 MB	SIG
XZ compressed source tarball	Source release		d9eee4b20155553830a2025e4dcaa7b3	17.6 MB	SIG
macOS 64-bit Intel installer	macOS	for macOS 10.9 and later	690ddb1be403a7efb202e93f3a994a49	28.5 MB	SIG
macOS 64-bit universal2 installer	macOS	experimental, for macOS 11 Big Sur and later; recommended on Apple Silicon	ae8a1ae082074b260381c058d0336d05	35.6 MB	SIG
Windows installer (64-bit)	Windows	Recommended	62cf1a12a5276b0259e8761d4cf4fe42	27.0 MB	SIG
Windows installer (32-bit)	Windows		b355cfc84b681ace8908ae50908e8761	25.9 MB	SIG
Windows help file	Windows		a06af1ff933a13f6901a75e59247cf95	8.2 MB	SIG
Windows embeddable package (64-bit)	Windows		3acb1d7d9bde5a79f840167b166bb633	7.8 MB	SIG
Windows embeddable package (32-bit)	Windows		659adf421e90fba0f56a9631f79e70fb	7.0 MB	SIG

Figura 8 - Schermata di Python 3.8.10 dal sito ufficiale

Installate Python e cliccare su “Disable path length limit” (il software finale potrebbe girare in una directory molto lunga).

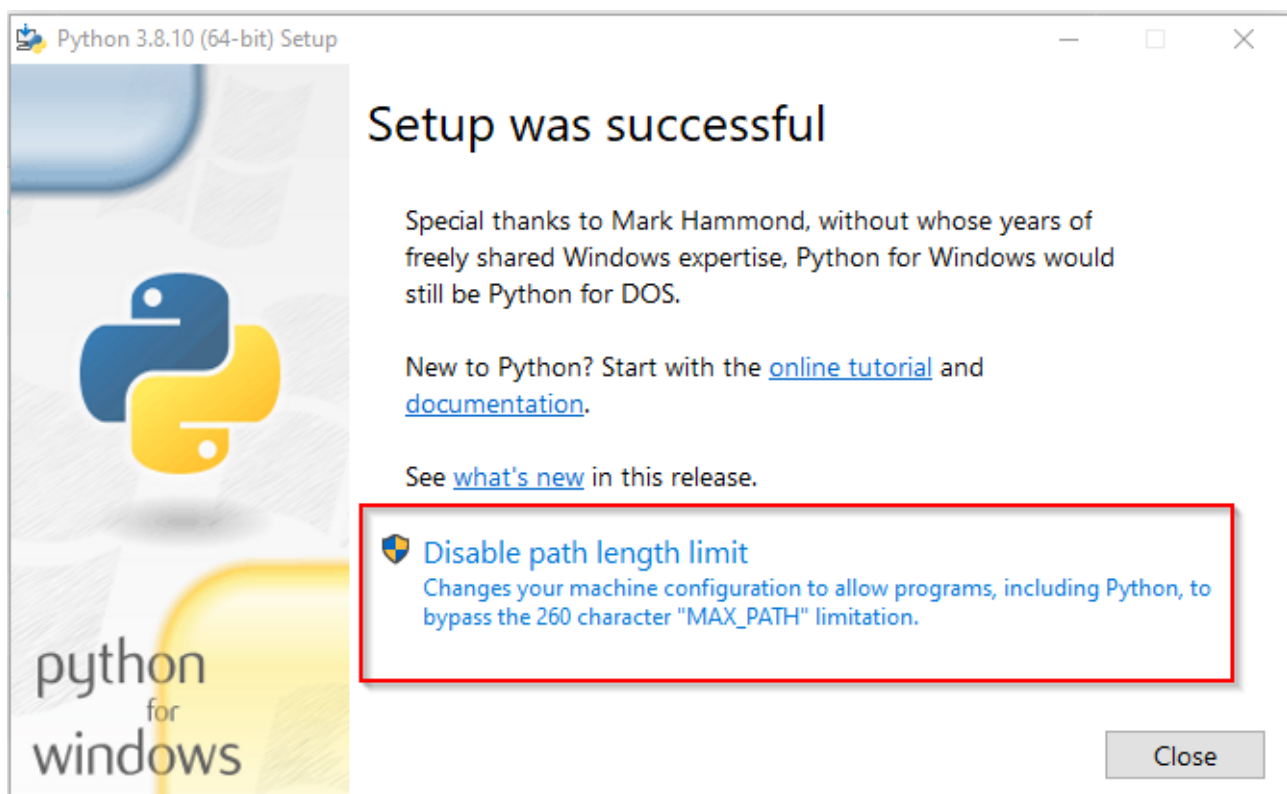


Figura 9 - Python installazione step finale

Riavviate il computer, verificate da PowerShell che la versione di Python sia quella corretta con il comando “python --version”.

```
PS C:\Users\Leap Motion> python --version
Python 3.8.10
```

Figura 10 - Versione Python da PowerShell

La libreria potrebbe avere dei problemi se ci sono diverse versioni di Python nella stessa macchina: consiglio caldamente di far girare tutto il software in una virtual machine, in un utente diverso da quello abituale o fisicamente in un computer differente con solo la versione di Python indicato nel README.

