

: N° d'ordre  
: N° de série

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



UNIVERSITE ECHAHID HAMMA LAKHDAR - EL OUED  
FACULTÉ DES SCIENCES EXACTES  
Département D'Informatique



Mémoire de Fin D'étude  
Présenté pour l'obtention du Diplôme de

## MASTER ACADEMIQUE

Domaine : **Mathématique et Informatique**  
Filière : **Informatique**  
Spécialité : **Systèmes Distribués et Intelligence Artificielle**

### Thème

**Développement d'un système de  
vérification de la prière à  
base de deep learning**

Présenté par :

- CHAMI AMEL
- BEKKOUCHE SALIMA

Suivie par :

M. ZAIZ FAOUZI

Soutenue le :08-06- 2023

Devant le jury:

M.	NEDIOUI ABDELHAMID	MAA	Président
M.	OTHMANI SAMIR	MAA	Rapporteur

Année Universitaire: 2022/2023

# Dédicaces

*Tout d'abord, je veux rendre grâce à Dieu, le Clément et le Très Miséricordieux pour son amour éternel. C'est ainsi que je dédie cette thèse à:*

*ma mère pour sa tendresse et mon père pour sa patience et encouragement  
mes très chers frères et ma chère soeur pour leurs conseils,  
mes cousins et cousines,  
tous ceux que j'aime,  
tous mes amis.*

# Remerciements

*Nous tenons remercier avant tout **DIEU** qui nous a donné la force, la volonté , le courage, et la patience de pouvoir réaliser ce modeste travail.*

*On tient aussi remercier **Docteur ZAIZ Faouzi** ,son précieux conseil et son aide durant toute la période du travail. Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions. Nos remerciements s'étendent également à tous nos enseignants durant les années des études cœur.*

*On remercie également tous ceux qui ont participé de près ou de loin à élaborer ce travail*

# Résumé

L'islam a accordé à la prière un statut élevé, car c'est la première chose que Dieu a commandée en matière de culte, et c'est le premier culte pour lequel un musulman est tenu responsable. Malheureusement, il y a beaucoup de facteurs qui affectent la mémoire humaine et lui font oublier (Ex : Alzheimer, vieillissement, stress et manque de sommeil). Cela affecte négativement l'exécution correcte de la prière. Dans ce travail, nous sommes intéressés à proposer un système intelligent qui utilise les techniques d'IA dans le but de vérifier la validité de la séquence de la prière. Le système doit apprendre à extraire des poses clés afin de réaliser la tâche en temps réel. Ceci est réalisé en utilisant les extraits des caractéristiques à partir des points clés obtenus à l'aide de mediapipe puis traité les données extraites avec un classificateur (Forêt Aléatoire, SVM, KNN, VGG16). Enfin, nous vérifions les résultats avec une séquence référence de prière. Les résultats obtenus sont très encourageant, mais il est possible de faire des améliorations sur le système.

**Mots Clés:** prière, Deep-learning, Machine learning, mediapipe.

# Abstract

Islam has given prayer a high status because it is the first thing God has commanded in matters of worship, and it is the first worship for which a Muslim is held accountable. Unfortunately, there are many factors that affect human memory and make it forget (Ex: Alzheimer's, aging, stress and lack of sleep). This negatively affects the correct performance of the prayer. In this work, we are interested in proposing an intelligent system that uses AI techniques in order to verify the validity of the prayer sequence. The system must learn to extract key poses (tracking) in order to perform the task in real time. This is achieved by using posture Information extracted from features from key points obtained using mediapipe. Instead of general features of distances and angles, geometric moments of posture is used as features for pose classification and then processed the extracted data with a classifier (Random Forest, SVM, KNN, VGG16). Finally, we check the results with a reference prayer sequence. The obtained results are very encouraging, but still possible to do some improvements to the system.

**Keywords:** prayer, Deep-learning, Machine learning, mediapipe.

## الملخص

لقد جعل الإسلام الصلاة مكانة عالية لأنها أول ما يأمر به الله في العبادة ، وهي أول عبادة يحاسب عليها المسلم. للأسف هناك العديد من العوامل التي تؤثر على ذاكرة الإنسان وتجعله ينسى (مثل : الزهايمر، الشيخوخة ،الإجهاد وقلة النوم).مما هذا يؤثر سلبا على الأداء الصحيح للصلاة. في هذا العمل ، نحن مهتمون باقتراح نظام ذكي يستخدم تقنيات الذكاء الاصطناعي للتحقق من صحة تسلسل الصلاة. يجب أن يتعلم النظام استخراج الأوضاع الرئيسية من أجل أداء المهمة في الوقت الفعلي. يتم تحقيق ذلك باستخدام مقتطفات الميزات من النقاط الرئيسية التي تم الحصول عليها باستخدام ميديايبب ثم معالجة البيانات المستخرجة باستخدام مصنف (الغابة عشوائية ، شعاع الدعم الآلي ، كي أقرب جار ،مجموعة الهندسة المرئية ١٦ ).

و أخيراً التحقق من النتائج باستخدام قائمة مرجعية تحتوي على تسلسل لوضعيات الصلاة . النتائج التي تم الحصول عليها مشجعة للغاية ، ولكن من الممكن إجراء تحسينات على النظام.

**كلمات مفتاحية:** الصلاة ، التعلم العميق، التعلم الآلي ، ميديايبب .

# Table des matières

<b>Dédicaces</b>	<b>i</b>
<b>Remerciements</b>	<b>ii</b>
<b>Résumé</b>	<b>iii</b>
<b>Table des matières</b>	<b>vi</b>
<b>Liste des figures</b>	<b>ix</b>
<b>Liste des tableaux</b>	<b>1</b>
<b>Introduction</b>	<b>2</b>
<b>1 Etat de l'art</b>	<b>4</b>
Introduction . . . . .	4
1.1 Reconnaissance des actions . . . . .	4
1.1.1 Approches d'extraction des primitives . . . . .	4
1.1.2 Approches basées sur des caractéristiques locales : . . . . .	8
1.2 Analyse des mouvements humain Détection et Suivi . . . . .	10
1.2.1 La Détection . . . . .	10
1.2.2 Le Suivi . . . . .	12
1.3 Estimation de mouvement . . . . .	13
1.3.1 Sans modèle . . . . .	14
1.3.2 Assemblage probabiliste de parties . . . . .	14
1.3.3 Méthodes basées exemple . . . . .	14
1.3.4 Utilisation indirecte de modèles . . . . .	14

1.3.5	Utilisation directe de modèles . . . . .	15
1.4	Collecte de données . . . . .	15
1.4.1	Qualité des données . . . . .	16
1.4.2	quantité de données . . . . .	16
1.4.3	Intégration des données . . . . .	16
1.4.4	Annotation (Étiquetage) des données . . . . .	17
1.5	Pré-traitement de données . . . . .	17
1.5.1	Techniques Prétraitement des données d'image . . . . .	18
1.6	Intelligence artificielle . . . . .	19
1.6.1	Apprentissage automatique . . . . .	19
1.6.2	Apprentissage profond . . . . .	21
1.7	La prière . . . . .	22
1.8	Correction prière . . . . .	22
1.9	Travaux récents sur la reconnaissance de l'activité prière . . . . .	23
	Conclusion . . . . .	26
<b>2</b>	<b>Conception du système</b> . . . . .	<b>27</b>
	Introduction . . . . .	27
2.1	Choix de la prière . . . . .	27
2.2	calcul du nombre des raka3ats . . . . .	29
2.3	Générationla séquence de référence . . . . .	29
2.4	Système de vérification . . . . .	30
2.4.1	Extraction des points d'intérêts à l'aide de MediaPipe . . . . .	31
2.5	Extraction des points d'intérêts à l'aide YOLO . . . . .	32
2.5.1	YOLO . . . . .	32
2.5.2	YOLOv7 . . . . .	32
2.6	Comparaison entre mediapipe et YOLOv7 . . . . .	32
2.6.1	Phase d'apprentissage . . . . .	32
2.6.2	Algorithmes utilisés . . . . .	39
2.6.3	Phase de prédiction . . . . .	39
2.6.4	Vérifier la correction de prière . . . . .	41
	Conclusion . . . . .	42

<b>3 Tests et Résultats</b>	<b>43</b>
Introduction . . . . .	43
3.1 Environnement matériels(Hardware) . . . . .	43
3.2 Environnement logiciel (Software) . . . . .	43
3.3 Langage de programmation et bibliothèques . . . . .	44
3.4 Résultats et discussion . . . . .	45
Conclusion . . . . .	47
 <b>Conclusion et perspectives</b>	 <b>48</b>
 <b>Bibliographie</b>	 <b>49</b>

# Liste des figures

<b>1</b>	<b>Etat de l'art</b>	<b>4</b>
1.1	Illustration du mouvement des points repères[27]. . . . .	5
1.2	Modèles du corps humain (a) l'image originale de corps, (b) modèle cinématique basé sur 13 articulations, (c) le modèle 3D [29]. . . . .	6
1.3	Illustration de modèle holistique et volume temporel[21]. . . . .	7
1.4	histogramme de gradients orientés[24]. . . . .	8
1.5	Schéma de décomposition du domaine de l'intelligence artificielle et de ces sous-domaines[41]. . . . .	19
1.6	Les différents types d'apprentissage automatique[45]. . . . .	20
<b>2</b>	<b>Conception du système</b>	<b>27</b>
2.1	Schéma général du système de vérification le séquence de la prière. . . . .	27
2.2	Exemple du fichier XML. . . . .	28
2.3	Illustration du calendrier de la ministère des affaires religieuses. . . . .	28
2.4	L'architecture du système de vérification . . . . .	30
2.5	Extraction des points d'intérêts à l'aide de MediaPipe . . . . .	31
2.6	Posture julus . . . . .	33
2.7	Posture quiam . . . . .	33
2.8	Posture ruku . . . . .	33
2.9	Posture sujud . . . . .	33
2.10	Les points clés pour chaque repère de posture quiam . . . . .	36
2.11	Les points clés pour chaque repère de posture ruku . . . . .	36
2.12	Accuracy score du model . . . . .	37
2.13	La matrice de confusion de model . . . . .	38
2.14	Rapport de classement Le score F1 . . . . .	38
2.15	Posture quiam . . . . .	40
2.16	Posture quiam . . . . .	40
2.17	Posture de ruku . . . . .	40

2.18 Posture de ruku . . . . .	40
2.19 Posture de sujud . . . . .	40
2.20 Posture de sujud . . . . .	40
2.21 Posture de jujus . . . . .	41
2.22 Posture de julus . . . . .	41
2.23 Resultat de systeme de verification . . . . .	41

# Liste des tableaux

<b>1</b>	<b>Etat de l'art</b>	<b>4</b>
1.1	Les cinq prière . . . . .	22
<b>2</b>	<b>Conception du système</b>	<b>27</b>
2.1	Comparaison entre mediapipe et YOLOv7. . . . .	32
2.2	Nombre d'images collectées pour chaque posture . . . . .	33
2.3	Comparision les résultat entre les deffirents algoithmes . . . . .	39
<b>3</b>	<b>Tests et Résultats</b>	<b>43</b>
3.1	Taux détection les postures de quiam. . . . .	45
3.2	Taux détection les postures de quiam. . . . .	45
3.3	Taux détection les Postures de ruku. . . . .	45
3.4	Taux détection les postures de sujud. . . . .	46
3.5	Taux détection les postures de julus. . . . .	46
3.6	Taux détection les postures de julus. . . . .	46

# Introduction

La prière (Salat) est l'un des principaux piliers de l'islam et est considérée comme l'un des aspects les plus importants de notre foi. Pendant ce temps, les musulmans apprennent à accomplir des prières selon les ordres et les actions du prophète Mahomet. Cela se fait en observant les mouvements et les commandes directement en utilisant la vue et l'ouïe. En d'autres termes, la technologie utilisée pour apprendre la salat à l'époque ne détectait que les actions et les mots corrects en utilisant les sens humains. Alors que les musulmans de nos jours ne peuvent apprendre à prier qu'à travers les paroles et les actes du Prophète, à mesure que le monde vieillit, certains musulmans ont tendance à oublier la bonne façon de prier en raison de contraintes séculaires.

