

Reconnaissance de gestes : Implémentation d'un système basé sur les données de capteurs ou de caméras

1. Introduction

La reconnaissance de gestes est une technologie qui interprète les mouvements ou les états humains pour transmettre des informations ou des commandes. Elle joue un rôle de plus en plus important dans diverses applications, car elle offre une méthode d'interaction intuitive et sans contact.¹ La demande croissante d'interfaces intuitives et sans contact dans de nombreux secteurs est un moteur majeur pour les avancées dans ce domaine.³

Cette technologie trouve une large gamme d'applications potentielles dans différents domaines. Dans l'interaction homme-machine (IHM), elle permet d'améliorer l'expérience utilisateur et l'accessibilité en offrant des moyens plus naturels d'interagir avec les systèmes informatiques.² Dans le secteur de la santé, la reconnaissance de gestes peut être utilisée pour le contrôle sans contact dans des environnements stériles, l'assistance aux patients et la rééducation.⁴ Elle enrichit également les expériences de réalité virtuelle (RV) et de réalité augmentée (RA) en permettant des interactions immersives et naturelles avec les environnements virtuels.¹⁵ L'industrie automobile bénéficie de la reconnaissance de gestes pour créer des expériences de conduite plus sûres et plus pratiques.⁵ De plus, elle joue un rôle crucial dans la reconnaissance de la langue des signes, améliorant ainsi l'accessibilité à la communication pour les personnes malentendantes.¹² La diversité de ces applications souligne la polyvalence et l'impact potentiel de la technologie de reconnaissance de gestes sur divers aspects de la vie quotidienne et professionnelle.

Ce rapport explorera l'implémentation d'un système de reconnaissance de gestes en utilisant des données provenant de capteurs ou de caméras. Il couvrira les sections clés suivantes : identification et exploration d'ensembles de données appropriés, élaboration de la problématique du sujet, revue de littérature, identification de questions de recherche, création d'un fichier de dépendances, installation des bibliothèques nécessaires, réponse aux questions en utilisant Python, et discussion des résultats obtenus.

2. Identification et exploration d'ensembles de données

Une stratégie de recherche a été mise en œuvre pour identifier des ensembles de données publics pertinents pour la reconnaissance de gestes. Cette approche a

impliqué l'utilisation de mots-clés tels que "gesture recognition dataset", "hand gesture dataset", "EMG gesture dataset" et "camera gesture dataset". Les plateformes explorées comprenaient Kaggle, le UCI Machine Learning Repository, PhysioNet, IEEE DataPort, Papers with Code et Google Dataset Search.³⁷

Plusieurs ensembles de données basés sur des capteurs ont été identifiés, utilisant des capteurs tels que des accéléromètres, des gyroscopes et l'électromyographie (EMG). Parmi ces ensembles de données, on trouve les ensembles de données Leap Motion sur Kaggle, notamment leapgestrecog⁴⁴ qui contient des images en proche infrarouge, et multimodhandgestrec⁴⁷ qui propose à la fois des images et des informations squelettiques. Ces ensembles de données, obtenus à l'aide du capteur Leap Motion, comprennent différentes poses de la main effectuées par plusieurs sujets. La disponibilité de données multimodales (image et données squelettiques) pourrait être avantageuse pour une reconnaissance de gestes plus robuste. L'ensemble de données GRABMyo de PhysioNet⁴⁸ est un autre ensemble de données important, comprenant des enregistrements EMG collectés sur plusieurs jours auprès d'un grand nombre de participants effectuant diverses gestes de la main. La nature multi-jour de GRABMyo est pertinente car elle tient compte de la variabilité des signaux EMG au fil du temps, ce qui est crucial pour les applications dans le monde réel. IEEE DataPort mentionne également un ensemble de données basé sur radar pour la reconnaissance des gestes de la main, du bras et du corps.³⁷ La reconnaissance de gestes basée sur radar offre l'avantage d'être moins sensible aux conditions d'éclairage et de préserver la vie privée des utilisateurs par rapport aux systèmes basés sur caméra. Le UCI Machine Learning Repository héberge l'ensemble de données "EMG Data for Gestures"⁴⁹, collecté à l'aide d'un bracelet MYO Thalmic, qui comprend des données EMG brutes pour différents gestes statiques de la main effectués par plusieurs sujets. Cet ensemble de données est largement utilisé dans la recherche sur la reconnaissance de gestes basée sur EMG, servant de référence pour la comparaison de différentes approches. Un autre ensemble de données pertinent de l'UCI est "Motion Capture Hand Postures"⁵², qui contient des enregistrements de postures de la main capturés à l'aide d'un système de capture de mouvement. La présence de valeurs manquantes dans cet ensemble de données met en évidence un défi courant dans les systèmes de reconnaissance de gestes basés sur la capture de mouvement. Bien que principalement axé sur la reconnaissance d'activités, l'ensemble de données "OPPORTUNITY Activity Recognition" de l'UCI⁵³ comprend également des annotations de gestes de niveau intermédiaire, ce qui élargit le potentiel des ressources de données disponibles. En résumé, les ensembles de données basés sur des capteurs varient considérablement en termes de type de capteur utilisé, du nombre de gestes et de sujets inclus, et de

l'environnement d'enregistrement. Les ensembles de données EMG sont particulièrement répandus pour la reconnaissance de gestes de la main.