Andate sulla directory della libreria scaricata da Github con PowerShell, usando il comando “cd” ed eseguire le istruzioni di installazione contenute nel README <https://github.com/ultraleap/leapc-python-bindings/blob/main/README.md>

Installation:

This module makes use of a compiled module called `leapc_cffi`. We include some pre-compiled python objects with our Gemini installation from 5.17 onwards. Supported versions can be found [here](#). If you have the matching python version and have installed Gemini into the default location you can follow the steps below:

```
# Create and activate a virtual environment
pip install -r requirements.txt
pip install -e leapc-python-api
python examples/tracking_event_example.py
```

Figura 11 - Comandi per installare la libreria

Provate il funzionamento tra il Leap Motion e il pc digitando il comando “python examples/tracking_event_example.py” o uno tra i tanti file di esempio contenuti nella cartella.

```
PS C:\Users\Leap Motion\Downloads\leapc-python-bindings-main\leapc-python-bindings-main> python .\examples\tracking_event_example.py
Connected
Found device LP13431895929
Frame 3911 with 0 hands.
Frame 3912 with 0 hands.
Frame 3913 with 0 hands.
Frame 3914 with 1 hands.
Hand id 8 is a left hand with position (-7.710934638977051, 275.6236572265625, 87.25321197509766).
Frame 3915 with 1 hands.
Hand id 8 is a left hand with position (-7.52630615234375, 277.4698181152344, 88.71944427490234).
Frame 3916 with 1 hands.
Hand id 8 is a left hand with position (-7.491710662841797, 278.5382080078125, 89.4766616821289).
Frame 3917 with 1 hands.
Hand id 8 is a left hand with position (-7.589117050170898, 279.24493408203125, 89.71138000488281).
Frame 3918 with 1 hands.
Hand id 8 is a left hand with position (-7.877882480621338, 278.4035949707031, 89.33525085449219).
Frame 3919 with 1 hands.
Hand id 8 is a left hand with position (-8.139447212219238, 277.7308044433594, 89.04198455810547).
Frame 3920 with 1 hands.
Hand id 8 is a left hand with position (-8.39783763885498, 277.21783447265625, 88.82675170898438).
Frame 3921 with 1 hands.
Hand id 8 is a left hand with position (-8.666625022888184, 276.7685546875, 88.65727996826172).
Frame 3922 with 1 hands.
Hand id 8 is a left hand with position (-9.762511253356934, 276.43994140625, 88.40921020507812).
Frame 3923 with 1 hands.
Hand id 8 is a left hand with position (-10.526000022888184, 276.1640319824219, 88.17623901367188).
Frame 3924 with 1 hands.
Hand id 8 is a left hand with position (-10.96016788482666, 275.8433532714844, 87.99351501464844).
Frame 3925 with 1 hands.
```

Figura 12- Schermata con file di esempio funzionante

Fine della parte dalla documentazione ufficiale del software Ultraleap.

Programma per salvare i dati delle mani

Preparativi – da fare solo una volta

Scaricate la libreria pillow con il comando da PowerShell “pip install pillow”

```
PS C:\Users\Leap Motion> pip install pillow
Collecting pillow
  Using cached pillow-10.3.0-cp38-cp38-win_amd64.whl (2.5 MB)
Installing collected packages: pillow
Successfully installed pillow-10.3.0
```

Figura 13 - Installare module aggiuntivo

Andate sulla cartella della libreria iniziale della libreria “leapc-python-bindings-main” e cliccate sulla cartella “leapc-python-api”.