Aujourd'hui, la technologie se développe à un rythme très rapide. De nombreux types de recherche et de développement ont été réalisés pour améliorer nos vies. Cela place une grande responsabilité sur les érudits musulmans pour développer des technologies au profit de ce monde et du monde à venir. À cette fin, le "Système intelligent de surveillance et de formation à la prière" a été développé comme un outil pédagogique qui peut aider de nombreux musulmans à apprendre et à reconnaître la bonne façon de prier. Par conséquent, pour effectuer une inspection de la prière et un système de formation pour la communauté musulmane, la technologie la plus importante que nous devons apprendre et maîtriser est la vision artificielle et le traitement d'image .

La vision par ordinateur est utilisée pour examiner et suivre les mouvements humains dans divers domaines tels que le sport, la santé et même les jeux. En tant que musulman, nous remarquons et comprenons généralement comment nous accomplissons la prière en suivant les autres. Nous suivons les postures et les mouvements des autres dans la prière principalement par le biais du balayage oculaire. Ensuite, nous analysons et traitons si l'apprentissage de la prière est correct ou faux. Cependant, en combinant cette technologie avec l'aspect religieux, nous obtenons beaucoup d'avantages. En utilisant ce système, nous pouvons apprendre la manière correcte de salat en regardant la posture correcte de salat installée dans le système de base de données ; par conséquent, Donnez

aux utilisateurs des commentaires appropriés.

De nombreux chercheurs ont trouvé des algorithmes pour détecter des parties du corps humain, telles que les visages, les mains, les mouvements et les poses. Certains algorithmes peuvent détecter les poses humaines. Dans ce travail, nous proposons un système de vérification de séquence de la prière. Ce dernier commence par l'extraction d'un ensemble de points d'articulation à partir de chaque frame du vidéo à l'aide de mediapipe, puis les stockons dans une dataset. Par la suite, il fait la classification de nouvelle posture à l'aide d'un classificateur forêt aléatoire . le modèle obtenu est utilisé pour vérifier la validité sequence de différent posture de la prière.

Le reste du document est organisé comme suit :

- Dans le premier chapitre, nous avons essayé de donner un aperçu les approches et méthodes utilisées dans le domaine de reconnaissance d'action utilisées.
- Dans le deuxième chapitre, nous avons donné une description des différents modules du système proposé.
- Le troisième chapitre, présente les résultats obtenus après avoir testé le système, ainsi qu'une discussion de ces résultats obtenus.
- Puis nous terminons par une conclusion général, qui présente un synthèse du travail et les améliorations et les perspectives que nous pouvons faire sur le système proposé.

# Etat de l'art

## Introduction

La reconnaissance de l'activité humaine (HAR) est un sujet de recherche très actif depuis deux décennies en raison de ses applications dans divers domaines tels que la santé, la surveillance à distance, les jeux, la sécurité et la surveillance, et l'interaction homme-ordinateur. La reconnaissance d'activité peut être définie comme la capacité à reconnaître ou détecter les activités en cours sur la base des informations reçues de différents capteurs. Ces capteurs peuvent être des caméras, des capteurs portables, des capteurs fixés sur des objets du quotidien ou déployés dans l'environnement.

Dans ce chapitre, nous allons présenter un aperçu des principales sur les approches de reconnaissance d'action utilisées dans l'état de l'art les analyse le mouvement humaine différent type de l'apprentissage ,la perire dans islam ,finallement queleque exemple de traveaux récents sur la reconnaissance de la prière

## 1.1 Reconnaissance des actions

La reconnaissance d'action consiste généralement en une étape d'extraction brute et une étape de classification. L'extraction primitive consiste à identifier des caractéristiques uniques à partir de séquences vidéo tout en étant robuste au bruit. Dans l'étape de classification, nous nous intéressons à la possibilité d'identifier des actions à l'aide de méthodes d'apprentissage automatique, en tenant compte de la grande variabilité à laquelle les classes d'actions peuvent être exposées, en particulier si elles sont effectuées par différents sujets de sexe et de taille différents, et avec des vitesses et des vitesses différentes [20].

### 1.1.1 Approches d'extraction des primitives

Cette approche considéré comme l'une des tâches principales pour la reconnaissance des actions à partir de séquences vidéo est l'extraction des primitives qui caractérise le

mouvement dans ces séquences. Généralement, on peut distinguer trois types d'approches d'extraction de primitives sont :[42]

- Approches basées sur un modèle du corps humain
- Approches holistiques
- Approches basées sur des caractéristiques locales

#### 1.1.1.1 Approches basées sur un modèle du corps humain

Cette approche se base sur une étude psychophysique sur l'interprétation visuelle du mouvement biologique qui montre que les humains sont capables de reconnaître les actions uniquement à partir de quelques points de repère en mouvement «Moving light displays » attachés au corps humain (voir la figure 1.1). Plusieurs travaux de reconnaissance des actions se sont inspirés de cette étude et ont utilisé une représentation similaire basée sur les trajectoires des points repères sur le corps humain, tels que la tête, les mains, les jambes, etc. Cette étude a été également à l'origine des travaux qui se basent sur des modèles de mouvement du corps [27].

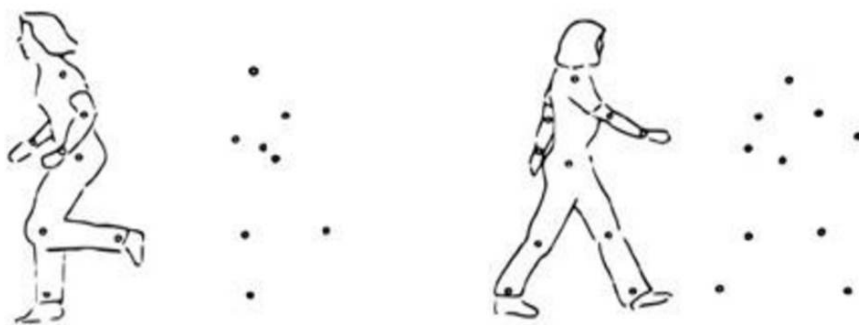


Figure 1.1: Illustration du mouvement des points repères[27].

La reconstitution du corps humain peut être réalisée à partir d'un modèle cinématique des articulations de corps humain. Ainsi, la reconnaissance est déterminée par l'évolution et la trajectoire des articulations. La difficulté majeure de cette modélisation est le grand nombre de degrés de liberté possibles des différentes articulations du corps humain et les grandes variabilités des formes de corps humain[42]. En outre, Ke et al [29] proposent une méthode de reconstruction de pose 3D en temps réel (Figure 1.2 ). L'apparence, la couleur et l'information temporelle sont conjointement utilisées pour effectuer un suivi des parties du corps humain en 2D. Ensuite, l'algorithme d'optimisation « downhill simplex » est utilisé pour faire l'alignement entre les primitives 2D suivies et des modèles humains 3D prédéfinis. Cette méthode est invariante par rapport aux angles de vue qui est l'un des défis les plus importants dans la reconnaissance d'actions. Cependant, elle est sujette

aux erreurs précoces dans l'étape de suivi. Ramanan et Forsyth [40] ont aussi utilisé un apprentissage de capture de mouvements 3D et une reconnaissance de mouvements se basant sur une capture de mouvements 2D. Une phase d'annotation manuelle des données 3D est nécessaire mais cela permet ensuite de reconnaître les actions avec une acquisition vidéo 2D. Une modélisation par chaînes de Markov et le classifieur de maximum a posteriori (MAP) sont utilisés pour la reconnaissance d'actions.

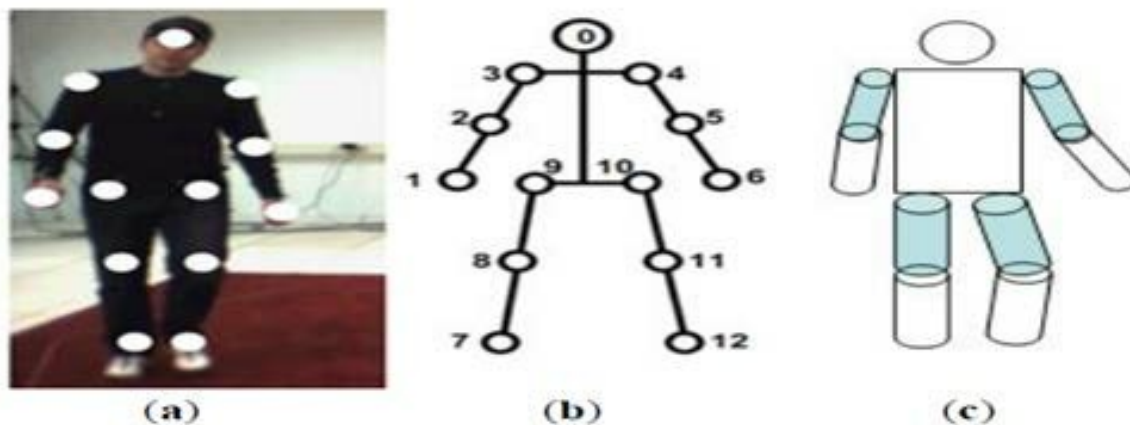


Figure 1.2: Modèles du corps humain (a) l'image originale de corps, (b) modèle cinématique basé sur 13 articulations, (c) le modèle 3D [29].

### 1.1.1.2 Approches holistiques

Les approches holistiques utilisent la structure globale et la dynamique du corps pour représenter le comportement humain. L'idée clé est que, étant donné une région d'intérêt anthropocentrique, la dynamique globale est suffisamment discriminante pour représenter le comportement humain. Les représentations globales sont beaucoup plus grandes par rapport aux méthodes qui utilisent explicitement des modèles corporels ou des informations sur les parties du corps, car elles ne modélisent que le mouvement et l'apparence générale du corps humain. Par conséquent, en général, ces primitives sont calculées de manière plus robuste et efficace. Ces méthodes peuvent être divisées en deux sous-catégories : les méthodes basées sur les contours et les méthodes basées sur le flux optique ou le gradient [42].