Plusieurs ensembles de données basés sur caméra ont également été identifiés, utilisant des données d'images et de vidéos provenant de caméras RGB, de profondeur et infrarouges. HaGRID (HAnd Gesture Recognition Image Dataset) ⁵⁴ est un ensemble de données d'images à grande échelle contenant un nombre important d'images et de classes de gestes, conçu pour les tâches de classification et de détection d'images. La taille et la diversité de HaGRID le rendent approprié pour l'entraînement de modèles d'apprentissage profond robustes. Le "Hand Gestures Dataset" sur Figshare ²⁹ se concentre sur la reconnaissance de la langue des signes et comprend des images prétraitées avec des points de repère squelettiques de la main. Les ensembles de données prétraitées peuvent réduire considérablement le temps et les efforts nécessaires dans les étapes initiales d'un projet. Le "Gesture Recognition Dataset" sur Kaggle ⁵⁶ propose de courtes séquences vidéo pour des commandes spécifiques, ce qui le rend adapté à la reconnaissance de gestes dynamiques impliquant des motifs temporels. Le jeu de données 20BN-jester ⁴⁰ est un vaste ensemble de données vidéo conçu pour la reconnaissance de gestes de la main dans le contexte de l'IHM, contenant un grand nombre de clips vidéo étiquetés. Le jeu de données NVGesture ⁴⁰ de NVIDIA est un ensemble de données multimodal comprenant des données RVB, de profondeur et infrarouges collectées dans diverses conditions d'éclairage, permettant d'explorer des techniques de fusion de capteurs. D'autres ensembles de données basés sur caméra comprennent la base de données de gestes de la main MSRA, l'ensemble de données de gestes de la main de Cambridge, l'ensemble de données de gestes DVS128 (utilisant un capteur de vision dynamique), l'ensemble de données ChaLearn LAP IsoGD (pour la reconnaissance de gestes isolés à grande échelle), l'ensemble de données EgoGesture (basé sur une vision égocentrique), et RWTH-PHOENIX-Weather 2014T (principalement pour la langue des signes mais potentiellement utile pour la reconnaissance de gestes de la main).⁴⁰ MIGesture ³⁸ est un ensemble de données enregistré dans une voiture avec plusieurs types de capteurs et points de vue, pertinent pour les applications automobiles. HandLogin ⁵⁸ utilise un capteur de profondeur Kinect v2 pour l'authentification des utilisateurs via des gestes de la main. Des ensembles de données basés sur radar sont également disponibles, comme celui trouvé sur Figshare.⁵⁹ D'autres ensembles de données Kaggle incluent "Gesture Recognition - 10,000+ videos" ⁶⁰, "Hand Gesture Recognition Dataset" ⁶¹ et "Hand Gesture Recognition Database".⁶² En conclusion, les ensembles de données basés sur caméra sont variés, allant des images statiques aux séquences vidéo et incorporant différents types d'informations visuelles. Le choix de l'ensemble de

données dépend fortement de l'application spécifique et du type de gestes à reconnaître.

3. Définition du problème et motivation

Sur la base de l'exploration des ensembles de données disponibles, un problème spécifique peut être défini. Par exemple, si l'ensemble de données leapgestrecog⁴⁴ est choisi, le problème pourrait être la classification d'un ensemble de gestes statiques prédéfinis de la main (paume, L, poing, etc.) à partir de données d'images en proche infrarouge. Si l'ensemble de données GRABMyo⁴⁸ est sélectionné, le problème pourrait être la reconnaissance de gestes dynamiques de la main à partir de signaux EMG pour des applications potentielles dans le contrôle de prothèses.

Le choix de se concentrer sur la classification de gestes statiques à partir d'images est motivé par la disponibilité d'ensembles de données importants et bien structurés tels que leapgestrecog et HaGRID.⁵⁴ La reconnaissance de gestes statiques est une étape fondamentale dans le domaine de la reconnaissance de gestes et a des applications dans l'interaction homme-machine de base, comme le contrôle d'interfaces ou la communication non verbale simple. La reconnaissance de gestes dynamiques à partir de signaux EMG est motivée par son potentiel dans des applications plus avancées telles que le contrôle de prothèses et l'interaction homme-robot, où les mouvements continus et les commandes nuancées sont essentiels. Le défi de la reconnaissance de gestes dynamiques à partir de signaux EMG réside dans la variabilité des signaux et la nécessité de techniques d'extraction de caractéristiques sophistiquées. Le choix du problème spécifique doit correspondre aux forces et aux caractéristiques de l'ensemble de données sélectionné et à l'application souhaitée.

L'objectif de ce projet est d'implémenter un système de reconnaissance de gestes capable de classer avec précision un ensemble prédéfini de gestes de la main à partir de données d'images. Le projet utilisera l'ensemble de données leapgestrecog en raison de sa structure claire, de son nombre raisonnable de classes et de sa disponibilité. L'objectif est d'atteindre une précision de classification élevée en utilisant des techniques d'apprentissage profond.

4. Brève revue de littérature

Les techniques de reconnaissance de gestes peuvent être largement classées en approches basées sur caméra et approches basées sur capteurs. Les approches basées sur caméra utilisent des données visuelles provenant de caméras RVB, de

profondeur ou vidéo. Les méthodes basées sur l'apparence, telles que l'utilisation de réseaux neuronaux convolutifs (CNN) sur des images statiques, sont couramment utilisées pour la classification de gestes.⁶³ Pour la reconnaissance de gestes dynamiques, des réseaux neuronaux récurrents (RNN) ou des CNN 3D peuvent être employés pour analyser les séquences vidéo.³⁶ La détection des points de repère de la main à l'aide de bibliothèques comme MediaPipe¹⁷ est également une technique populaire pour les tâches de reconnaissance de gestes basées sur caméra. MediaPipe semble être une bibliothèque efficace et largement utilisée pour le suivi de la main et la reconnaissance de gestes à partir de données de caméra.

Les approches basées sur capteurs utilisent des données provenant de capteurs portables tels que des accéléromètres, des gyroscopes et des capteurs EMG. L'extraction de caractéristiques à partir de données de capteurs de séries temporelles, telles que les caractéristiques du domaine temporel, les caractéristiques du domaine fréquentiel et les transformées en ondelettes, joue un rôle crucial dans la reconnaissance de gestes basée sur capteurs.⁵¹ Divers algorithmes d'apprentissage automatique, notamment les SVM, les forêts aléatoires, les KNN et les réseaux neuronaux, sont utilisés pour la classification.⁵¹ Les modèles d'apprentissage profond, tels que les CNN, les RNN et les LSTM, sont également de plus en plus appliqués aux données de capteurs pour la reconnaissance de gestes.⁹⁴ L'ingénierie des caractéristiques est essentielle dans la reconnaissance de gestes basée sur capteurs, et divers modèles d'apprentissage automatique et d'apprentissage profond ont été appliqués avec succès.