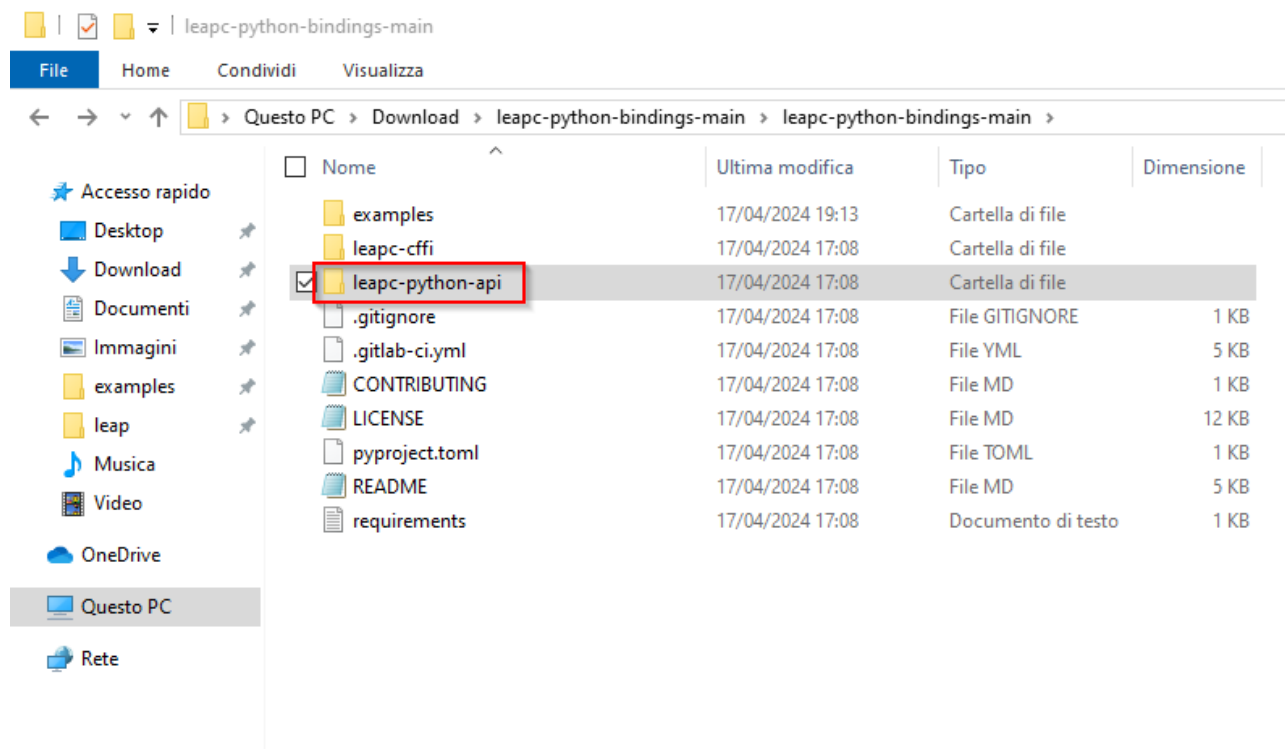


Figura 14 - Directory in cui è contenuto il file da modificare

Continuate nelle sottocartelle seguendo il percorso /src/leap

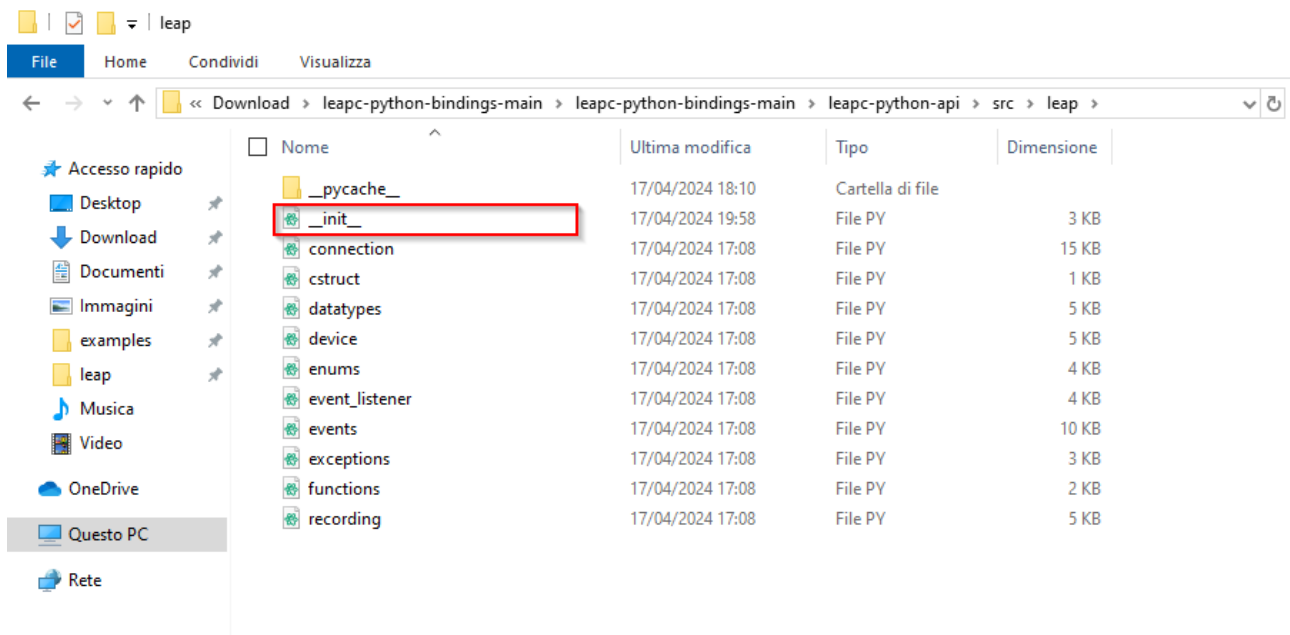


Figura 15 - Cartella e file da modificare

Aprire il file con un qualsiasi editor di testo (in questo caso con Atom, ma potete usare Visual Studio Code o semplicemente il Blocco note) e aggiungete alla riga 95 “,PolicyFlag”

```
__init__.py
61 cffi_location = _OS_DEFAULT_CFFI_INSTALL_LOCATION[get_system()]
62 if _OVERRIDE_LEAPSDK_LOCATION is not None:
63     cffi_location = _OVERRIDE_LEAPSDK_LOCATION
64
65 cffi_path = os.path.join(cffi_location, "leapc_cffi")
66 if os.path.isdir(cffi_path):
67     ret = check_required_files(cffi_path)
68
69 # TODO: If we can't find leapc_cffi, we could try building it
70
71 sys.path.append(cffi_location)
72
73 try:
74     from leapc_cffi import ffi, libleapc
75 except ImportError as import_error:
76     if not ret:
77         error_msg = f"Missing required files within {cffi_location}."
78     else:
79         error_msg = f"Unknown error, please consult readme for help. Attempting to find leapc_cffi within {cffi_location}"
80     raise ImportError(
81         f"Cannot import leapc_cffi: {error_msg}. Caught ImportError: {import_error}"
82     )
83 else:
84     error_msg = f"Error: Unable to find leapc_cffi dir within directory {cffi_location}"
85     raise Exception(error_msg)
86
87 from .functions import (
88     get_now,
89     get_server_status,
90     get_frame_size,
91     interpolate_frame,
92     get_extrinsic_matrix,
93 )
94 from .connection import Connection
95 from .enums import EventType, TrackingMode, HandType, PolicyFlag
96 from .event_listener import Listener
97 from .exceptions import LeapError
98 from .recording import Recording, Recorder
```

Figura 16 - Aggiunta al file __init__.py

Ritornate nella directory con i file di esempio e scaricate la cartella “Acquisizione dataset finale” https://github.com/ciccio25/leap_motion_to_csv/tree/main/examples/ ed estratela nella directory della libreria negli esempi.