- **Méthodes basées sur la silhouette:**

L'une des utilisations les plus anciennes de la silhouette remonte à Yamato et al [38]. Leur méthode se concentre sur la classification des mouvements de tennis basée sur des techniques de quantification vectorielle de séquences d'images binaires. Cette quantification est réalisée en divisant la région d'intérêt correspondant au contour en une grille, où chaque cellule est représentée par le rapport du nombre de pixels noirs et blancs. Ce vecteur primitif est utilisé comme entrée du modèle de Markov à l'état caché, Hidden Markov Models (HMM).

En outre, Bobick et Davis [21] ont proposé le concept de « Motifs Spatio-temporels » (Spatio-Temporal Template). Les auteurs procèdent à l'extraction des silhouettes à partir d'une seule vue et, ensuite, ils regroupent les différences entre les images d'une séquence représentant une action. Un volume spatio-temporel est ainsi construit constituant l'Image d'Énergie du Mouvement (MEI : Motion Energy Image) et aussi l'Image d'Historique du Mouvement (MHI : Motion History Image) dont l'intensité des pixels représente une trace des mouvements de la silhouette (figure 1.3). Cette modélisation simple permet de reconnaître des actions comme le fait de s'asseoir, agiter les bras ou s'accroupir en temps réel.

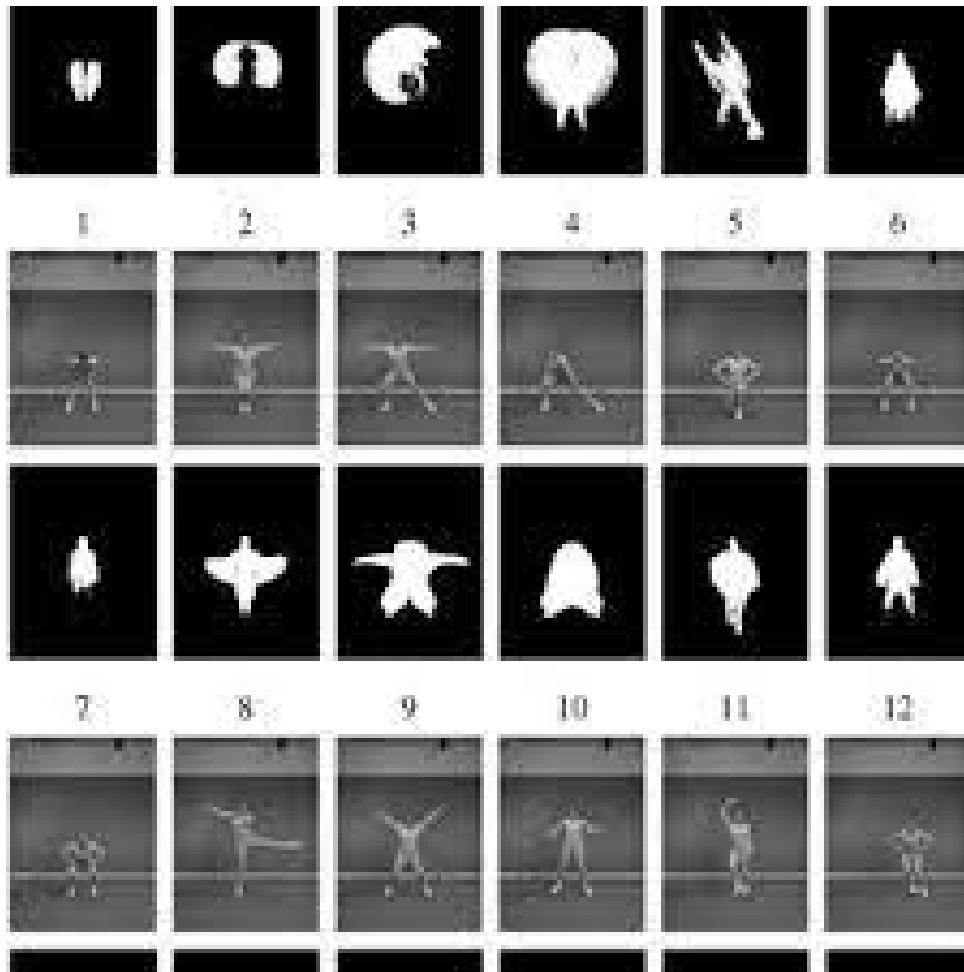


Figure 1.3: Illustration de modèle holistique et volume temporel[21].

- **Méthodes basées sur le calcul de flux optique:**

La seconde catégorie des approches holistiques utilise le flux optique calculé à partir d'images consécutives. Cette catégorie ne nécessite pas de soustraction de fonds qui est généralement une étape difficile. Parmi les travaux qui ont utilisé le flux optique, Efros et al. [37] ont proposé de diviser le flux optique en quatre champs scalaires différents (correspondant à la composante négative et positive horizontale et verticale de flux), et de les utiliser séparément pour l'appariement. Cette représentation a

également été utilisée dans [44] calculent des masques de forme en s'appuyant sur la détection des zones contenant un mouvement significatif. Ensuite, un descripteur de contexte est calculé dans des régions constantes de mouvement en utilisant une grille polaire [44]. Chaque cellule de la grille est décrite avec un histogramme sur les descripteurs SIFTs quantifiés. Le descripteur final pour une séquence vidéo correspond à la somme des descripteurs des différentes cellules. La classification est effectuée en utilisant un SVM ainsi que des modèles pour l'Analyse Sémantique Latente Probabiliste(ASLP).

- **Méthodes basées sur le de gradients orientés:**

L'histogramme de gradients orientés (HOG) est une méthode d'extraction des caractéristiques utilisée en vision par ordinateur pour la détection d'objet,La technique consisté à calcule des histogrammes locaux de l'orientation du gradient sur une grille dense. La méthode est particulièrement efficace pour la détection de personnes et d'objets .

Dalal, Navneet and Triggs, Bill[24] ont proposé une méthode de suivi et de reconnaissance des actions basée sur le descripteur histogrammes de gradients orientés (HOG). Ce descripteur est calculé globalement pour chacune des trames. L'ACP est ensuite appliquée pour réduire la dimension du descripteur. Un HMM est utilisé pour modéliser les actions telles que la course, le patinage, etc. Thureau et Hlavac [34] . ont étendu le descripteur HOG. En effet, au lieu de calculer un seul histogramme par trame, les auteurs divisent chaque trame en des grilles qui se chevauchent. Un descripteur HOG est calculé dans chacune des cellules. La reconnaissance des actions se fait par la méthode du plus proche voisin.

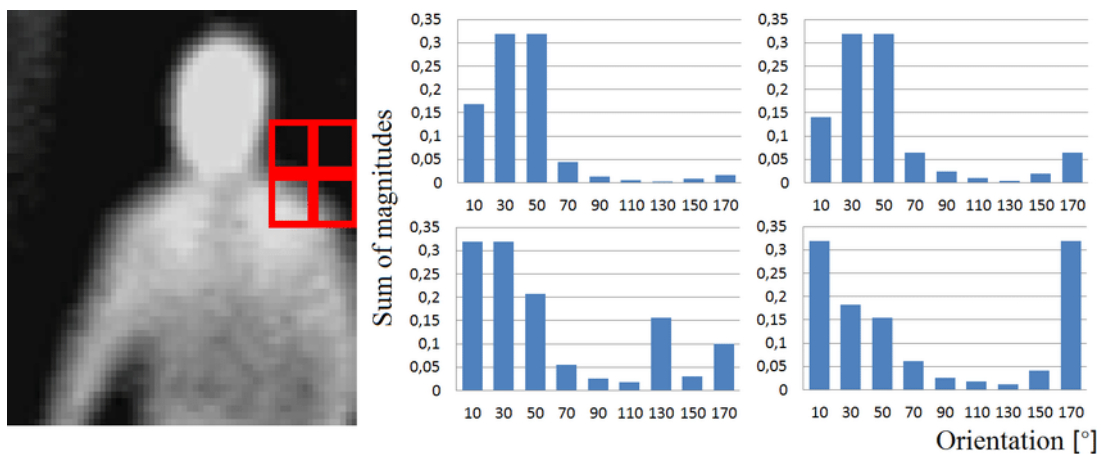


Figure 1.4: histogramme de gradients orientés[24].

### 1.1.2 Approches basées sur des caractéristiques locales :

Les méthodes qui entrent dans cette catégorie décrivent les observations comme un ensemble de descripteurs locaux ou de correctifs. Aucune étape de prétraitement telle

que la suppression de l'arrière-plan ou le suivi des bords n'est requise. Ainsi, cela évite la propagation des erreurs générées par les étapes de prétraitement, en particulier pour les scènes réelles avec des arrière-plans dynamiques. Ces primitives sont généralement robustes et invariantes aux changements d'angle, à l'apparence humaine et à l'occlusion partielle. Les premières études utilisant cette représentation pour reconnaître les actions humaines ont été motivées par le succès des POI dans la reconnaissance d'objets. Ces points correspondent à l'ensemble des pixels présentant une singularité au niveau du gradient ou du contour[42].

- **Détecteurs de points d'intérêt spatio-temporels:**

Les points d'intérêt spatiotemporels (STIP) sont particulièrement intéressants car ce sont des caractéristiques de bas niveau simples et robustes qui permettent une bonne caractérisation des objets en mouvement.elle utulse pour détecter des objets en mouvement et pour caractériser les changements spécifiques dans les mouvements de ces objets. Parmi les premiers travaux proposés pour extraire les points d'intérêts spatio-temporels (STIPs) figure celui de Laptev et Lindeberg [33]. Le auteur Harris ont étendu le détecteur de coins en lui rajoutant la dimension temporelle [26] .

Ce détecteur spatio-temporel, communément appelé Harris 3D, leur permet d'extraire des motifs de mouvement. Ces points d'intérêt spatio temporels correspondent aux points dont le voisinage local est soumis à une variation spatiale et temporelle significative. L'échelle spatiale et temporelle du voisinage est automatiquement sélectionnée. Ce processe a été amélioré par Laptev et al [32] pour compenser les mouvements relatifs aux caméras.

- **Descripteurs locaux**

Pour caractériser les points d'intérêt, des descripteurs locaux sont calculés par rapport au volume spatio-temporel extrait autour d'un point d'intérêt.Il s'agit de capturer l'information portée par un pixel dans l'espace spatio-temporel. La dimension spatiale et temporelle du patch est généralement déterminée par l'échelle du point d'intérêt. Laptev et al [32] ont,été les pionniers de ce type de description pour les séquences vidéo. Les auteurs ont proposé les histogrammes de flux optique (HOF), les HOGs. Les auteurs divisent une région cuboïde de taille  $M \times M \times M$  en une grille de cellules. Pour chaque cellule, on calcule des histogrammes de 4 types d'orientations du gradient (HOG) et des histogrammes de 5 types d'orientations de flux optique (HOF). Dans les gradients calculés pour chaque pixel dans une région cuboïde sont concaténés en un seul vecteur. On utilise ensuite l'ACP pour projeter les vecteurs sur un espace de dimension plus faible. Kläser et al [30] ont étendu HOG au domaine 3D. Les auteurs utilisent une représentation de vidéo intégrale, extension de l'idée d'image intégrale en 3D, pour calculer de façon rapide et efficace les gradients 3D à plusieurs échelles spatiales et temporelles. Des polyèdres réguliers sont ensuite utilisés pour quantifier les orientations 3D du gradient de façon uniforme.

