

Installatie Rapport: NGT-spellen-demo

Dit document beschrijft de stappen en probleemoplossingen die nodig waren om de applicatie (`main_app.py`) succesvol te installeren en uit te voeren in een lokale Windows-omgeving.

Systeemomgeving

- **Besturingssysteem:** Windows
- **Python versie:** 3.11.8

Uitgevoerde stappen

1. **Repository Klonen** De repository is succesvol gekloond vanaf GitHub:

```
git clone https://github.com/NOLAI/NGT-spellen-demo .
```

2. **Virtuele Omgeving Aanmaken** Er is een nieuwe virtuele omgeving (`venv`) aangemaakt en geactiveerd:

```
python -m venv venv
.\venv\Scripts\Activate.ps1
```

3. **Installatie van Dependencies (Probleemoplossing)** Tijdens het installeren van de packages via `pip install -r envs\NOLAI_gesture_live.txt` traden er verschillende compatibiliteitsproblemen op, voornamelijk door de gebruikte Python versie (3.11). De volgende aanpassingen moesten worden gedaan in de requirements:

- **Verwijderen van `backports.strenum==1.3.1`** : Dit pakket is gelimiteerd tot Python versies ouder dan 3.11 (`Requires-Python >=3.8.6,<3.11`). Vanaf Python 3.11 zit `StrEnum` standaard in de `enum` library, waardoor deze backport overbodig is en foutmeldingen veroorzaakt bij installatie. Deze is uit de requirements verwijderd.
- **Verwijderen van `tensorflow-io-gcs-filesystem==0.37.1`** : Voor deze specifieke versie kon geen geschikte distributie voor de huidige architectuur/Python-versie gevonden worden via pip. Aangezien we de app alleen live willen draaien en geen gebruik maken van Google Cloud Storage voor de modellen in deze fase, is deze dependency verwijderd om de rest van de installatie succesvol te laten verlopen.
- **Oplossen van `typing_extensions conflict`**: In de requirements stond `typing_extensions==4.5.0` vastgepind. Echter had een andere dependency (`optree==0.18.0`) versie `>=4.6.0` nodig. De harde eis van `==4.5.0` is verwijderd uit `NOLAI_gesture_live.txt` zodat `pip` automatisch de correcte (nieuwere) versie (`4.15.0`) kon selecteren om aan alle eisen te voldoen.
- **OpenCV OSError (Permissieconflict)**: Tijdens het uitpakken van de OpenCV binaries trad er een `[WinError 32]` (bestandsvergrendeling/permissie) op (bij het uitpakken van een Haar Cascade XML bestand). Dit gebeurt soms bij de installatie van `opencv-python` en `opencv-contrib-python` op Windows als er bestanden gelijktijdig door achtergrondprocessen gescand of geblokkeerd worden. *Oplossing*: Het geforceerd deïnstalleren (`pip uninstall -y opencv-python opencv-contrib-python`) en opnieuw uitvoeren van de installatie loste dit probleem op.

4. **Klaar voor gebruik** Na deze stappen kon de applicatie zonder verdere import errors gestart worden:

```
python main_app.py
```

Opmerking bij opstarten: Er verscheen een Scikit-Learn `InconsistentVersionWarning` (omdat een pre-trained LabelEncoder met versie 1.4.2 was opgeslagen, terwijl we nu 1.7.2 gebruiken), maar dit hindert de werking van de applicatie op het eerste gezicht niet.

Aanbevelingen voor de ontwikkelaars

- **Python-versie specificatie:** Geef duidelijk aan voor welke specifieke Python versies (bijv. 3.9 of 3.10) de environments getest zijn. De README noemt Python 3.10.19; als nieuwere versies worden gebruikt (zoals 3.11+) is het raadzaam de requirements af te zwakken (bijv. `>=` in plaats van `=`) of conditionele dependencies (environment markers) toe te voegen voor packages zoals `backports.strenum`.
- **Loskoppelen van strenge versies:** Voor een live/democase kan het handig zijn om niet alle dependencies op een exacte pinnen (gebruik van `=`), behalve de kerncomponenten (zoals `tensorflow` / `keras` of `mediapipe`). Dit voorkomt de conflicten die nu ontstonden bij dependencies van dependencies (zoals `typing_extensions`).

Zodra de virtuele omgeving actief is, draait de app direct!

Standalone Executable (Windows)

De applicatie is succesvol verpakt als een standalone executable met **PyInstaller** (`--onedir` mode). Dit betekent dat de app overgezet en gedraaid kan worden op andere Windows-computers zonder dat Python of de bijbehorende dependencies geïnstalleerd hoeven te worden.

Voor de verpakking is er een `resource_path` functie aan `main_app.py` toegevoegd om bestanden (zoals afbeeldingen en modellen) correct te lokaliseren, zowel in de broncode als in de verpakte `.exe`.

De volgende commando is gebruikt om de executable succesvol te bouwen inclusief de benodigde verborgen pakketten en datamappen:

```
pyinstaller --noconfirm --onedir --windowed --add-data "train;train" --add-data "images;images" --hidden-import mediapipe main_app.py
```

Locatie van de Executable

De gecompileerde map bevindt zich in: `d:\Temp\Python\ngt-spellen-demo\dist\main_app`

Daarin vind je het uitvoerbare bestand **main_app.exe**. Als je dit start, begint de NGT-Spellen direct, zonder een storend zwart command-prompt venster.

De app verspreiden

Om de applicatie aan iemand anders te geven:

1. Ga naar `d:\Temp\Python\ngt-spellen-demo\dist\`
2. Maak een ZIP-bestand van de hele map `main_app/` (inclusief alle inhoud!).
3. Degene die dit ontvangt kan de ZIP uitpakken en dubbelklikken op `main_app.exe`.