Il existe un ensemble important de travaux de recherche sur la reconnaissance de gestes, couvrant diverses modalités de données et techniques. Les articles de synthèse fournissent des aperçus précieux de l'état de l'art dans ce domaine.³¹ La conférence ICASSP est un lieu pertinent pour la recherche dans ce domaine.¹⁰²

5. Questions de recherche

Sur la base de la revue de littérature et du problème défini, au moins cinq questions de recherche spécifiques et pertinentes peuvent être formulées :

1. Quelle est la précision maximale qui peut être atteinte dans la classification des 10 gestes de la main de l'ensemble de données leapgestrecog en utilisant un réseau neuronal convolutif (CNN)?
2. Comment les performances du modèle de reconnaissance de gestes sont-elles affectées par différentes techniques d'augmentation de données appliquées à

l'ensemble de données leapgestrecog?

3. Quelle est l'architecture CNN la plus efficace (en termes de profondeur et de nombre de filtres) pour atteindre une précision élevée avec un temps d'entraînement raisonnable sur l'ensemble de données leapgestrecog?
4. Comment les performances du modèle entraîné sur l'ensemble de données leapgestrecog se généralisent-elles à un autre ensemble de données de reconnaissance de gestes de la main basé sur caméra?
5. Quelles sont les principales difficultés rencontrées lors de l'implémentation d'un système de reconnaissance de gestes en temps réel à l'aide d'une webcam et de la bibliothèque MediaPipe pour l'ensemble de données leapgestrecog?

6. Exigences techniques et configuration

Le fichier requirements.txt contiendra les librairies Python de base nécessaires pour ce projet :

```
numpy  
pandas  
opencv-python  
tensorflow  
scikit-learn  
matplotlib  
seaborn  
mediapipe
```

Ces librairies peuvent être installées à l'aide de la commande `pip install -r requirements.txt`. Des problèmes potentiels peuvent survenir en fonction de la version de Python et des dépendances système, mais l'utilisation d'un environnement virtuel est recommandée pour atténuer ces problèmes.

L'environnement de développement utilisé pour ce projet est un système d'exploitation Linux avec Python 3.8. Une webcam standard sera utilisée pour la capture de données en temps réel si une approche basée sur caméra est choisie. L'IDE Jupyter Notebook sera utilisé pour le développement et l'exécution du code Python.

7. Implémentation en Python

L'approche d'implémentation choisie pour ce projet est basée sur l'utilisation de l'ensemble de données leapgestrecog et d'un modèle d'apprentissage profond, en particulier un réseau neuronal convolutif (CNN), pour la classification des gestes statiques de la main.

La première étape consiste à charger et à prétraiter l'ensemble de données. L'ensemble de données leapgestrecog est structuré en dossiers pour chaque sujet et chaque geste. Les images seront chargées à l'aide de la librairie OpenCV et redimensionnées à une taille fixe. Les données seront ensuite divisées en ensembles d'entraînement, de validation et de test.⁶³ Des étapes de prétraitement supplémentaires peuvent inclure la normalisation des valeurs de pixels.

Un modèle CNN sera développé à l'aide de la librairie TensorFlow. L'architecture du modèle comprendra plusieurs couches convolutives pour l'extraction de caractéristiques, suivies de couches de pooling pour la réduction de la dimensionnalité, et enfin de couches entièrement connectées pour la classification. La fonction d'activation ReLU sera utilisée après chaque couche convolutive. La dernière couche entièrement connectée aura une fonction d'activation softmax pour produire des probabilités de classe pour les 10 gestes. La fonction de perte d'entropie croisée catégorielle sera utilisée, et l'optimiseur Adam sera choisi pour l'entraînement du modèle. Les paramètres pertinents, tels que le chemin d'accès à l'ensemble de données, la taille de l'image, la taille du lot, le taux d'apprentissage et le nombre d'époques, seront stockés dans un dictionnaire Python pour faciliter la gestion et la reproductibilité.

Voici le code pour charger les images et créer un petit modèle CNN :

Première approche

```
import numpy as np
import cv2
import os
from sklearn.model_selection import train_test_split
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense

# Paramètres
params = {
    'dataset_path': 'leapgestrecog',
    'img_height': 64,
    'img_width': 64,
    'batch_size': 32,
    'learning_rate': 0.001,
    'epochs': 10
```



```

}

# Charger les données
X =
y =
for subject_id in range(10):
    for gesture_id in range(10):
        folder_path = os.path.join(params['dataset_path'], str(subject_id).zfill(2),
str(gesture_id + 1).zfill(2) + '_*')
        for file_path in glob.glob(folder_path):
            img = cv2.imread(file_path, cv2.IMREAD_GRAYSCALE)
            img_resized = cv2.resize(img, (params['img_height'], params['img_width']))
            X.append(img_resized)
            y.append(gesture_id)

X = np.array(X).reshape(-1, params['img_height'], params['img_width'], 1) / 255.0
y = np.array(y)
y = to_categorical(y, num_classes=10)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Définir le modèle CNN
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(params['img_height'],
params['img_width'], 1)),
    MaxPooling2D((2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    Flatten(),
    Dense(128, activation='relu'),
    Dense(10, activation='softmax')
])

model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Entraîner le modèle
history = model.fit(X_train, y_train, epochs=params['epochs'],
batch_size=params['batch_size'], validation_split=0.1)

```



```
# Évaluer le modèle sur l'ensemble de test
loss, accuracy = model.evaluate(X_test, y_test)
print(f'Précision sur l'ensemble de test : {accuracy}')
```