## 1.2 Analyse des mouvements humain Détection et Suivi

L'analyse du mouvement humain suscite un intérêt croissant et des progrès significatifs ont été réalisés ces dernières années, en particulier dans les problèmes de vision de haut niveau tels que l'activité humaine et la compréhension du comportement. Nous nous concentrerons sur un aperçu général de l'ensemble du processus du système d'analyse du mouvement humain. Trois tâches principales de ce processus (détection humaine, suivi humain et reconnaissance de mouvement humain) seront examinées en particulier. La plupart des travaux antérieurs en analyse du mouvement humain ont été réalisés dans le cadre du suivi détection [43].

### 1.2.1 La Détection

les systèmes d'analyse du mouvement humain en générale commencent par la détection de l'homme. La détection vise à segmenter les zones correspondant à des personnes du reste de l'image. C'est une question importante dans un système d'analyse de mouvement humain puisque les processus subséquents telles que le suivi et la reconnaissance d'action sont grandement dépendants de la qualité de la détection. Ce processus implique généralement la segmentation de mouvement[20].

#### 1.2.1.1 Segmentation de mouvement

La segmentation de mouvement dans des séquences vidéo est représente un problème important et difficile qui vise à détecter les régions correspondant aux objets mobiles tels que les véhicules et les personnes dans les scènes naturelles. La détection des blobs mobile fournit le centre d'attention pour les processus ultérieurs tels que le suivi et l'analyse d'activité parce que seuls les pixels changeant doivent être considérées. Cependant, les changements de la météo, l'éclairage, les ombres et les mouvements répétitifs d'un fond encombré rendent la segmentation de mouvement difficile à traiter rapidement et de manière fiable [20].

#### 1.2.1.2 Soustraction de fond

La soustraction de fond est une méthode particulièrement populaire pour la segmentation de mouvement, en particulier dans des [28]situations avec un fond relativement statique. Elle tente de détecter les zones en mouvement dans une image par différence entre l'image courante et une image de fond de référence dans un mode pixel par pixel. Toutefois, cette technique est extrêmement sensible aux dynamiques de changements de scènes en raison d'événements extérieurs et d'éclairage.

Plusieurs recherches se sont intéressées à la construction de différents modèles d'arrière-plan adaptatif afin de réduire l'influence des changements dynamiques de scène sur la segmentation du mouvement. Par exemple, certaines études préliminaires données par

Karman et Brandt et Kilger [28], on proposé un modèle d'arrière-plan adaptatif basé sur le filtrage de Kalman pour s'adapter aux changements temporels de la météo et de l'éclairage.

### 1.2.1.3 Méthodes statistiques

les méthodes statistiques permet extraire les régions de changements de l'arrière-plan se sont inspirées des méthodes de base de soustraction de fond (modèle d'arrière-plan adaptatif). Les approches statistiques utilisent les caractéristiques des pixels individuels ou de groupes de pixels pour construire des modèles d'arrière-plan plus avancés, les statistiques des arrière-plans peuvent être mises à jour dynamiquement au cours du traitement. Chaque pixel de l'image en cours peut être classé en premier plan ou d'arrière-plan en comparant les statistiques du modèle de base actuel. Cette approche est de plus en plus populaire en raison de sa robustesse au bruit, à l'ombre, au changement des conditions d'éclairage, etc[20].

Une étude récente de Haritaoglu et al [25] présente la construction du modèle statistique adaptatif par un représentant de chaque pixel avec trois valeurs: les valeurs d'intensité minimale et maximale, et la différence d'intensité maximale entre les images consécutives observées au cours de la période de formation. Les paramètres du modèle sont mis à jour périodiquement.

### 1.2.1.4 Différence inter-images

Le problème peut se définir simplement de la manière suivante : à partir d'une séquence d'images capturée par une caméra fixe, nous voulons détecter les objets ne faisant pas partie de la scène. La solution peut se décrire naïvement : la détection s'effectue par soustraction d'une image statique de la scène à l'image courante.

La différence d'images est une technique relativement rapide utilisée pour récupérer les objets en mouvement dans une scène. Cette méthode est étudiée depuis la fin des années 1970 . Les images consécutives sont soustraites les unes aux autres, l'application d'un seuil sur l'image des différences permet ensuite de récupérer les pixels correspondant à des régions en mouvement[43].

### 1.2.1.5 Flot optique

L'utilisation du flux optique procure une information de mouvement pour chaque pixel de l'image. Ainsi, il mesure les vecteurs de déplacement à partir de l'intensité des pixels de deux images consécutives ou temporellement rapprochées. Dans un contexte de détection de mouvement, les pixels inactifs posséderont alors une vitesse nulle contrairement aux pixels appartenant à des objets dynamiques.

Une classification sous forme de regroupement est donc nécessaire afin d'isoler et de localiser les zones représentant du mouvement. Cette technique a notamment été utilisée pour la détection de piétons [43].

## 1.2.2 Le Suivi

est l'un des sous-domaines importants de la reconnaissance de l'activité humaine. Dans un environnement extérieur, le suivi peut être facilement effectué à l'aide du système de positionnement global (GPS), mais le GPS n'est pas applicable dans les environnements intérieurs. Le suivi a de nombreuses utilisations dans diverses applications telles que la réalité augmentée, la détection de l'occupation d'une pièce et la navigation intérieure. En raison de son importance croissante, des travaux importants ont été réalisés dans ce domaine.

Les techniques de suivi peuvent aussi être regroupées selon d'autres critères tels que la dimension de l'espace de suivi (2D vs 3D), l'environnement du suivi (à l'intérieur à l'extérieur), le nombre d'humain suivis (un seul humain, humains multiples, des groupes humains), l'état de la caméra (mouvement vs stationnaire), multiplicité de capteur (monoculaire vs stéréo), etc. Différentes méthodes sont utilisées dans le processus de suivi. Le suivi peut être à base de modèles, de régions, ou de caractéristiques. Ces méthodes sont détaillées ci-dessous [43].

### 1.2.2.1 Suivi à base de modèle

Pour détecter et identifier les différents membres du corps, il faut avoir un modèle géométrique en 2D (avec ou sans modèle explicite de la forme) ou en 3D. Le suivi avec modèle explicite compare les données issues de l'image avec un modèle de l'objet ou de la personne à suivre. Cette méthode requiert le développement d'un modèle 2D ou 3D de la personne selon l'application [43].

Les méthodes basées modèles sont robustes aux occultations car elles possèdent une connaissance a priori d'un modèle de la forme, contrairement aux autres méthodes sans modèle, mais elles demandent un coût de calcul important. L'information structurelle du modèle de la forme sert à mettre en correspondance les données image avec le modèle, soit par une approche ascendante associant des hypothèses images, soit par une approche descendante où on cherche le modèle ayant le maximum de corrélation avec les données image. Mais elles demandent un coût de calcul important[43].

### 1.2.2.2 Suivi basé régions

L'approche par région se caractérise par l'extraction dans l'image courante de régions dénommées « blobs », ensemble de pixels connexes et regroupés en fonction d'un critère déterminé, par exemple les pixels dont la valeur est différente avec ceux de l'image précédente, et le suivi des régions homogènes au cours de la séquence. Cette méthode est basée sur la variation du mouvement dans les régions de l'image. Elle ne résout pas les occultations entre objets. L'hypothèse est faite qu'à l'intérieur d'une région, l'apparence est invariante et le mouvement est homogène, par exemple le suivi de « blobs » par filtrage de Kalman [43].

### 1.2.2.3 Suivi basé caractéristiques

Dans les méthodes de suivi de caractéristiques, l'objectif est de détecter des descripteurs liés à des points particuliers, et décrivant l'objet par un ensemble d'attributs géométriques (points, segments, courbes paramétriques, arêtes, contours), ou des régions de l'image. Ces méthodes ont l'avantage d'une bonne robustesse aux occultations car des associations qui n'ont pas pu se faire sur certains points de l'objet cachés dans l'image, ne mettent pas en échec le suivi sur l'ensemble des points [43].

## 1.3 Estimation de mouvement

est un processus qui étudie le mouvement des objets dans une séquence vidéo en trouvant des corrélations entre deux images consécutives pour prédire les changements de position du contenu. Le mouvement est un problème mal posé en vidéo car il décrit un contexte 3D, alors qu'une image est une projection d'une scène 3D sur un plan 2D.

En général, il est représenté par un vecteur de mouvement qui décrit une transformation d'une image en deux dimensions vers une autre. Les vecteurs de mouvement lient deux blocs par projection des blocs sur une même image, c'est-à-dire que les coordonnées du vecteur ne sont définies que spatialement sans contraintes temporelles. Les blocs en question sont différents selon l'algorithme : toute l'image peut être liée au vecteur comme c'est le cas pour l'estimation de mouvement global, ou juste des parties spécifiques de l'image, tels que des blocs rectangulaires, des formes arbitraires ou même par pixel. Les vecteurs peuvent être représentés par un modèle traditionnel ou par de nombreux autres modèles qui peuvent se rapprocher du mouvement d'une caméra vidéo réelle, tel que la rotation et la translation dans les trois dimensions sans oublier le zoom. Cette technique de prédiction est surtout utilisée en compression vidéo, en robotique et en traitement d'image. Il existe plusieurs méthodes d'estimation de mouvement [5] :

### 1.3.1 Sans modèle

Cette méthode inclut dans les méthodes qui n'ont pas de modèle a priori clair. Une tendance récente pour surmonter les limites du suivi de longues séquences a été la recherche directe sur la reconnaissance de pose basée sur des images. Deux approches qui entrent dans cette catégorie d'estimation de pose sans modèle ont été étudiées : les sous-ensembles probabilistes, où les parties individuelles du corps sont d'abord détectées puis assemblées pour estimer la pose 2D ; Méthodes basées sur des exemples qui apprennent à mapper directement de l'espace image 2D à l'espace modèle 3D [5].

### 1.3.2 Assemblage probabiliste de parties

L'assemblage probabiliste de pièces a été appliqué à l'estimation de pose 2D ascendante directe, en commençant par identifier les emplacements probables des parties du corps humain, puis en assemblant ces positions pour obtenir la configuration qui correspond le mieux aux observations. Un avantage potentiel de la détection et du suivi est que la pose peut être estimée indépendamment à chaque image, permettant une estimation de la pose pour un mouvement rapide [20].

### 1.3.3 Méthodes basées exemple

Un certain nombre de méthodes basées sur l'exemple pour l'estimation de pose humaine ont été proposé, elles comparent l'image observée avec une base de données d'échantillons. Brand [28] Utilisez un modèle de Markov caché (HMM) pour représenter le mappage entre les séquences de silhouettes 2D dans l'espace image et le mouvement du squelette dans l'espace de pose 3D . Dans ce travail, nous avons appris à cartographier certaines séquences de mouvement à l'aide d'images de silhouette d'un modèle humanoïde. HMM a été utilisé pour estimer la séquence la plus probable de poses 3D en fonction de la séquence de silhouettes 2D observées à un point de vue donné. Les approches basées sur des cas représentent une cartographie entre l'image et la pose, fournissant un mécanisme efficace pour l'estimation directe de la pose 3D. En règle générale, ces approches utilisent des captures de mouvement données pour fournir des exemples de formation utilisant des poses 3D connues. Une limitation courante des approches basées sur des modèles est la limitation des postures ou des mouvements utilisés dans l'apprentissage. Élargir le vocabulaire du mouvement peut introduire des ambiguïtés dans la mapping.