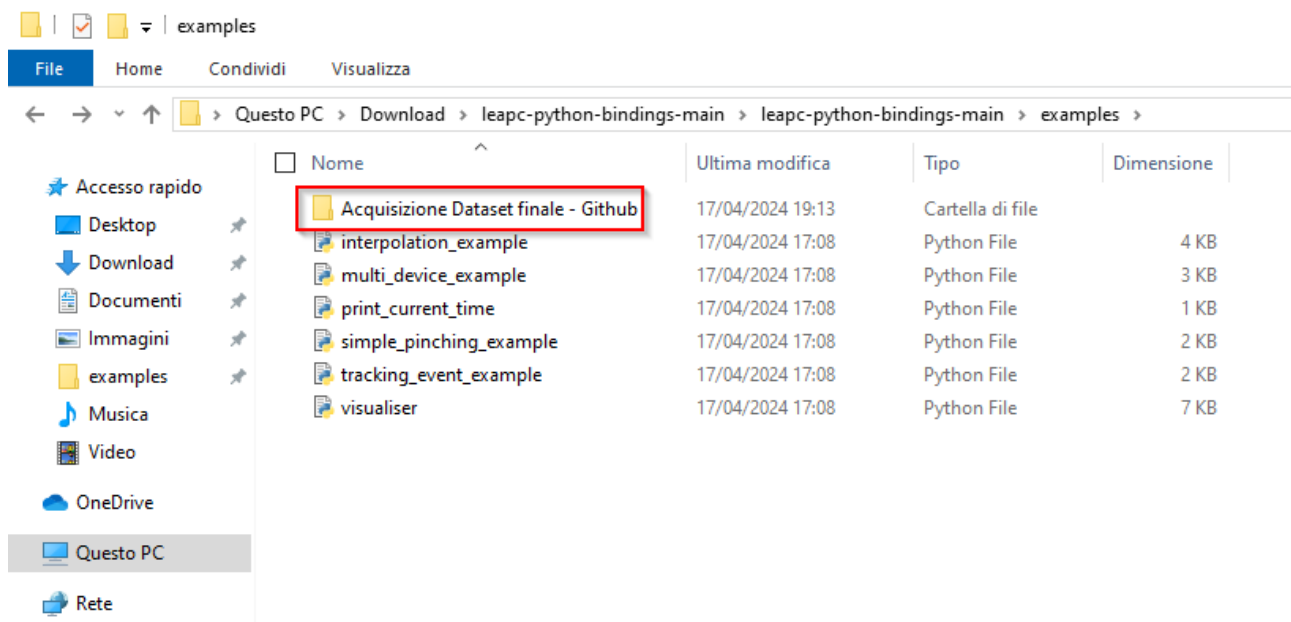


Figura 17 - File che devono essere presenti nella cartella

Come eseguire il programma per salvare le informazioni delle mani – da ripetere per ogni candidato

Cliccate sulla cartella finché non troviamo solo la cartella “Vuoto – da copiare per ogni persona”. Come è scritto nel nome della cartella, copiate e incollare la suddetta cartella e rinominatela nel nominativo del candidato che andrà a svolgere la prova

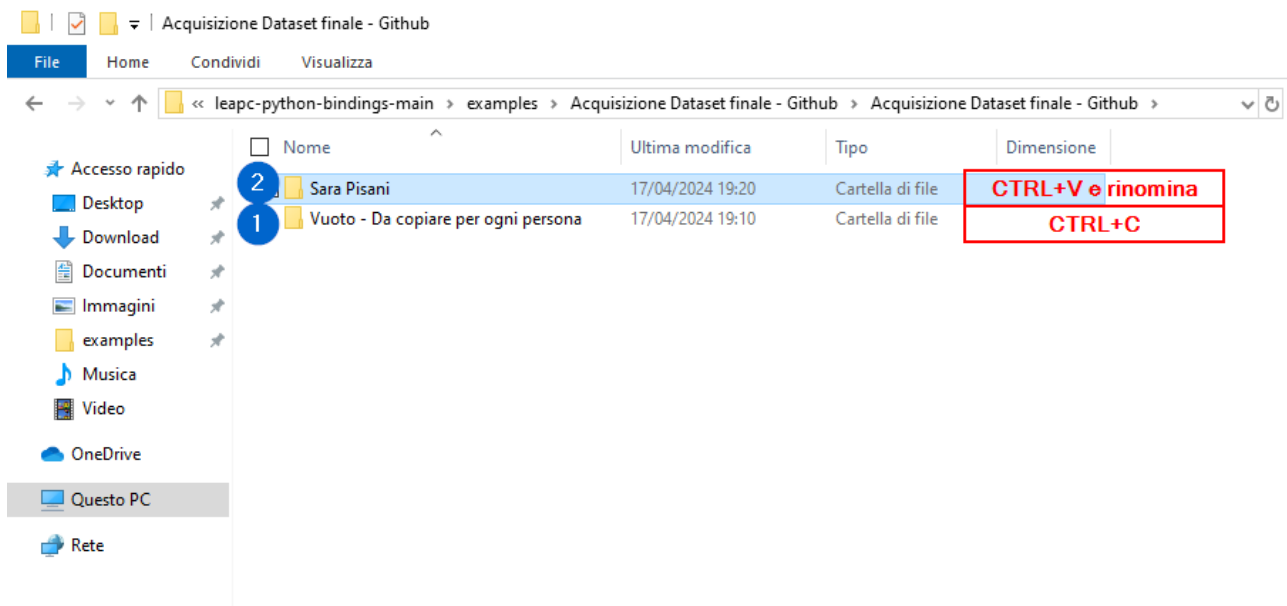


Figura 18 - File che dovranno essere presenti

Ora clicchiamo sulla cartella del candidato e ci troveremo il programma “acquisizione_dataset_finale.py”.