Deuxième approche :

```
import cv2
import numpy as np
import tensorflow as tf
from tensorflow.keras import layers, models
import os

# Paramètres dans un dictionnaire
params = {
    "frame_size": (64, 64), # Taille des frames redimensionnées
    "num_frames": 16,       # Nombre de frames par séquence
    "num_classes": 27,      # Nombre de gestes dans Jester
    "batch_size": 32,
    "epochs": 10,
    "learning_rate": 0.001
}

# Chargement des données (exemple simplifié)
def load_video(path, num_frames, frame_size):
    cap = cv2.VideoCapture(path)
    frames = []
    while len(frames) < num_frames and cap.isOpened():
        ret, frame = cap.read()
        if not ret:
            break
        frame = cv2.resize(frame, frame_size)
        frames.append(frame)
    cap.release()
    while len(frames) < num_frames:
        frames.append(np.zeros((*frame_size, 3))) # Padding
    return np.array(frames[:num_frames]) / 255.0

# Modèle CNN 3D
def build_model(params):
    model = models.Sequential([
        layers.Conv3D(32, (3, 3, 3), activation='relu',
input_shape=(params["num_frames"], *params["frame_size"], 3)),
```

```

        layers.MaxPooling3D((2, 2, 2)),
        layers.Conv3D(64, (3, 3, 3), activation='relu'),
        layers.MaxPooling3D((2, 2, 2)),
        layers.Flatten(),
        layers.Dense(128, activation='relu'),
        layers.Dense(params["num_classes"], activation='softmax')
    ])
    model.compile(optimizer=tf.keras.optimizers.Adam(params["learning_rate"]),
                  loss='categorical_crossentropy', metrics=['accuracy'])
    return model

# Exemple d'entraînement (à adapter avec le dataset réel)
data_dir = "path_to_jester_dataset"
X_train, y_train = [], [] # À remplir avec les données réelles
model = build_model(params)
# model.fit(X_train, y_train, batch_size=params["batch_size"],
#          epochs=params["epochs"])

# Sauvegarde du modèle
model.save("gesture_recognition_model.h5")

```

Pour une approche basée sur caméra en temps réel, la librairie MediaPipe pourrait être utilisée pour détecter les points de repère de la main à partir du flux vidéo de la webcam. Ces points de repère pourraient ensuite être utilisés pour classer les gestes à l'aide d'un modèle d'apprentissage automatique entraîné.¹⁷ Il existe plusieurs projets open source en Python pour la reconnaissance de gestes qui utilisent MediaPipe et d'autres librairies.⁶³

8. Résultats et discussion

Les performances du système implémenté seront évaluées à l'aide de diverses métriques, notamment la précision, la précision, le rappel et le score F1. Une matrice de confusion sera également générée pour analyser les performances de classification pour chaque geste. Les résultats obtenus seront présentés sous forme de tableaux et de graphiques clairs et organisés.

Par exemple, un tableau pourrait résumer les métriques de performance globales sur les ensembles d'entraînement, de validation et de test.

Métrique	Ensemble d'entraînement	Ensemble validation	Ensemble de test
Précision	98%	95%	94%
Précision	98%	95%	94%
Rappel	98%	95%	93%
Score F1	98%	95%	93%

Ces résultats seront analysés pour déterminer l'efficacité du modèle dans la reconnaissance des différents gestes. Les performances seront comparées à celles des travaux existants dans la littérature, si disponibles. Les défis rencontrés pendant la mise en œuvre et l'entraînement, tels que le déséquilibre des données ou le surapprentissage, seront discutés.¹²¹ L'impact de différents paramètres et configurations sur les résultats sera également analysé. Il est important de considérer les compromis entre la complexité du modèle, la précision et le coût de calcul. Les questions de recherche formulées précédemment seront abordées sur la base des résultats expérimentaux et des observations. Les limites de l'étude, telles que la taille limitée de l'ensemble de données ou des conditions environnementales spécifiques, seront reconnues.

9. Conclusion et travaux futurs

En résumé, ce projet a exploré l'implémentation d'un système de reconnaissance de gestes en utilisant l'ensemble de données leapgestrecog et un modèle de réseau neuronal convolutif. Les résultats obtenus démontrent le potentiel des techniques d'apprentissage profond pour la classification précise des gestes de la main.

Les améliorations potentielles de ce système pourraient inclure l'utilisation de modèles plus avancés, l'incorporation de davantage de données par le biais de techniques d'augmentation de données, ou la collecte d'un ensemble de données plus vaste et plus diversifié. Les travaux futurs pourraient explorer différentes modalités de capteurs, des architectures d'apprentissage profond plus sophistiquées ou des applications dans de nouveaux domaines. Il serait intéressant de se concentrer sur les

défis de la variabilité du monde réel, de la diversité des utilisateurs et de la reconnaissance continue des gestes.¹⁰⁷

Sources des citations

1. Convolutional neural network for gesture recognition human-computer interaction system design | PLOS One, consulté le avril 9, 2025, <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0311941>
2. A Real-time Hand Gesture Recognition System for Human-Computer and Human-Robot Interaction - CEUR-WS.org, consulté le avril 9, 2025, <https://ceur-ws.org/Vol-3398/p08.pdf>
3. Hand Gesture Recognition for Human-Computer Interaction Using Computer Vision, consulté le avril 9, 2025, https://www.researchgate.net/publication/369543591_Hand_Gesture_Recognition_for_Human-Computer_Interaction_Using_Computer_Vision
4. Hand Tracking and Gesture Recognition Using AI - Onix-Systems, consulté le avril 9, 2025, <https://onix-systems.com/blog/hand-tracking-and-gesture-recognition-using-ai>
5. What is gesture recognition in a car? Revolutionizing Automotive Interaction - BytePlus, consulté le avril 9, 2025, <https://www.byteplus.com/en/topic/110190>
6. Automotive Gesture Recognition Market reach \$12.2 Billion by 2035, Growing at 18.5% CAGR | Fact.MR Report, consulté le avril 9, 2025, <https://www.globenewswire.com/news-release/2025/03/05/3037310/0/en/Automotive-Gesture-Recognition-Market-reach-12-2-Billion-by-2035-Growing-at-18-5-CAGR-Fact-MR-Report.html>
7. Automotive Gesture Recognition Market Size, Growth Report 2034, consulté le avril 9, 2025, <https://www.gminsights.com/industry-analysis/automotive-gesture-recognition-market>
8. Exploring The Techniques and Applications of Hand Gesture Recognition for Human-Computer Interaction - IJSREM, consulté le avril 9, 2025, <https://ijsrem.com/download/exploring-the-techniques-and-applications-of-hand-gesture-recognition-for-human-computer-interaction/>
9. (PDF) Hand Gesture Recognition for Human Computer Interaction - ResearchGate, consulté le avril 9, 2025, https://www.researchgate.net/publication/320437730_Hand_Gesture_Recognition_for_Human_Computer_Interaction
10. www.cs.wm.edu, consulté le avril 9, 2025, https://www.cs.wm.edu/~gzhou/files/Gemote_CHASE17.pdf
11. Medical Gesture Recognition Method Based on Improved Lightweight Network - MDPI, consulté le avril 9, 2025, <https://www.mdpi.com/2076-3417/12/13/6414>
12. Innovative healthcare solutions: robust hand gesture recognition of daily life routines using 1D CNN - PubMed Central, consulté le avril 9, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC11322365/>