### 1.3.4 Utilisation indirecte de modèles

Dans cette classe, les méthodes utilisent un modèle a priori dans l'estimation de pose comme une référence ou table de consultation pour guider l'interprétation des données mesurées.

Un certain nombre de chercheurs ont étudié la reconstruction directe de la forme et des

mouvements du modèle à partir d'une coque visuelle (visual hull) [22] sans modèle a priori. Mikic et coll [35] présente un système intégré de récupération automatique d'un modèle du corps humain et du mouvement à partir de plusieurs vues d'une séquence d'images. L'acquisition de modèle est basée sur une approche hiérarchique basée sur des règles de localisation et d'étiquetage de parties de corps. La connaissance préalable de la forme des parties du corps, la taille relative, et la configuration sont utilisées pour segmenter la coque visuelle. Un filtre de KALMAN étendu est ensuite utilisé pour la reconstruction du mouvement humain entre les images.

### 1.3.5 Utilisation directe de modèles

Cette classe utilise explicitement une représentation géométrique 3D de forme humaine et une structure cinématique pour reconstituer la pose. La majorité des approches utilisent une méthode d'analyse par synthèse pour optimiser la similarité entre le mode de projection et l'image observé. Les paragraphes ci-dessous énumèrent quelques-uns des plus courants [5].

- le flux optique
- Block-Matching

#### 1.3.5.1 le flux optique

La technique qui se rapproche le plus de l'estimation de mouvement se nomme le flux optique, dont les vecteurs correspondent au mouvement perçu au niveau des pixels. En estimation de mouvement, la correspondance exacte de la position des pixels dans chaque image n'est pas une nécessité

#### 1.3.5.2 Block-Matching

L'algorithme de Block-Matching est une méthode de corrélation cherchant le bloc le plus ressemblant dans l'image de référence au bloc courant. L'image courante est divisée en blocs non chevauchés de tailles identiques qui sont vus comme des blocs indépendants où les pixels composant chaque bloc ont la même allure de mouvement. L'application est généralement limitée dans une fenêtre de recherche qui se dessine autour du bloc courant, pour des raisons de performances matérielles. Afin de déterminer le meilleur des blocs similaires, l'algorithme compare la différence entre le bloc source et les autres blocs à l'aide d'un critère de qualité tel que la SAD et le bloc ciblé qui minimise cette règle est retenu.

## 1.4 Collecte de données

Avant d'alimenter un système HAR, une collecte de données est réalisée en suivant un protocole bien particulier défini comme suivant

### 1.4.1 Qualité des données

- Taille: Nombre des échantillons (enregistrements). Certaines tâches nécessitent une grande taille de données pour qu'elles soient apprises [23] .
- Le nombre et le type de caractéristiques (nominales, binaires, ordinales ou continues).
- Le nombre des erreurs d'annotation.
- La quantité de bruits dans les données: erreurs et exceptions.

### 1.4.2 quantité de données

pour analyser la quantité de données, il existe deux types : données quantitatives continues et les données quantitatives discrètes [12] .

1. Les données quantitatives continues se sont les données lorsqu'elle prend un nombre infini de valeurs réelles à l'intérieur d'un intervalle donné.
2. Les données quantitatives discrètes en études statistiques, on désigne par données quantitatives discrètes des données qui ne peuvent prendre qu'un nombre fini de valeurs réelles possibles au sein d'un intervalle donné.

### 1.4.3 Intégration des données

Quand on veut collecter des données pour l'apprentissage automatique, souvent, on a besoin de combiner des données de différentes sources [2] .

- Données structurées .
  - Bases de données .
  - Fichiers de tableurs: CSV, etc.
- Données semi-structurées: XML, JSON, etc.
- Données non structurées: documents textes, images, métadonnées, etc.

Pour vérifier l'intégrité des données Il faut, tout d'abord :

- Vérifier que les fichiers XML sont conformes à leurs définitions XSD .
- Vérifier que les séparateurs des colonnes dans les fichiers CSV sont correctes (point-virgule ou virgule et pas les deux au même temps).

lorsque on joint deux schémas de données, on doit vérifier:

1. Problème de nommage: il se peut qu'on ait des données identiques avec des nominations différentes. Par exemple, si on veut joindre deux tables de données b1 et b2 qui ont deux attributs avec le même sens mais différents noms b1.numclient et b2.clientid, on doit unifier les noms des attributs[2].

2. Conflits de valeurs: les valeurs des attributs provenant de sources différentes sont représentées différemment. Par exemple, une source de données qui représente la taille en cm et une autre qui représente la taille en pouces.
3. Redondance: les attributs qu'on puisse déduire des autres, les enregistrements identiques.
4. Types différents des attributs.

#### 1.4.4 Annotation (Étiquetage) des données

L'annotation des données est la plus importante tâche dans l'apprentissage automatique. Si les données sont mal annotées, la performance de notre système d'apprentissage va diminuer . ImageNet (une base des images) a pris 9 années pour être annotée manuellement, avec un nombre de 14 millions images [2] il existe plusieurs type de annotation de données comme :

- Classification des images ,segmentation , reconnaissance et détection d'objets
- Annotation audio
- Annotation vidéo
- Annotation textuelle:annotation ,intention annotation ,annotation des sentiments

### 1.5 Pré-traitement de données

est une technique d'exploration de données qui consiste à transformer des données brutes en un format compréhensible. Les données du monde réel sont souvent incomplètes, incohérentes et/ou manquent de certains comportements ou tendances, et sont susceptibles de contenir de nombreuses erreurs. Le prétraitement des données est une méthode éprouvée pour résoudre ces problèmes. Lorsque nous parlons de données, nous pensons généralement à de grands ensembles de données avec un grand nombre de lignes et de colonnes. Bien que ce soit un scénario probable, ce n'est pas toujours le cas – les données peuvent se présenter sous de nombreuses formes différentes : tableaux structurés, images, fichiers audio, vidéos, etc.

Apprentissage automatique les algorithmes ne fonctionnent pas si bien avec le traitement des données brutes. Avant de pouvoir transmettre ces données à un algorithme ML, nous devons les prétraiter. En d'autres termes, nous devons lui appliquer des transformations. Avec le prétraitement des données, nous convertissons les données brutes en un ensemble de données propre [17] .

### 1.5.1 Techniques Prétraitement des données d'image

L'analyse d'images regroupe plusieurs disciplines que l'on classe en deux catégories. Les processus de bas-niveaux, qui nécessitent très peu d'informations sur le contenu des images. Il s'agit ici des processus de filtrage, d'amélioration et de restauration d'images, processus que nous regroupons sous le terme traitement d'images, ainsi que d'extraction d'indices. Les processus de haut-niveaux, qui fonctionnent en aval de ceux de bas-niveaux, et qui peuvent nécessiter des informations sur le contenu des images. Il s'agit de la reconstruction tri-dimensionnelle, la reconnaissance de formes, les processus cognitifs de façon générale il existe un certain nombre d'étapes de pré-traitement que nous pourrions souhaiter effectuer avant de l'utiliser dans un projet d'apprentissage en profondeur. Les paragraphes ci-dessous énumèrent quelques-uns des plus courants [17].

1. **Format d'image uniforme** l'une des premières étapes consiste à assurer que les images ont la même taille et le même rapport d'aspect. La plupart des modèles de réseaux neuronaux supposent une image d'entrée de forme carrée, ce qui signifie que chaque image doit être vérifiée si elle est carrée ou non, et recadrée de manière appropriée. Le recadrage peut être effectué pour sélectionner une partie carrée de l'image, comme indiqué. Lors du recadrage, nous nous soucions généralement de la partie au centre [17].
2. **Mise à l'échelle de l'image** une fois que nous nous sommes assurés que toutes les images sont carrées (ou ont un rapport d'aspect prédéterminé), il est temps de mettre à l'échelle chaque image de manière appropriée. Nous avons décidé d'avoir des images avec une largeur et une hauteur de 100 pixels. Il existe une grande variété de techniques de mise à l'échelle et de réduction d'échelle et nous utilisons généralement une fonction de bibliothèque pour le faire pour nous [17].
3. **Normalisation des entrées d'image** la normalisation des données est une étape importante qui garantit que chaque paramètre d'entrée (pixel, dans ce cas) a une distribution de données similaire. Cela rend la convergence plus rapide lors de la formation du réseau. La normalisation des données est effectuée en soustrayant la moyenne de chaque pixel, puis en divisant le résultat par l'écart type. La distribution de ces données ressemblerait à une courbe gaussienne centrée sur zéro. Pour les entrées d'image, nous avons besoin que les nombres de pixels soient positifs, nous pouvons donc choisir de mettre à l'échelle les données normalisées dans la plage [0..1] ou [0.. 255] [17].
4. **Réduction de la dimensionnalité**  
Nous pourrions choisir de réduire les canaux RVB en un seul canal d'échelle de gris. Il y a souvent des considérations pour réduire d'autres dimensions, lorsque la performance du réseau neuronal est autorisée à être invariante à cette dimension, ou pour rendre le problème de formation plus traitable [17].

5. **Augmentation des données** une autre technique de prétraitement courante consiste à augmenter l'ensemble de données existant avec des versions perturbées des images existantes. La mise à l'échelle, les rotations et autres transformations affines sont typiques. Ceci est fait pour exposer le réseau neuronal à une grande variété de variations. Cela rend moins probable que le réseau neuronal reconnaisse des caractéristiques indésirables dans l'ensemble de données [17].

## 1.6 Intelligence artificielle

La définition générale de l'intelligence artificielle est un domaine qui combine le calcul et de puissants ensembles de données pour résoudre des problèmes. Il comprend également les sous-domaines de l'apprentissage automatique et de l'apprentissage en profondeur, qui sont souvent mentionnés [41]

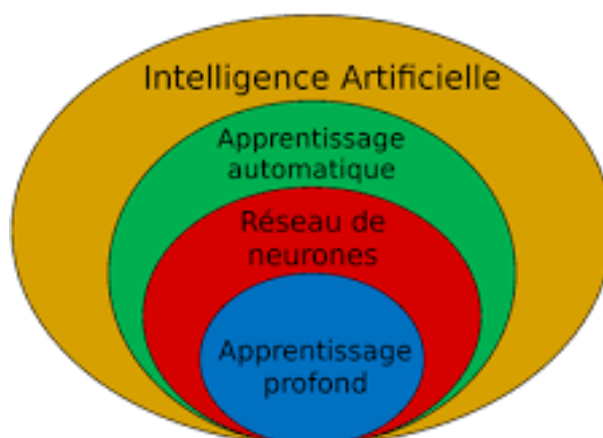


Figure 1.5: Schéma de décomposition du domaine de l'intelligence artificielle et de ces sous-domaines[41].