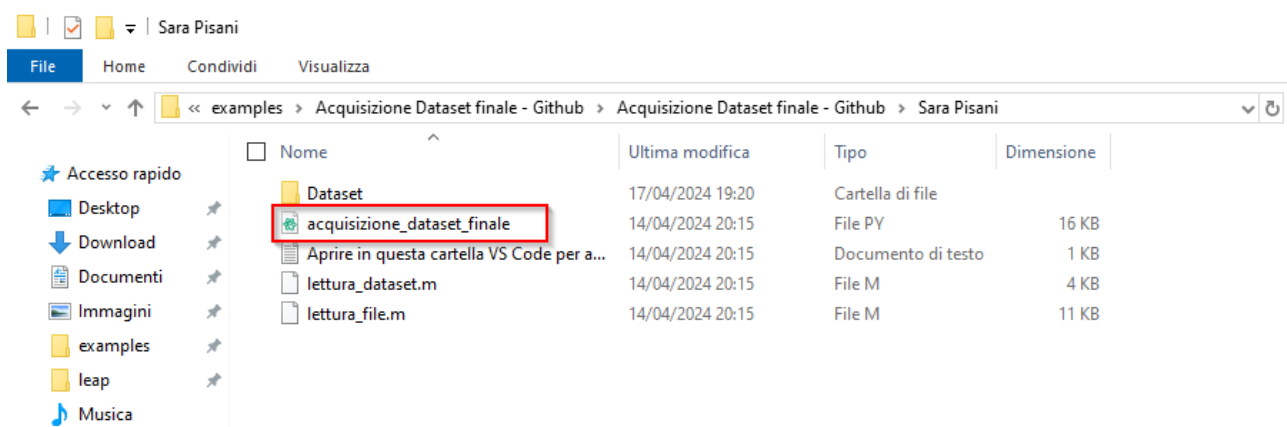


Figura 19 - File da eseguire

Ritornate su PowerShell in quella stessa directory e avviate il programma “acquisizione_dataset_finale.py”

Riconoscimento dei gesti offline

Obiettivo dell'esperimento

Per riprodurre lo stesso esperimento del paper <https://ieeexplore.ieee.org/document/7939924> ho svolto un piccolo dataset con diversi candidati usando solamente la mano destra.

La cattura è stata svolta con Python, mentre l'elaborazione dei dati usando Matlab R2024a. Il riconoscimento dei gesti è offline perché non è in real-time; quindi, è stata svolta in un momento successivo all'acquisizione.

Dalla cattura all'elaborazione

In una singola cattura sono state acquisite le coordinate delle punte delle dita (tip positions) e del palmo della mano (palm position) in 120 frame.

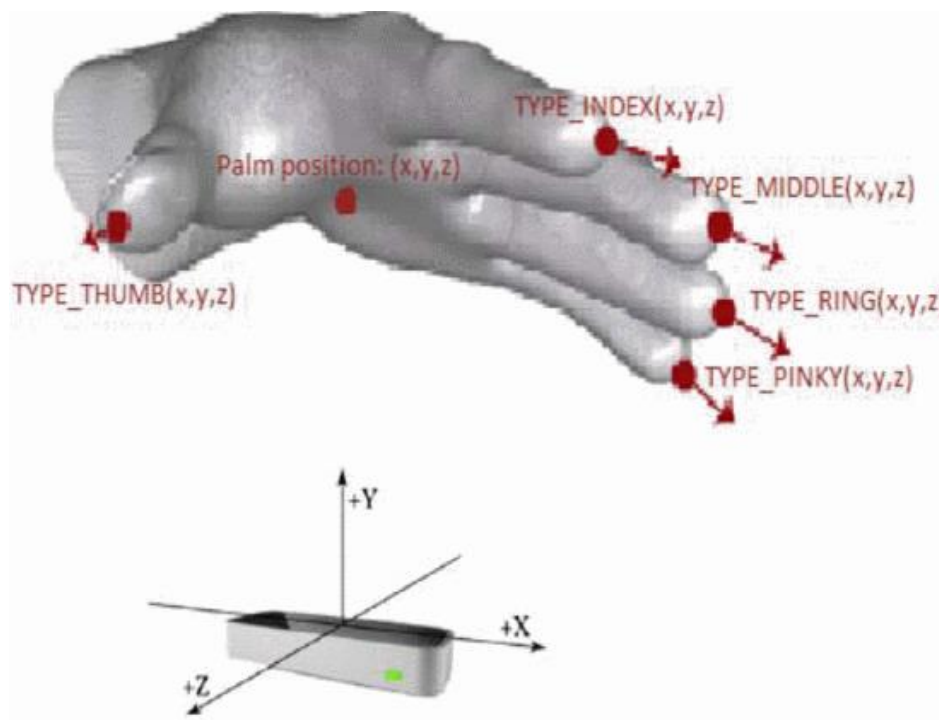


Figura 22 - Coordinate acquisite

Dalle coordinate di ogni oggetto (singolo dito e palmo della mano), viene rielaborato, la media aritmetica (arithmetic mean), la deviazione standard (standard deviation), la covarianza (covariance), il valore quadratico medio (Root Mean Square).



Figura 23 - Dati rielaborati

Questi dati sono stati rielaborati usando la funzione `lettura_dataset.m` (che dipende da `lettura_file.m`).

Dall'esecuzione di questo programma, viene elaborato il file `dataset_finale.csv`, in cui sono presenti le rielaborazioni delle coordinate per ogni candidato.