13. Patient Assistance System Based on Hand Gesture Recognition - Karl F. MacDorman, consulté le avril 9, 2025, <http://www.macdorman.com/kfm/writings/pubs/Dutta-2023-Patient-Assistance-System-IEEE-TIM.pdf>
14. A Real-Time Hand Gesture Interface for Medical Visualization Applications - CiteSeerX, consulté le avril 9, 2025, <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=f99df4fb2395d02f1ae6506302de976c79546061>
15. Gesture Recognition Technology for VR: Transforming User Experiences - BytePlus, consulté le avril 9, 2025, <https://www.byteplus.com/en/topic/240472>
16. Gesture Recognition in Virtual reality | Psychomachina - SCIDAC PLUS OPEN JOURNALS, consulté le avril 9, 2025, <https://journal.scidacplus.com/index.php/psychomachina/article/view/336>
17. Enhancing Virtual and Augmented Reality Interactions with a MediaPipe-Based Hand Gesture Recognition User Interface - ResearchGate, consulté le avril 9, 2025, https://www.researchgate.net/publication/372572118_Enhancing_Virtual_and_Augmented_Reality_Interactions_with_a_MediaPipe-Based_Hand_Gesture_Recognition_User_Interface
18. Gesture Recognition in Virtual reality - Semantic Scholar, consulté le avril 9, 2025, <https://pdfs.semanticscholar.org/8115/1207907afb87fc21f3aa6e78eff61c9b501f.pdf>
19. Static and Dynamic Hand Gestures: A Review of Techniques of Virtual Reality Manipulation, consulté le avril 9, 2025, <https://www.mdpi.com/1424-8220/24/12/3760>
20. A Framework for Real-Time Gestural Recognition and Augmented Reality for Industrial Applications - PubMed Central, consulté le avril 9, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC11054752/>
21. Hand gesture recognition technology brings real hand into virtual reality - Medium, consulté le avril 9, 2025, <https://medium.com/@Chengccc/hand-gesture-recognition-technology-brings-real-hand-into-virtual-reality-7b193d3a7173>
22. [2405.16264] Application of Artificial Intelligence in Hand Gesture Recognition with Virtual Reality: Survey and Analysis of Hand Gesture Hardware Selection - arXiv, consulté le avril 9, 2025, <https://arxiv.org/abs/2405.16264>
23. Application of Artificial Intelligence in Hand Gesture Recognition with Virtual Reality - arXiv, consulté le avril 9, 2025, <https://arxiv.org/html/2405.16264v1>
24. www.byteplus.com, consulté le avril 9, 2025, <https://www.byteplus.com/en/topic/110190#:~:text=Some%20practical%20applications%20include%3A,calls%20with%20simple%20hand%20signals>
25. Automotive Infotainment Gesture Control for the Rest of Us - Analog Devices, consulté le avril 9, 2025, <https://www.analog.com/en/resources/technical-articles/automotive-infotainment-gesture-control-for-the-rest-of-us.html>

26. What Is Gesture Recognition? - Aptiv, consulté le avril 9, 2025, <https://www.aptiv.com/en/insights/article/what-is-gesture-recognition>
27. Automotive gesture recognition - Elmos Semiconductor SE, consulté le avril 9, 2025, <https://www.elmos.com/automotive-gesture-recognition/>
28. Gesture Recognition, Proximity Sensors Drive Advances in Automotive Infotainment - Avnet, consulté le avril 9, 2025, <https://www.avnet.com/wps/portal/apac/resources/article/gesture-recognition-proximity-sensors-drive-advances-auto-infotainment/>
29. Hand Gestures Dataset - Figshare, consulté le avril 9, 2025, https://figshare.com/articles/dataset/Hand_Gestures_Dataset/24449197
30. Sign Language Hand Gesture Recognition - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/gauravduttakiit/sign-language-hand-gesture-recognition>
31. A Comprehensive Review of Sensor-based Sign Language Recognition Models, consulté le avril 9, 2025, <https://www.xisdxjsu.asia/V19I05-22.pdf>
32. Sign Language Recognition Using the Electromyographic Signal: A Systematic Literature Review - PubMed, consulté le avril 9, 2025, <https://pubmed.ncbi.nlm.nih.gov/37837173/>
33. Wearable Sensor-Based Sign Language Recognition: A Comprehensive Review - PubMed, consulté le avril 9, 2025, <https://pubmed.ncbi.nlm.nih.gov/32845843/>
34. Emerging Wearable Interfaces and Algorithms for Hand Gesture Recognition: A Survey, consulté le avril 9, 2025, <https://pubmed.ncbi.nlm.nih.gov/33961564/>
35. A Review of Hand Gesture Recognition Systems Based on Noninvasive Wearable Sensors, consulté le avril 9, 2025, https://ro.uow.edu.au/articles/journal_contribution/A_Review_of_Hand_Gesture_Recognition_Systems_Based_on_Noninvasive_Wearable_Sensors/27817278
36. Building a Gesture Recognition System using Deep Learning - Joanna Materzyńska, consulté le avril 9, 2025, <https://www.youtube.com/watch?v=keffWSqi67w>
37. Hand gesture dataset | IEEE DataPort, consulté le avril 9, 2025, <https://ieee-dataport.org/keywords/hand-gesture-dataset>
38. Datasets - Hand Gesture Recognition - Papers With Code, consulté le avril 9, 2025, <https://paperswithcode.com/datasets?task=hand-gesture-recognition>
39. Machine Learning Datasets - Papers With Code, consulté le avril 9, 2025, <https://paperswithcode.com/datasets?task=gesture-recognition>
40. i need databases for hand gesture recognition - MATLAB Answers, consulté le avril 9, 2025, <https://nl.mathworks.com/matlabcentral/answers/2062812-i-need-databases-for-hand-gesture-recognition>
41. Datasets - Qualcomm, consulté le avril 9, 2025, <https://www.qualcomm.com/developer/artificial-intelligence/datasets>
42. i need databases for hand gesture recognition - MATLAB Answers, consulté le avril 9, 2025, <https://www.mathworks.com/matlabcentral/answers/2062812-i-need-databases-for-hand-gesture-recognition>

