Protocol for the Online MASC experiment

Luca Verschure

University of Twente

NeuroTech Lab

Centre Hospitalier Universitaire Vaudois (CHUV), Lausanne, Switzerland

04.02.2025

Movie for the Assessment of Social Cognition (MASC) – Web browser adaptation

1. Overview

This protocol outlines the setup, execution, and modification of the Movie for the Assessment of Social Cognition (MASC) task adapted for online administration via Pavlovia. The goal is to facilitate remote social cognition assessments while ensuring validity and usability.

The experiment was developed using PsychoPy v2024.2.4 (and required Python 3.8+) and uploaded to Pavlovia for browser-based accessibility. PsychoPy, a Python-based software, allows for certain JavaScript auto-translation, enabling web deployment. The experiment was designed using a combination of the PsychoPy Builder UI tool and custom Python code, which was automatically converted to JavaScript for online execution. Since Pavlovia only supports JavaScript, all experiment logic had to be compatible with both Python and JavaScript to ensure smooth execution across different browsers.

The purpose of this experiment is to assess social cognition by evaluating participants' ability to interpret social interactions. The Movie for the Assessment of Social Cognition (MASC) (Dziobek, 2006) presents short film clips of social interactions, followed by multiple-choice questions that assess mentalizing and Theory of Mind (ToM) abilities. This adapted experiment aims to provide a scalable, remote version of the MASC test, allowing researchers to expand data collection beyond traditional lab settings and include a more diverse participant pool.

The experiment generates structured CSV output files, capturing the following data for each participant:

- Response accuracy (correct/incorrect answers)
- Reaction times (time taken to select an answer)
- Confidence ratings (self-reported certainty in responses)

- Error categorizations (overmentalization, undermentalization, absence of Theory of Mind, control)
- Trial-by-trial logs (timestamps and trial progression)

All participant responses are automatically stored in the Pavlovia server's " /data/" directory, allowing researchers to download and analyze the results post-experiment. However, in its current piloting mode, the experiment also saves a local CSV file on the participant's device. This ensures that data is accessible even if server synchronization issues occur and allows for immediate inspection of trial-level responses before full deployment.

One major limitation of the current experiment is its incompatibility with mobile devices. Since certain actions require arrow key inputs, which are not readily available on touchscreen devices, the experiment cannot be reliably conducted on smartphones or tablets. Future improvements could include implementing an on-screen button interface or enabling touch gestures to enhance accessibility for mobile users.

2. Files & Project Structure

Main Experiment Files

- MASC.psyexp – PsychoPy Builder project file

Python Scripts

- MASC.py – Main Python script
- MASC_lastrun.py – Auto-generated last-run script

JavaScript Conversion (for Pavlovia)

- MASC.js – Automatically generated for Pavlovia

- MASC-legacy-browsers.js – Alternative script for older browsers

Stimuli Files

- Individual video files (01.mp4, 02.mp4, ..., 46.mp4)
- Some clips are combined (e.g., 06-07.mp4, 22-23.mp4, 36-37.mp4)

Conditions & Questions Files

- MASC_Characters.xlsx – Character-related data
- MASC_Cleaned_SingleRowPerQuestion_Corrected (1).xlsx – Cleaned dataset with corrected questions
- MASC_Instructions.xlsx – Instruction content

Participant Data Storage

- data/ – Folder where Pavlovia automatically saves participant responses in CSV format

Other Assets

- Character images:
    - Anna.png, Ben.png, Marie.png, Michael.png
- index.html – Main webpage for Pavlovia deployment
- readme.md – Documentation file

3. How to Run the Experiment Locally

Install PsychoPy (if not already installed)

Download from the PsychoPy website (this tutorial gives extra information https://www.youtube.com/watch?v=9KpWOqsoa4k)

Open the PsychoPy Project

1. Load MASC.psyexp in PsychoPy Builder
2. Ensure all file paths are correctly linked:
    - **Verify video paths** inside the folder
    - **Check spreadsheet links** for MASC_Cleaned_SingleRowPerQuestion_Corrected (1).xlsx

Run the Experiment Locally

1. Click **"Run Experiment"** in PsychoPy
2. Verify:
    - **Video playback** works correctly
    - **Responses** are recorded properly
    - Bear in mind that the experiment will visually be slightly different locally and online due to inherent differences between Python and JavaScript

Check Output Files

- Data should be saved in the /data/ folder with filenames following the format: [participantID]_MASC_[YYYY-MM-DD_HHhMM.SS.mmm].csv

4. How to Deploy & Run the Experiment on Pavlovia

Create a Pavlovia Account

- Sign up at Pavlovia.org

Sync the Experiment

1. In PsychoPy, go to "File" → "Export HTML"
2. Click "Sync with Pavlovia"

Test the Experiment Online

1. Navigate to your Pavlovia project page
2. Click "Pilot" to verify everything before data collection

Launch Data Collection

- Click "Run the Study"
- The experiment will be publicly accessible via a URL

5. Clone repository

To work with the experiment files locally, you need to clone the repository from Pavlovia. You can do this using either SSH or HTTPS:

Clone with SSH (recommended for frequent contributors with SSH keys set up) (tutorial on how to get one can be found here https://www.google.com/search?q=pavlovia+ssh+key&oq=pavlovia+ssh+key&gs_lcrp=EgZjaHJvbWUyBggAEEUYOTIGCAEQLhhA0gEIMjUxMmowajGoAgCwAgA&sourceid=chrome&ie=UTF-8#fpstate=ive&vld=cid:4fd7a480,vid:8AcNwLzBwPA,st:0):

git clone git@gitlab.pavlovia.org:LucaVerschure/masc.git

Clone with HTTPS (simpler, requires login for pushing changes):

git clone https://gitlab.pavlovia.org/LucaVerschure/masc.git

Once cloned, navigate into the project folder using:

“cd masc”

You can then open the project in PsychoPy or make modifications as needed.