Con il programma `lettura_dataset_persone.m` vengono raggruppati in un unico csv tutti i `dataset_finale.csv` di ogni candidato in un unico file `datase_persone_finale.csv`.

Rispetto al paper, personalmente mi sono soffermato solamente sulle lettere in cui la mano rimane fissa: da `datase_persone_finale.csv` con il programma `dataset_SOLO_LETTERE_FISSE.m` viene estratto il file `dataset_lettere_fisse_casuale.csv` (Casuale perché la rete dovrà essere allenata e testata con la disposizione delle lettere in modo casuale)

Classification Learner

Usando l'app di Matlab Classification Learner

<https://it.mathworks.com/help/stats/classificationlearner-app.html> ho importato dataset [lettere_fisse_casuale.csv](#) ho scelto la rete NN.

Allenando tutte le reti ed i classificatori disponibili su Matlab, ho scelto una Wide Neural Network con accuratezza di validazione del 92,9% e del test di 87,2%

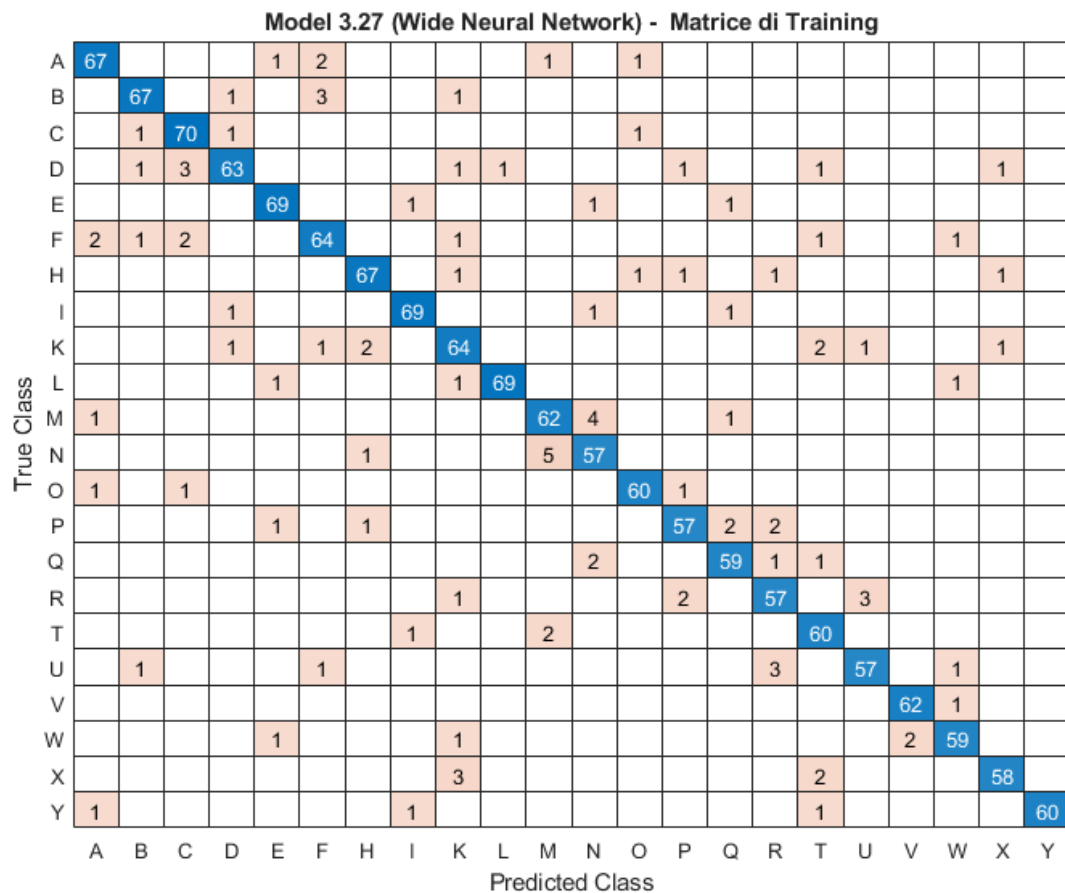


Figura 24 – Matrice di training della rete NN scelta

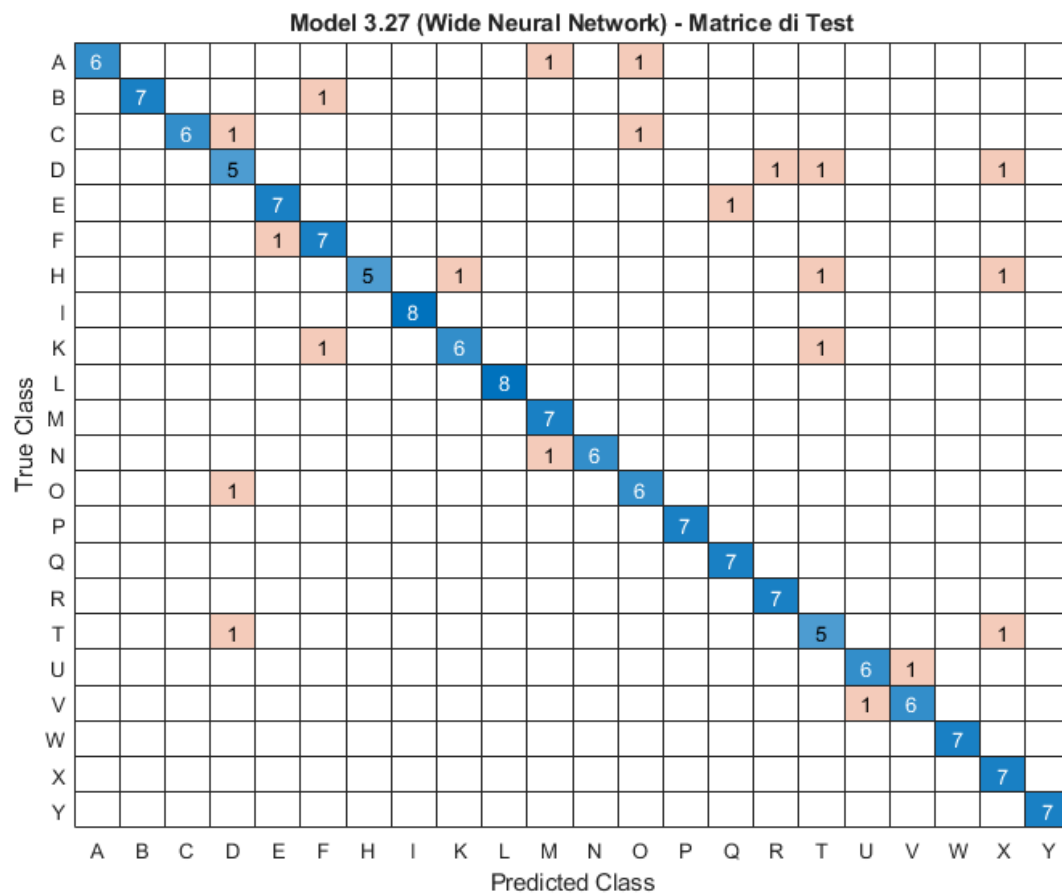


Figura 25 – Matrice di test della rete NN scelta

La sessione del Classification Learner è salvata nel dataset con il nome di ClassificationLearnerSession.mat

Conclusioni

In questo esperimento ho riproposto la stessa modalità di acquisizione e di rielaborazione del paper <https://ieeexplore.ieee.org/document/7939924> , impiegando la versione più recente del software di Ultraleap per l'acquisizione dei dati, utilizzando una rete neurale al posto del classificatore SVM grazie alla sua maggior accuratezza per il riconoscimento dei gesti.