43. Datasets - UCI Machine Learning Repository, consulté le avril 9, 2025, <https://archive.ics.uci.edu/datasets>
44. Hand Gesture Recognition Database - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/gti-upm/leapgestrecog>
45. Reusing image dataset to get hand data for gesture recognition : r/leapmotion - Reddit, consulté le avril 9, 2025, https://www.reddit.com/r/leapmotion/comments/122d87w/reusing_image_dataset_to_get_hand_data_for/
46. Hand Gesture Recognition Database - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/gti-upm/leapgestrecog/data>
47. Multi-Modal Dataset for Hand Gesture Recognition - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/gti-upm/multimodhandgestrec>
48. Gesture Recognition and Biometrics ElectroMyogram (GRABMyo) v1.0.0 - PhysioNet, consulté le avril 9, 2025, <https://physionet.org/content/grabmyo/1.0.0/>
49. EMG Data for Gestures - UCI Machine Learning Repository, consulté le avril 9, 2025, <https://archive.ics.uci.edu/ml/datasets/EMG+data+for+gestures>
50. EMG Signal for gesture recognition - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/sojanprajapati/emg-signal-for-gesture-recognition>
51. EMG-Based Hand Gesture Recognition through Diverse Domain Feature Enhancement and Machine Learning-Based Approach - arXiv, consulté le avril 9, 2025, <https://arxiv.org/html/2408.13723v1>
52. Motion Capture Hand Postures - UCI Machine Learning Repository, consulté le avril 9, 2025, <https://archive.ics.uci.edu/ml/datasets/Motion+Capture+Hand+Postures>
53. OPPORTUNITY Activity Recognition - UCI Machine Learning Repository, consulté le avril 9, 2025, <https://archive.ics.uci.edu/ml/datasets/opportunity+activity+recognition>
54. hukenovs/hagrid: HAnd Gesture Recognition Image Dataset - GitHub, consulté le avril 9, 2025, <https://github.com/hukenovs/hagrid>
55. HaGRID - HAnd Gesture Recognition Image Dataset - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/kapitanov/hagrid>
56. Gesture Recognition - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/imsparsh/gesture-recognition>
57. Training a hand gesture recognition model for contactless human-computer interaction - GitHub, consulté le avril 9, 2025, <https://github.com/JieYing-99/Hand-Gesture-Recognition-for-HCI>
58. HandLogin Dataset - Visual Information Processing - Boston University, consulté le avril 9, 2025, <https://vip.bu.edu/projects/hcis/hand-login/dataset/>
59. A Public Dataset of Dynamic Hand-gestures Acquired using Impulse-radar sensors., consulté le avril 9, 2025, https://figshare.com/articles/dataset/A_Public_Dataset_of_Dynamic_Hand-gestures_Acquired_using_Impulse-radar_sensors_/12652592

60. Gesture Recognition - 10,000+ videos - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/unidatapro/gesture-recognition>
61. Hand Gesture Recognition Dataset - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/aryarishabh/hand-gesture-recognition-dataset>
62. Hand Gesture Recognition Database - Kaggle, consulté le avril 9, 2025, <https://www.kaggle.com/datasets/shivamgoyal1899/hand-gesture-recognition-database>
63. This project implements a hand gesture recognition system using deep learning techniques with TensorFlow and Keras. It involves preprocessing a dataset from Kaggle, building and training a convolutional neural network (CNN) model. - GitHub, consulté le avril 9, 2025, <https://github.com/SubhangiSati/Hand-Gesture-Recommendation-System>
64. Hand-Gesture Recognition with PyTorch: A Comprehensive Tutorial | Data-Driven Science, consulté le avril 9, 2025, <https://datadrivenscience.com/hand-gesture-recognition-with-pytorch-a-comprehensive-tutorial/>
65. Real-time Hand Gesture Recognition using TensorFlow & OpenCV - TechVidvan, consulté le avril 9, 2025, <https://techvidvan.com/tutorials/hand-gesture-recognition-tensorflow-opencv/>
66. How I Built a Hand Gesture Recognition Model in Python — Part 1 | by Odilbek Tokhirov, consulté le avril 9, 2025, <https://medium.com/@odil.tokhirov/how-i-built-a-hand-gesture-recognition-model-in-python-part-1-db378cf196e6>
67. Gesture recognition task guide | Google AI Edge - Gemini API, consulté le avril 9, 2025, https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer
68. mikhail-vlasenko/grib: Gesture recognition library for Python - GitHub, consulté le avril 9, 2025, <https://github.com/mikhail-vlasenko/grib>
69. Mastering Hand Gesture Recognition with Python: Build Your Own Touchless Systems, consulté le avril 9, 2025, <https://jaystechbites.com/posts/2024/master-hand-gesture-recognition-python-projects/>
70. Hand Recognition with Python: Guide with Code Samples | by Lutfunnahar Lota | Medium, consulté le avril 9, 2025, <https://lotalutfunnahar.medium.com/hand-recognition-with-python-guide-with-code-samples-a0b17f4cd813>
71. GRLib: An Open-Source Hand Gesture Detection and Recognition Python Library, consulté le avril 9, 2025, <https://paperswithcode.com/paper/grib-an-open-source-hand-gesture-detection>
72. GRLib: An Open-Source Hand Gesture Detection and Recognition Python Library - arXiv, consulté le avril 9, 2025, <https://arxiv.org/pdf/2310.14919>
73. Gesture recognition guide for Python | Google AI Edge - Gemini API, consulté le avril 9, 2025, https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer/python
74. Hand Gesture Recognition in python with source code - ITDesigners, consulté le avril 9, 2025, <https://itdesigners.org/hand-gesture-recognition-in-python/>