### 1.6.1 Apprentissage automatique

L'intelligence artificielle est devenue un sujet brûlant dans les médias et les magazines scientifiques en raison de nombreuses réalisations, dont beaucoup sont le résultat d'avancées dans le domaine de l'apprentissage automatique (ML). Une caractéristique souhaitable de l'IA est sa capacité à rationaliser et à prendre des mesures qui sont les plus susceptibles d'atteindre un objectif spécifique. Un sous-ensemble de l'intelligence artificielle est l'apprentissage automatique, qui fait référence au concept selon lequel les programmes informatiques peuvent automatiquement apprendre de l'expérience et acquérir des compétences sans intervention humaine [8].

### 1.6.1.1 Fonctionnement de l'apprentissage automatique

l'apprentissage automatique se compose de différents types de modèles apprentissage automatique, utilisant diverses techniques algorithmiques. Selon la nature des données et le résultat souhaité, l'un des quatre modèles d'apprentissage peut être utilisé : supervisé, non supervisé, semi-supervisé ou renforcé. Dans chacun de ces modèles, une ou plusieurs techniques algorithmiques peuvent être appliquées en fonction de l'ensemble de données utilisé et du résultat souhaité. Ces algorithmes peuvent être utilisés un par un ou en combinaison pour obtenir des données complexes et Des enjeux imprévisibles [13].

### 1.6.1.2 Les méthodes d'apprentissage automatique

Deux des méthodes d'apprentissage automatique les plus populaires sont apprentissage supervisé ,apprentissage semi-supervisé et non supervisé. a apprentissage supervisé l'apprentissage supervisé est considéré comme la classe la plus élémentaire d'algorithmes d'apprentissage automatique [4].

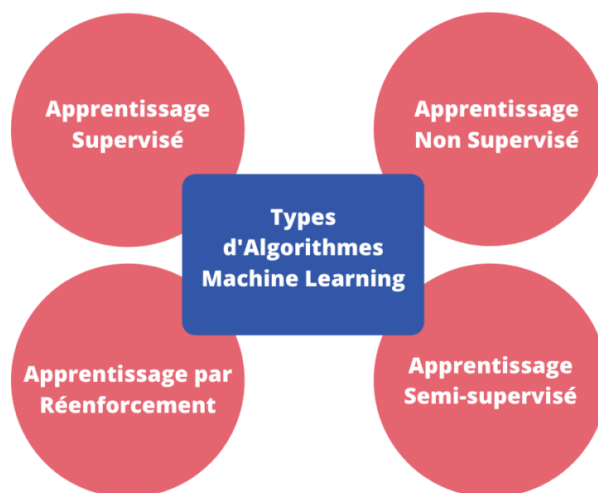


Figure 1.6: Les différents types d'apprentissage automatique[45].

1. **Apprentissage supervisé** Défini comme l'utilisation d'ensembles de données étiquetés pour former des algorithmes capables de classer les données ou de prédire avec précision les résultats. Au fur et à mesure que les données d'entrée sont introduites dans le modèle, le modèle ajuste ses poids jusqu'à ce que le modèle corresponde [4].
2. **Apprentissage non-supervisé** l'apprentissage non supervisé est un type d'algorithme de Machine Learning utilisé pour définir des inférences à partir de jeux de données sans intervention humaine, contrairement à l'apprentissage supervisé où des étiquettes sont fournies avec les données.

La méthode d'apprentissage non supervisé la plus courante est l'analyse de clusters, qui applique des méthodes de clustering pour explorer des données et y trouver des

patterns ou des groupements cachés.relations [4].

3. **Apprentissage semi-supervisé** consiste à entraîner un modèle d'apprentissage sur un jeu de données partiellement annoté qui comporte quelques données annotées et beaucoup de données non annotées. L'idée est d'attribuer les annotations en utilisant la similarité entre les données annotées et les données non annotées.

L'apprentissage semi-supervisé se situe ainsi entre l'apprentissage supervisé qui n'utilise que des données annotées et l'apprentissage non supervisé qui n'emploie que des données non annotées. La combinaison de ces deux ensembles de données permet d'améliorer sensiblement les résultats sans avoir recours à l'intervention fastidieuse (coûteuse et chronophage) de l'annotation manuelle [45].

4. **Apprentissage par renforcement** Dans le cadre de l'apprentissage par renforcement, l'agent interagit avec son environnement en effectuant des actions pour recevoir des récompenses. La récompense peut être positive, négative ou nulle. Prenons par exemple, un robot qui joue au jeu de Tetris, quand il gagne il y a une récompense positive, et dans le cas contraire une récompense négative [45].

## 1.6.2 Apprentissage profond

est l'une des technologies principales du Machine learning. Avec lepprentissage profond nous parlons d'algorithmes capables de mimer les actions du cerveau humain grâce à des réseaux de neurones artificielles. Les réseaux sont composés de dizaines voire de centaines de «couches» de neurones, chacune recevant et interprétant les informations de la couche précédente [4].

### 1.6.2.1 Le fonctionnement de apprentissage profond

Les réseaux de neurones sont des couches de nœuds, tout comme le cerveau humain est composé de neurones, les nœuds au sein des couches individuelles sont connectés aux couches adjacentes. On dit que le réseau est plus profond en fonction du nombre de couches dont il dispose, un seul neurone du cerveau humain reçoit des milliers de signaux d'autres neurones. Dans un réseau neuronal artificiel, les signaux voyagent entre les nœuds et attribuent des poids correspondants, un nœud pondéré plus lourd exercera plus d'effet sur la couche de nœuds sui- vante. La couche finale compile les entrées pondérées pour produire une sortie. Les systèmes d'apprentissage en profondeur nécessitent un matériel puissant car ils traitent une grande quantité de données et impliquent plusieurs calculs mathématiques complexes, cepen- dant, même avec un matériel aussi avancé, les calculs de formation en apprentissage profond peuvent prendre des semaines. Les systèmes d'apprentissage en profondeur nécessitent de grandes quantités de données pour renvoyer des résultats précis , en conséquence, les informations sont alimentées sous la forme d'énormes ensembles de données. Lors du traitement des données, les réseaux de

neurones artificiels sont capables de classer les données avec les réponses reçues à partir d'une série de questions binaires vraies ou fausses impliquant des calculs mathématiques très complexes. Par exemple, un programme de reconnaissance faciale fonctionne en apprenant à détecter et à reconnaître les arêtes et les lignes des visages, puis les parties les plus significatives des visages et, enfin, les représentations globales des visages. Au fil du temps, le programme s'entraîne et la probabilité de réponses correctes augmente. Dans ce cas, le programme de reconnaissance faciale identifiera avec précision les visages avec le temps[4].

### 1.6.2.2 Les de modèles utilisant l' apprentissage profond

- Réseaux neuronaux convolutifs (CNN) :
- Réseaux neuronaux récurrents (RNN)
- Réseaux de mémoire à long et court terme (LSTM)
- Réseaux de fonction de base radiale (RBFN)
- Réseaux adversariaux génératifs (GAN)
- Machines de Boltzmann restreintes (RBM)

## 1.7 La prière

Dans l'islam, la prière exprime la foi à travers la communication personnelle avec Allah. La prière peut s'effectuer dans l'intimité ou en public. Chaque musulman doit effectuer quotidiennement cinq prières en direction de la Mecque, la Fara'idh. tourné vers la qibla, qui est la direction de la Kaaba de La Mecque. La prière est constituée de rakaat (unité de prière) en fonction de la prière effectuée (Fajr, Dhuhr, 'Asr, Maghrib ou encore 'Isha) ce nombre varie de 2 à 4 rakaat [6].

N°	Nom du Prière	NNbre des raka3ats	temps
1	Fajr	2unités (Rak'ah)	à l'aube
2	Dhuhr	4unités (Rak'ah)	l'après-midi
3	asr	4unités (Rak'ah)	au milieu de l'après-midi
4	Maghrib	3unités (Rak'ah)	coucher du soleil
5	isha	4unités (Rak'ah)	la nuit

Tableau 1.1: Les cinq prière

## 1.8 Correction prière

pour correcte la prière on fait Sudjud Al-Sahow,il s'agit de deux prostrations(sujud) qui ont pour le but de corriger les distraction qui ont eu lieu pendant la prière, elles sont effectuées avant ou après les salutation, il est fait pour dissimuler des défauts dans

l'exécution de la prière dus à l'oubli, il y a trois raisons pour sujud sahwi : à cause de l'ajout , à cause du manque et à cause du doute sont defini comme suit :

1. **Sudjud Al-Sahow à cause de L'ajout** Quiconque oublie dans sa prière et ensuite s'incline, ou se prosterne, et ainsi de suite, alors il doit se prosterner sahwi après les salutation .
2. **Sudjud Al-Sahow à cause la diminution** Quiconque, donc, oublie, quitte alors l'une des circoncisions ab'adh, alors il doit se prosterner avant le salam. Par exemple, un makmum oublie le tahiyat initial et ne s'en souvient qu'après avoir terminé le quiam , il n'a donc pas besoin de s'asseoir et il lui suffit de se prosterner Sahow avant les salutation . Cependant, s'il se souvient avant la moitié de s'être levé pour le quiam, alors il peut s'asseoir et terminer le tahiyat initial et il n'est pas nécessaire de se prosterner sahwi.
3. **Sudjud Al-Sahow à cause du doute** Le doute dans la prière est dû au fait de ne pas croire si c'est plus ou moins, par exemple quelqu'un doute s'il est dans le troisième ou le quatrième rakat. Il existe deux types de doute :
  - Si une personne est plus encline à une chose (illusion, ou plus de 50 % sûre), par exemple, elle est plus confiante qu'elle est maintenant dans le troisième rakat et non dans le quatrième rakat, alors elle devrait obéir et adopter l'attitude de ce qui il est plus convaincu, alors il exécute la prosternation sahwi après le salut
  - Si une personne est dans le doute entre deux choses et n'est pas inclinée vers l'une d'elles (dzan, ou seulement 50-50 sûr), alors elle devrait prendre position sur la chose qui est certaine de sa vérité, qui est un petit nombre de rak'a. Couvrez ensuite le manque, puis inclinez-vous avant le salam.

## 1.9 Travaux récents sur la reconnaissance de l'activité prière

Dans le but de proposer une système de vérifier la séquence de salat bien défini , nous allons étudier quelque des méthodes existantes de reconnaissance de l'activité de la salat dans le but d'exploiter leur rechercher dans notre projet .

1. **Anis Koubaa ,Ammar Adel [31]** : ont proposé une methode de développer une intelligence artificielle cadre qui guide les fidèles pour évaluer l'exactitude des postures de leurs prières. L'objectif de cette système est s'attaque au problème de la reconnaissance des gestes de base de la prière islamique en utilisant Réseaux de neurones convolutifs (CNN). Ils ont construit un ensemble de données pour les différent positions de base de Salat puis utiliser un réseau de neurones YOLOv3 pour la reconnaissance de gestes.les résultats obtenus montrent que la moyenne a

précision moyenne atteint 85% pour un ensemble de données d'entraînement de 764 images des différentes postures .