L'esperimento ha impiegato molto tempo e molte risorse a causa della pochissima documentazione online del Leap Motion per la versione più aggiornata di quest'ultimo (a metà febbraio Orion è stato dismesso da Ultraleap) e per la poca trasparenza dei paper letti disponibili nell'IEEE, spiegando solo i concetti teorici senza un repository del codice sorgente.

Scrivere il codice è stato un insieme tra vecchia documentazione, quest'ultima disponibile solo su <https://web.archive.org/web/20170720234439/https://developer.leapmotion.com/documentation/python/index.html> , una playlist su YouTube <https://www.youtube.com/watch?v=T9k7rdY625M&list=PLgTGpdiW0iTELuljclIdTkA5SjHa5tudP> , e nuova documentazione, di cui la guida ufficiale in C <https://docs.ultraleap.com/api-reference/tracking-api/index.html> e i file di esempio di <https://github.com/ultraleap/leapc-python-bindings>

Nonostante i problemi citati, è stato molto piacevole imparare nuovi concetti che potrò impiegare nei miei futuri studi (reti neurali, lettura di paper accademici, Matlab) e poter scrivere, dopo tanto tempo, in Python.

Oltre alla parte tecnica, ringrazio con affetto amici e parenti che mi hanno aiutato alla realizzazione del dataset https://univpm-my.sharepoint.com/:f:/g/personal/s1098092_studenti_univpm_it/Et9JYClKM_pPjQJUaL3Xn3wByGynE7Mfd4wiH0bNRL4S6g?e=cDKVzb , in particolare Francesco con il quale potrò iniziare a parlare finalmente in LIS.