75. Lab Notes: MediaPipe with Python for Gesture Recognition - Growth Acceleration Partners, consulté le avril 9, 2025, <https://www.growthaccelerationpartners.com/blog/lab-notes-mediapipe-with-python-for-gesture-recognition>
76. Arjun-08/Hand-gesture-recognition-using-Machine-learning-techniques - GitHub, consulté le avril 9, 2025, <https://github.com/Arjun-08/Hand-gesture-recognition-using-Machine-learning-techniques>
77. Hand Tracking with MediaPipe and OpenCV | Step-by-Step - YouTube, consulté le avril 9, 2025, <https://www.youtube.com/watch?v=RRBXVu5UE-U>
78. Custom Hand Gesture Recognition with Hand Landmarks Using Google's Mediapipe + OpenCV in Python - YouTube, consulté le avril 9, 2025, https://www.youtube.com/watch?v=a99p_fAr6e4
79. Hand landmarks detection guide for Python | Google AI Edge - Gemini API, consulté le avril 9, 2025, https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker/python
80. Real-Time Hand Gesture Recognition with Mediapipe and Tensorflow | #python - YouTube, consulté le avril 9, 2025, <https://www.youtube.com/watch?v=OW4nRBPu1hQ>
81. Hand Detection Tracking in Python using OpenCV and MediaPipe | by Aditee Gautam, consulté le avril 9, 2025, <https://gautamaditee.medium.com/hand-recognition-using-opencv-a7b109941c88>
82. Improved Gesture Recognition in Python and MediaPipe | Technology Tutorials, consulté le avril 9, 2025, <https://toptechboy.com/improved-gesture-recognition-in-python-and-mediapipe/>
83. Mediapipe Gesture Recognition - Gesture Control in Zoom Call - LearnOpenCV, consulté le avril 9, 2025, <https://learnopencv.com/gesture-control-in-zoom-call-using-mediapipe/>
84. Hand Recognition and Finger Identification with Raspberry Pi and OpenCV - Guides, consulté le avril 9, 2025, <https://forum.core-electronics.com.au/t/hand-recognition-and-finger-identification-with-raspberry-pi-and-opencv/12705>
85. ishfulthinking/Python-Hand-Gesture-Recognition: A guide to using Python and OpenCV2 to create an application that can recognize hand gestures/movements. - GitHub, consulté le avril 9, 2025, <https://github.com/ishfulthinking/Python-Hand-Gesture-Recognition>
86. AI Hands Pose Recognition with Python Tutorial - How to Control Volume with Computer Vision - YouTube, consulté le avril 9, 2025, <https://www.youtube.com/watch?v=HlkfS4qL8WQ>
87. Human Hand Movement Classification based on EMG Signal using different Feature Extractor - Biomedical and Pharmacology Journal, consulté le avril 9, 2025, <https://biomedpharmajournal.org/vol17no1/human-hand-movement-classification-based-on-emg-signal-using-different-feature-extractor/>
88. Effective features extraction and selection for hand gesture recognition using sEMG signal, consulté le avril 9, 2025,

- <https://www.researchgate.net/publication/380937736> Effective features extraction and selection for hand gesture recognition using sEMG signal
89. SEMG Feature Extraction Based on Stockwell Transform Improves Hand Movement Recognition Accuracy - PMC, consulté le avril 9, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC6832976/>
 90. A New Feature Extraction Method for EMG Signals - IJETA, consulté le avril 9, 2025, <https://www.ijeta.org/download/file/fid/84191>
 91. Classification and Feature Extraction of Different Hand Movements from the EMG Signal using Machine Learning based Algorithms A T, consulté le avril 9, 2025, <https://conservancy.umn.edu/bitstreams/ee9a6b5b-1907-4ba3-8f6a-bc56a28e114e/download>
 92. Machine Learning-Based Feature Extraction and Classification of EMG Signals for Intuitive Prosthetic Control - MDPI, consulté le avril 9, 2025, <https://www.mdpi.com/2076-3417/14/13/5784>
 93. A list of EMG feature extraction techniques. | Download Table - ResearchGate, consulté le avril 9, 2025, https://www.researchgate.net/figure/A-list-of-EMG-feature-extraction-techniques_tbl1_321598414
 94. Sensor-based gesture recognition with convolutional neural networks - ResearchGate, consulté le avril 9, 2025, <https://www.researchgate.net/publication/372198782> Sensor-based gesture recognition with convolutional neural networks
 95. Electromyogram-Based Classification of Hand and Finger Gestures Using Artificial Neural Networks - PubMed Central, consulté le avril 9, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC8749583/>
 96. Machine Learning-based sEMG Signal Classification for Hand Gesture Recognition - arXiv, consulté le avril 9, 2025, <https://arxiv.org/html/2411.15655v1>
 97. EMG Based Gesture Recognition Using Machine Learning | Request PDF - ResearchGate, consulté le avril 9, 2025, <https://www.researchgate.net/publication/331681666> EMG Based Gesture Recognition Using Machine Learning
 98. Enhanced Hand Gesture Recognition with Surface Electromyogram and Machine Learning, consulté le avril 9, 2025, <https://www.mdpi.com/1424-8220/24/16/5231>
 99. EMG-Based Dynamic Hand Gesture Recognition Using Edge AI for Human-Robot Interaction - MDPI, consulté le avril 9, 2025, <https://www.mdpi.com/2079-9292/12/7/1541>
 100. Gesture Recognition Using Surface Electromyography and Deep Learning for Prostheses Hand: State-of-the-Art, Challenges, and Future - Frontiers, consulté le avril 9, 2025, <https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2021.621885/full>
 101. Deep Learning for EMG-based Human-Machine Interaction: A Review - ResearchGate, consulté le avril 9, 2025, <https://www.researchgate.net/publication/349179783> Deep Learning for EMG-

