

# Projet Pouce

Contrôle gestuel en temps réel avec MediaPipe

05.02.2026

# Contents



1	Rappel du contexte .....	4
1.a	Pourquoi “Pouce” ? .....	5
2	Données utilisées .....	6
2.a	MediaPipe Hand Landmarker .....	7
3	Approches de détection .....	8
3.a	Première approche : Coordonnées Y .....	9
3.b	Deuxième approche : Distance au poignet .....	10
3.c	Troisième approche : Angles et Paume .....	11
4	Comparaison des approches .....	12
4.a	Tableau récapitulatif .....	13
5	Stack technique .....	14
5.a	Technologies utilisées .....	15
6	Défis et Solutions .....	16

## Contents (ii)



6.a Problèmes rencontrés .....	17
7 Démo fonctionnelle .....	18
7.a Interfaces .....	19
7.b Modes disponibles .....	20

# 1 Rappel du contexte

# Pourquoi “Pouce” ?



L'objectif est de créer une interface homme-machine (IHM) naturelle et sans contact.

- **Interaction intuitive** : Utilisation des mains pour interagir avec le système.
- **Accessibilité** : Une alternative aux périphériques classiques (souris/clavier).
- **Cas d'usage** : Contrôle multimédia, dessin virtuel, jeux, environnements stériles (médical).

## 2 Données utilisées

# MediaPipe Hand Landmarker



- **Source** : Flux vidéo de la webcam (640x480).
- **Modèle** : MediaPipe (Google) pré-entraîné.
- **Landmarks** : 21 points clés en 3D par main identifiée.
- **Fréquence** : Traitement asynchrone pour garantir la fluidité (30+ FPS).

## **3 Approches de détection**



## Première approche : Coordonnées Y



Comparer simplement la hauteur du bout du doigt (TIP) par rapport à l'articulation précédente (PIP).

- **Résultats** : Fonctionne bien si la main est parfaitement verticale.
- **Échec** : Dès que la main tourne (horizontale ou inclinée), la détection s'inverse ou échoue totalement. Ne gère pas le pouce (mouvement latéral).

## Deuxième approche : Distance au poignet

Calculer la distance Euclidienne entre le poignet (WRIST) et le bout du doigt.

- **Résultats** : Insensible à la rotation de la main dans le plan de l'image.
- **Échec** : Le pouce est problématique car son extension ne l'éloigne pas forcément du poignet de manière linéaire par rapport aux autres doigts. Problème de perspective (main face caméra).

## Troisième approche : Angles et Paume ●●●●●●●●●●○○○○○○○○○○

Approche actuelle combinant plusieurs critères :

1. **Angles** : Calcul de l'angle aux articulations (MCP-PIP-TIP). Un doigt est tendu si l'angle est proche de  $180^\circ$ .
2. **Centre de la paume** : Pour le pouce, on compare sa distance par rapport au centre de la paume (moyenne poignet + index\_mcp + pinky\_mcp).
3. **Marge dynamique** : Seuil basé sur la largeur de la paume pour s'adapter à la distance de la caméra.

## 4 Comparaison des approches

# Tableau récapitulatif



Approche	Avantages	Limites
Coordonnées Y	Simple, rapide	Sensible à l'orientation
Distance Poignet	Stable en rotation	Échec sur le pouce / perspective
Angles + Paume	Robuste, multi-angle	Plus complexe mathématiquement

## 5 Stack technique

# Technologies utilisées



- **Langage** : Python 3.13 (géré par `uv`).
- **Vision** :
  - **MediaPipe** : Détection des landmarks.
  - **OpenCV** : Capture vidéo et rendu de l'interface.
- **Interface** :
  - **Streamlit** : Dashboard interactif web.
  - **Streamlit-webrtc** : Streaming vidéo temps réel dans le navigateur.
- **Calcul** : **NumPy** et **Math** pour la géométrie 3D.
- **Déploiement** : Application locale ou web.

## 6 Défis et Solutions



## Problèmes rencontrés



- **Détection du pouce** : Le pouce a une liberté de mouvement unique.
  - **Solution** : Utilisation du centre de la paume comme point de référence latéral.
- **Luminosité** : Le modèle peut perdre la main en contre-jour.
  - **Solution** : Normalisation des coordonnées par MediaPipe.
- **Inversion miroir** : La webcam inverse l'image.
  - **Solution** : Flip horizontal via OpenCV pour une interaction “miroir” naturelle.

## **7 Démo fonctionnelle**

# Interfaces



- **Version CLI** : Interface OpenCV classique pour une performance maximale.
- **Version Web** : Interface Streamlit moderne, accessible via navigateur, facilitant le changement de mode.

# Modes disponibles



1. **Energy Ball** : Visualisation du “pinch” entre pouce et index (capture d’écran automatique).
2. **Air Painter** : Dessin virtuel (Pincer pour dessiner, 5 doigts pour effacer).
3. **Finger Count** : Compteur de doigts (gère 2 mains simultanément).
4. **Rock Paper Scissors** : Jeu de Pierre-Feuille-Ciseaux contre l’ordinateur.
5. **Virtual Mouse** : Contrôle du curseur avec la main (pincer pour cliquer, 2 mains pour déplacer).
6. **Camera Zoom** : Zoom avec deux mains (écarter pour zoomer, pincer pour dézoomer).