2. **Nafees Ahmad, Lansheng [18]**: ont proposé un modèle de reconnaissance des activités de Salah (SARM) pour La maladie d'Alzheimer est une maladie. Le modèle (SARM) utilisant un capteur mobile est proposé dans le but de reconnaître des activités spécifiques, telles que Al-Qayam (debout), Ruku (debout pour s'incliner) et Sujud (debout jusqu'à la prosternation). Ce modèle implique la collecte de données, la sélection et le placement de capteur, prétraitement des données, segmentation, extraction de caractéristiques et classification. Ils ont proposé fournira un tremplin pour développer une application pour observer la prière et ces activités reconnaissance, des ensembles de données collectés on été utilisés pour obtenir de meilleurs résultats. Des expériences approfondies ont été réalisées pour tester et valider les caractéristiques du modèle pour former la forêt aléatoire (RF), le Kplus proche voisin (KNN), le Bayes naïf (NB) et l'arbre de décision (DT). La précision moyenne prévue de RF, KNN, NB et DT était respectivement de de 97% , 94% , 71,6% et 95%.
3. **Syed Hamza Mohiuddin, Tahir Syed [36]** : ont été présentés une solution basée sur l'apprentissage en profondeur pour la reconnaissance d'activité Salat à l'aide de smartphones pour huit Postures de Salat. Ils ont montré qu'une profondeur considérablement plus petite le modèle d'apprentissage est plus performant que celle proposée dans des travaux similaires pour le problème à main. Ceci est particulièrement utile pour les smartphones car ils ont mémoire et puissance de traitement limitées ,De plus ils ont également montré comment une combinaison de modèle basé sur l'apprentissage en profondeur et de DFA peut être utilisé pour surveiller l'état de la Salat . La solution basée qui guide la personne à travers tous les gestes et pose dans le bon ordre via un retour visuel et audio. ils formuler le problème comme une reconnaissance d'action humaine en utilisant reconnaissance de pose à l'aide de réseaux de neurones convolutifs intégrés dans un automate fini déterministe de transitions de pose. Avec cette approche qui utilise des images vidéo au lieu de volumes, et donc avec un modèle, ils obtiennent de meilleurs résultats précision du test sur le jeu de données de référence, soit 83% contre 78% de le concurrent le plus proche.
4. **Reem Al-Ghannam et Hmood Al-Dossari [19]** : Ils introduit une application mobile pour surveiller et reconnaître de petites activités de prière c'est-à-dire se tenir debout, s'incliner, se prosterner et assis en utilisant les fonctions d'accélération du téléphone mobile pour miner l'exactitude de la prière (c'est-à-dire l'exhaustivité et ordre des activités). Les données de l'accéléromètre ont été collecter pour six prières, totalisant 118 échantillons, représentant quatre principales activités de prière(quiam,ruku,sujud ,julus) . Les données recueillies ont été utilisées

former et tester des algorithmes d'apprentissage automatique supervisés pour extraire et reconnaître les activités de prière. Dans leur projet ils montrer que les étapes de la prière peuvent être extraites et reconnues avec précision à l'aide d'algorithmes d'apprentissage automatique. Le WEKA La boîte à outils d'apprentissage automatique a été utilisée pour tester les classificateurs à l'aide de la caractéristiques extraites des données de l'accéléromètre. Trois différent classificateurs ont été testés : Naive Bayes, IB1 Algorithm et les arbres de décision J48 et leur précision dépassait 90%.

5. **Md Mozasser Rahman, Rayan Abbas Ahmed** Alharazi [39] : a proposé un système intelligent de surveillance et de formation Salat basé sur la vision artificielle et le traitement d'image. objectif de système est d'apprendre les postures correctes dans la salat, par lequel ils analysent les livres de hadiths et d'érudits musulmans dans la section revue de la littérature. Le deuxième objectif est de développer un algorithme de traitement d'images sous MATLAB, ce qui est également réalisé puisque le résultat montre d'assez bonnes performances. Le troisième objectif de cette étude est de tester les performances de la salat et de fournir une rétroaction à l'utilisateur. Ceci est également réalisé lorsque le résultat de sortie de l'image correspondante affiche le message sur la performance de la salat et forme l'utilisateur en donnant les instructions correctes concernant les postures actuelles de la salat. Les résultats sont assez précis, car la méthode proposée est capable d'identifier et de faire correspondre le motif pour reconnaître jusqu'à 90% et informer l'utilisateur de sa performance de salat.

## Conclusion

Dans ce chapitre nous avons bien défini les Approches proposées pour la reconnaissance des actions et aperçu sur les tâches principales d'un système d'analyse du mouvement humain qui sont : la détection, le suivi, l'estimation de pose comme nous avons identifier les méthode de collecter les donnée et les processus réaliser tel que la segmentation et pre-traitement pour aide de classifier les action et definition intelligence artificiel et les type de l'apprentissage et definition d'activity de prière on islam finalement on Listent quelques travaux récents sur la reconnaissance de l'activité prière.

# Conception du système

## Introduction

Dans ce chapitre, nous allons voir en détail les différents modules du système de vérification. Ce dernier commence par le choix de la prière afin de déterminer la séquence de référence. Par la suite, génère une autre séquence à partir de la séquence de la prière. Finalement, en comparant les deux séquences peut donner des remarques de corrections (au cours de la prière) ainsi qu'un rapport (Voir le schéma ci-dessous) sur la prière à la fin.

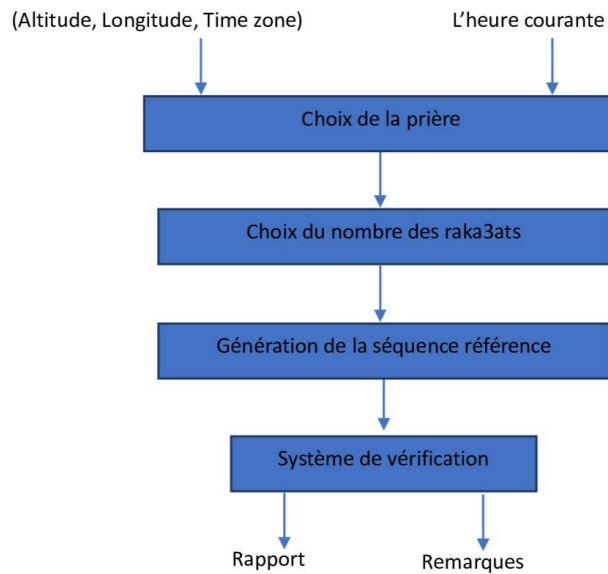


Figure 2.1: Schéma général du système de vérification le séquence de la prière.

## 2.1 Choix de la prière

Le choix peut être réalisé selon trois façons, savoir :

- L'utilisateur fait le choix de la prière

- Le choix de la prière peut être faite en utilisant le calendrier de la ministère des affaires religieuses (voir les figures 2.2 et 2.3). Malheureusement, il faut construire ce fichier XML chaque année suivant le calendrier pour-que le système soit à jour.

```
<?xml version="1.0" encoding="utf-8"?>
<Calender>
  <Month Name="Shawal">
    <!-- One week -->
    <Day Name="Monday" Number="10" SolarDate="01/05/2023">
      <Salat Name="El-Fadjr">
        4:25
      <Salat/>
      <Salat Name="El-Dhohr">
        12:45
      <Salat/>
      <Salat Name="El-Asn">
        16:28
      <Salat/>
      <Salat Name="El-Maghrib">
        19:35
      <Salat/>
      <Salat Name="El-Ishaa">
        21:00
      <Salat/>
    <Day/>
    <Day Name="Tuesday" Number="11" SolarDate="02/05/2023">
      <Salat Name="El-Fadjr">
        4:23
      <Salat/>
      <Salat Name="El-Dhohr">
        12:45
      <Salat/>
      <Salat Name="El-Asn">
        16:28
      <Salat/>
    <Day/>
  </Month>
</Calender>
```

Figure 2.2: Exemple du fichier XML.

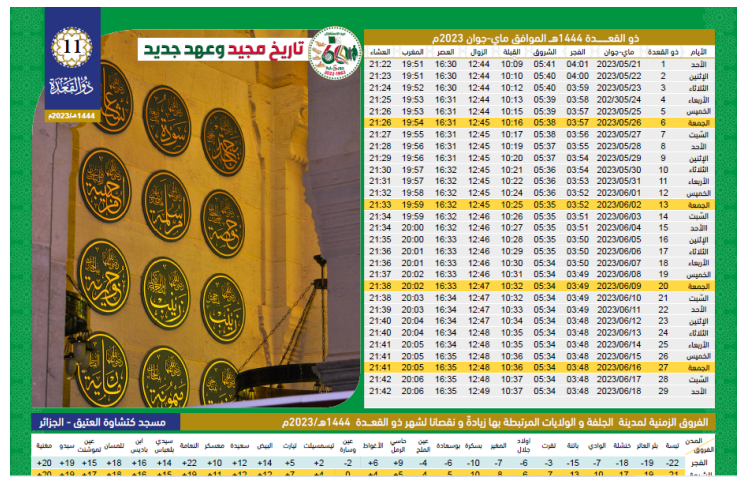


Figure 2.3: Illustration du calendrier de la ministère des affaires religieuses.

- En utilisant un algorithme qui calcule le temps de la prière en se basant sur la position géographique (Altitude, Latitude, Time Zone). ce algorithme permet d'estimer la prière courante en utilisant les trois paramètres suivantes: Longitude, Altitude et le TimeZone. Dans ce travail nous avons utiliser l' algorithme développé par S. Shafi et H. Zarrabi-Zadeh (voir le site suivant: <http://praytimes.org/manual>).

## 2.2 calcul du nombre des raka3ats

Après avoir connaître la prière courante ou choisissant une selon le cas voulut, maintenant, il est possible de calculer le nombre de raka3ats (2, 3 ou 4). Ce dernier, nous permet de par la suite de générer une liste de séquence référence.

```
from praytimes import *
import time
def verifyCurrentRaka3at (currentRaka3at ,oneRaka3at , currentRaka3atIndex ):
    i = 0
    print ('Tester Raka3at Num < ,currentRaka3atIndex +1,' >')
    for salat_pose in oneRaka3at:
        if (salat_pose != currentRaka3at [i]):
            print ('la posture correcte est ', salat_pose , ' mais vous av
            i = i+1
start = time.time ();
currentSalat = getSalatByTime ("21:00")
print ('currentSalat = '+currentSalat)
numofRaka3ats = getNumberOfRaka3ats (currentSalat)
print ('numOfRaka3ats = ', numofRaka3ats)
```

## 2.3 Génération la séquence de référence

La génération de la séquence de référence consiste à produire une liste contenant les étiquettes des différentes poses clés de la prière, à savoir: quiam, ruku, sujud et julus. La liste obtenue va être comparée avec une autre liste ( ReferenceSequence) générée durant les étapes de prédiction afin de déterminer si la prière est réalisée correctement ou non, le code de génération cette list est comme suivant:

```
from clasiffication import *
ReferenceSequence = []
ReferenceSequence.append ('takbir ')
for i in range (numOfRaka3ats):
    ReferenceSequence.append ('quiam ')
    ReferenceSequence.append ('ruku ')
    ReferenceSequence.append ('quiam ')
    ReferenceSequence.append ('sujud ')
    ReferenceSequence.append ('julus ')
    ReferenceSequence.append ('sujud ')
ReferenceSequence.append ('julus ')
print ("les sequence de posture de salat :",ReferenceSequence)
```

## 2.4 Système de vérification

Dans cette section nous décrivons les étapes de du système de vérification. Ce dernier utilise des points d'intérêts extrait à partir de chaque frame de la séquence vidéo choisit. Par la suite utilise ces points afin déterminer la classe de la pose courante.