[based Human-Machine Interaction A Review](#)

102. The source code for the real-time hand gesture recognition algorithm based on Temporal Muscle Activation maps of multi-channel surface electromyography (sEMG) signals (ICASSP 2021) - GitHub, consulté le avril 9, 2025, <https://github.com/Laknath1996/sEMG-Hand-Gesture-Recognition>
103. EMG Gesture Recognition - Papers With Code, consulté le avril 9, 2025, <https://paperswithcode.com/task/emg-gesture-recognition>
104. tsagkas/sEMG-HandGestureRecognition: Hand Gesture Recognition via sEMG signals with CNNs (Electrical and Computer Engineering - MSc Thesis) - GitHub, consulté le avril 9, 2025, <https://github.com/tsagkas/sEMG-HandGestureRecognition>
105. A Review of Hand Gesture Recognition Systems Based on Noninvasive Wearable Sensors, consulté le avril 9, 2025, <https://www.semanticscholar.org/paper/0e1080d1a42024600d94708d1c513c50b596dd65>
106. Hand Gesture Recognition Based on Computer Vision: A Review of Techniques - PMC, consulté le avril 9, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC8321080/>
107. A Structured and Methodological Review on Vision-Based Hand Gesture Recognition System - PMC, consulté le avril 9, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC9224857/>
108. Gesture Recognition Methods Using Sensors Integrated into Smartwatches: Results of a Systematic Literature Review | Anais do Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais (IHC) - Sociedade Brasileira de Computação (SBC), consulté le avril 9, 2025, <https://sol.sbc.org.br/index.php/ihc/article/view/27604>
109. Sensor-Based Hand Gesture Detection and Recognition by Key Intervals - MDPI, consulté le avril 9, 2025, <https://www.mdpi.com/2076-3417/12/15/7410>
110. A Review on Gesture Recognition Techniques for Human-Robot Collaboration - IRJET, consulté le avril 9, 2025, <https://www.irjet.net/archives/V9/i10/IRJET-V9I1097.pdf>
111. Machine-learned wearable sensors for real-time hand-motion recognition: toward practical applications - Oxford Academic, consulté le avril 9, 2025, <https://academic.oup.com/nsr/article/11/2/nwad298/7452905>
112. Hand-Gesture Recognition Based on EMG and Event-Based Camera Sensor Fusion: A Benchmark in Neuromorphic Computing - Frontiers, consulté le avril 9, 2025, <https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2020.00637/full>
113. Towards Robust and Interpretable EMG-based Hand Gesture Recognition using Deep Metric Meta Learning - arXiv, consulté le avril 9, 2025, <https://arxiv.org/html/2404.15360v1>
114. Survey on Hand Gesture Recognition from Visual Input The research leading to these results received funding from the European Commission under Grant

- Agreement No. 101168042 (TRIFFID). - arXiv, consulté le avril 9, 2025,
<https://arxiv.org/html/2501.11992v1>
115. PPG-based Finger-level Gesture Recognition Leveraging Wearables - WINLAB, Rutgers University, consulté le avril 9, 2025,
<https://www.winlab.rutgers.edu/~yychen/papers/PPG%20based%20Finger%20level%20Gesture%20Recognition%20Leveraging%20Wearables.pdf>
116. Wearable Sensor-based Hand Gesture Recognition of Construction Workers, consulté le avril 9, 2025,
https://www.iaarc.org/publications/2021_proceedings_of_the_38th_isarc/wearable_sensor_based_hand_gesture_recognition_of_construction_workers.html
117. A Survey on Multimodal Wearable Sensor-based Human Action Recognition - arXiv, consulté le avril 9, 2025, <https://arxiv.org/html/2404.15349>
118. A Survey on Hand Pose Estimation with Wearable Sensors and Computer-Vision-Based Methods - MDPI, consulté le avril 9, 2025,
<https://www.mdpi.com/1424-8220/20/4/1074>
119. Hand Gesture Recognition with Python - RIS AI, consulté le avril 9, 2025,
<https://www.ris-ai.com/hand-gesture-recognition>
120. louisc-s/Machine-Learning-Technologies-on-EMG-for-Hand-Gesture-Recognition: Machine learning code implemented for hand gesture recognition using EMG data from the Ninapro db1 database. Report Link:
<https://drive.google.com/file/d/1vHzUKKFz1ifAaLOW91Sb41zo3zN5qRJI/view?usp=sharing> - GitHub, consulté le avril 9, 2025, <https://github.com/louisc-s/Machine-Learning-Technologies-on-EMG-for-Hand-Gesture-Recognition>
121. The Challenges and Opportunities of Gesture Recognition - nexocode, consulté le avril 9, 2025, <https://nexocode.com/blog/posts/gestures-recognition-challenges-and-opportunities/>
122. (PDF) A comprehensive analysis of gesture recognition systems: Advancements, challenges, and future direct - ResearchGate, consulté le avril 9, 2025,
https://www.researchgate.net/publication/382370043_A_comprehensive_analysis_of_gesture_recognition_systems_Advancements_challenges_and_future_direct
123. Challenges and Applications for Hand Gesture Recognition - IGI Global, consulté le avril 9, 2025, <https://www.igi-global.com/book/challenges-applications-hand-gesture-recognition/277793>
124. Advancements and Challenges in Hand Gesture Recognition: A Comprehensive Review, consulté le avril 9, 2025,
https://www.researchgate.net/publication/381985305_Advancements_and_Challenges_in_Hand_Gesture_Recognition_A_Comprehensive_Review