Le système passe par de deux phases principales, à savoir: la phase d'apprentissage et la phase de prédiction (voir la figure (2.6)).

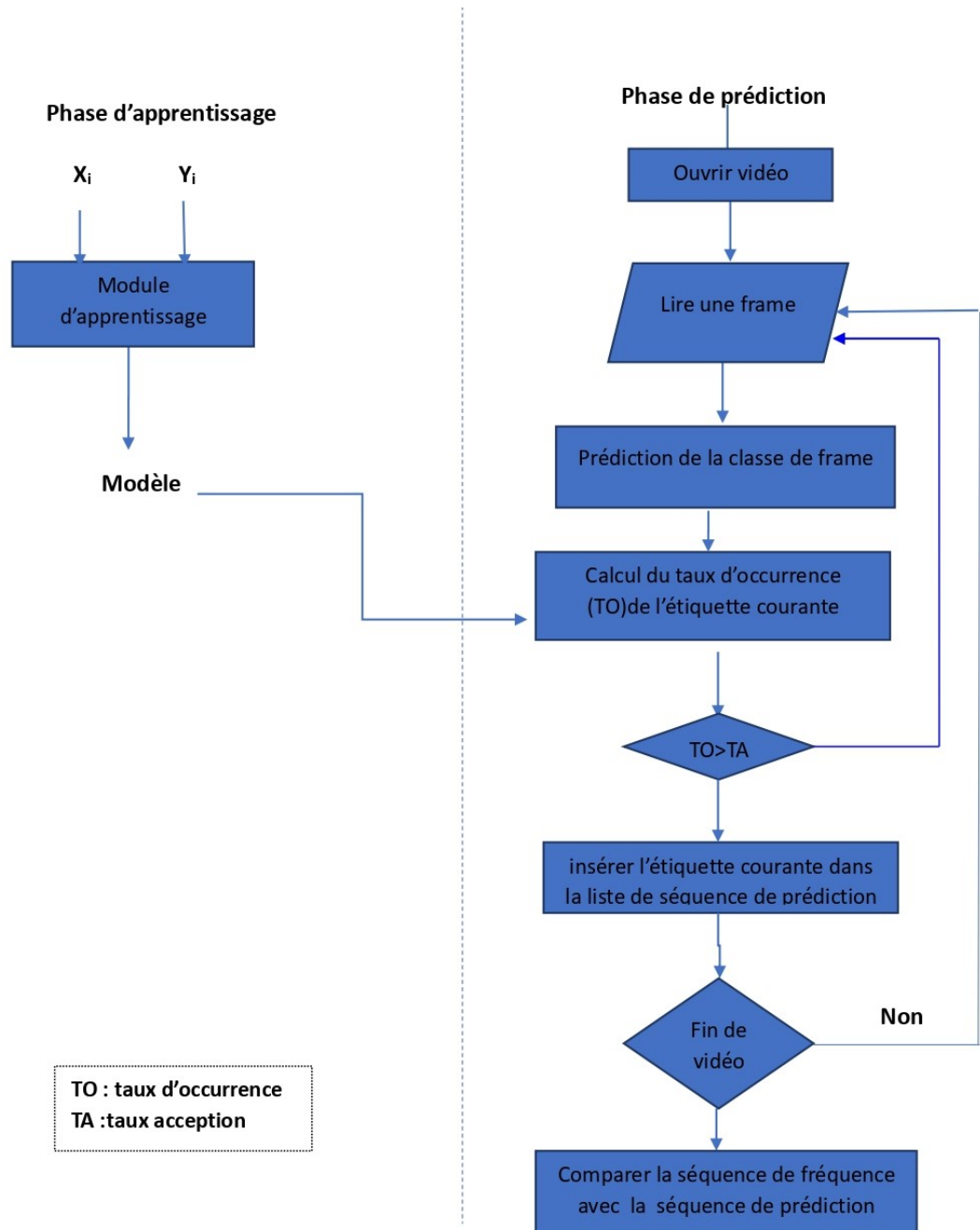


Figure 2.4: L'architecture du système de vérification

## 2.4.1 Extraction des points d'intérêts à l'aide de MediaPipe

MediaPipe est un cadre pour la construction de pipelines d'apprentissage automatique pour le traitement de données de séries chronologiques telles que la vidéo, l'audio, etc. Ce cadre multiplateforme fonctionne sur les ordinateurs de bureau/serveurs, Android, iOS ,il propose plusieurs solutions comme :

- Détection faciale
- Maillage du visage
- Iris
- Mains
- Pose
- Holistique
- Détection d'objet
- Suivi des boîtes

### 2.4.1.1 MediaPipe Pose

est une solution de suivi de pose corporelle haute fidélité qui restitue 33 points de repère 3D et un masque de segmentation d'arrière-plan sur l'ensemble du corps à partir d'images RVB (Notez l'image RVB ). [31]. La figure ci-dessous illustre ces points.

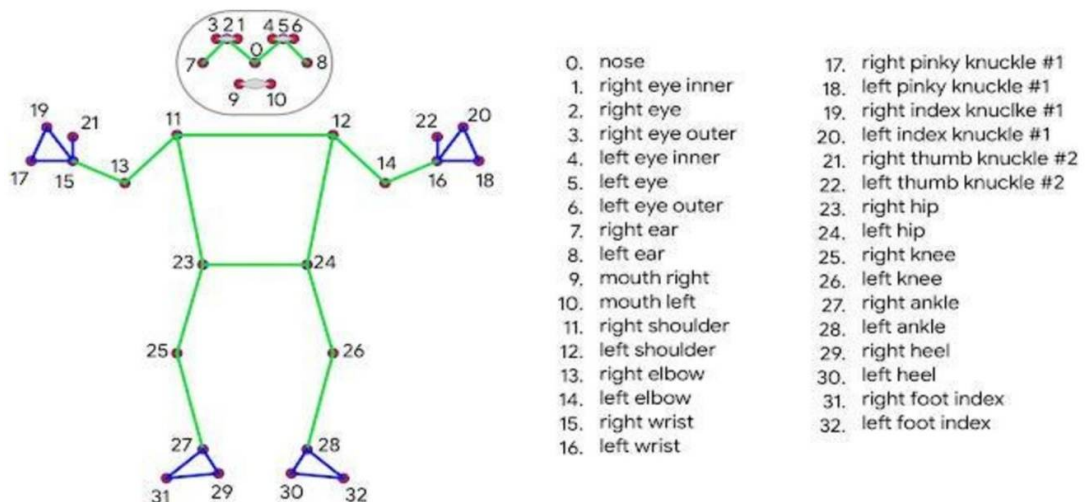


Figure 2.5: Extraction des points d'intérêts à l'aide de MediaPipe

## 2.5 Extraction des points d'intérêts à l'aide YOLO

### 2.5.1 YOLO

(You Only Look Once) est une méthodologie, ainsi qu'une famille de modèles conçus pour la détection d'objets. Depuis la création en 2015, YOLOv1, YOLOv2 et YOLOv3 utilise l'apprentissage en profondeur a continué avec des avancées open source au fil des années.

### 2.5.2 YOLOv7

est une architecture de détection d'objets et il fournit de nouvelles têtes de modèle qui peuvent générer des points clés (squelettes) et effectuer une segmentation d'instance en plus de la régression par boîte englobante, ce qui n'était pas standard avec les modèles YOLO précédents.

## 2.6 Comparaison entre mediapipe et YOLOv7

Caractéristique	YOLOv7 Pose	MediaPipe Pose
Topologie	17 Keypoints COCO	33 Keypoints COCO + Blaze Palm + Blaze Face
Flux de travail	Détection lancée avec chaque frame	Détection lancée une fois ensuite suivie par un tracker jusqu'à ce que l'occlusion se produise
GPU	CPU & GPU	seulement CPU
Segmentation	Segmentation non intégrée aux poses directement	Segmentation intégrée
Nombre de personnes	Multi-personne	une person

Tableau 2.1: Comparaison entre mediapipe et YOLOv7.

### 2.6.1 Phase d'apprentissage

cette phase commence par la collection des données afin de créer un modèle pour la prédiction.

#### 2.6.1.1 Collection les données

L'accès à l'ensemble de données disponible pour la reconnaissance les posture prière est limité. Nous avons donc collecté les images de postures de prière auprès des volontaires et à partir d'ensemble d'images de personne et effectuant différentes positions de Salat à l'aide de téléphones mobiles, sous différents angles de vue, spécialement capturés à cet effet, complétés par des images (et des cadres vidéo) de personnes en prière recueillies sur Interne , comme nous avons utiliser la les donnes collecter par autre chercheur, les

quatre(04) postures de salat différentes comprennent sont : (Julus) ,(Quiam), (Ruku), (Sujud) , Le tableau 01 indique le nombre d'images collectées pour chaque posture

N°fichier	Lable de posture	Nombre de donnes collecter
0	Julus	657
1	quiam	1050
2	ruku	725
3	sujud	678

Tableau 2.2: Nombre d'images collectées pour chaque posture

### 2.6.1.2 Queleque exemples de posture de prière



Figure 2.6: Posture julus



Figure 2.7: Posture quiam



Figure 2.8: Posture ruku



Figure 2.9: Posture sujud

Après la collection des images de chaque posture on stoker dans fichier 'data ', pour renforcer la base de données ,nous avons donc collecté les images de chaque postures de prière auprès des volontaires et et à partir d'ensemble d'images d'étudiants et de membres de laboratoire effectuant différentes positions de prière à l'aide de téléphones mobiles complétés par des images (et des cadres vidéo) de personnes en prière recueillies

sur internet le code python qui permet de collection les données à partir de video comme suivant :

```
import os
import cv2

DATA_DIR = './data'
if not os.path.exists(DATA_DIR):
    os.makedirs(DATA_DIR)
number_of_classes = 4
dataset_size = 100
cameraNo = 1
cap = cv2.VideoCapture(cameraNo)
address = "http://192.168.1.100:8080/video"
cap.open(address)
for j in range(number_of_classes):
    if not os.path.exists(os.path.join(DATA_DIR, str(j))):
        os.makedirs(os.path.join(DATA_DIR, str(j)))
    print('Collecting data for class {}'.format(j))

done = False
while True:
    ret, frame = cap.read()
    cv2.putText(frame, 'Ready? Press "Q" ! :)', (100, 50),
                cv2.IMSHOW('frame', frame)
    if cv2.waitKey(1) == ord('q'):
        break
counter = 0
while counter < dataset_size:
    ret, frame = cap.read()
    cv2.imshow('frame', frame)
    cv2.waitKey(1)
    cv2.imwrite(os.path.join(DATA_DIR, str(counter), frame)
    counter += 1

cap.release()
cv2.destroyAllWindows()
```

### 2.6.1.3 Création des données

Dans cette section, nous allons créer le jeu de données en utilisant la bibliothèque mediapipe. Nous allons prendre des images qui contiennent les différentes postures de prière, le fichier **'data'**, en extraire les points clés et les stocker dans un fichier pickle. Les étapes de création se déroulent comme suit :

- Tout d'abord nous avons importé les bibliothèques nécessaires qui aideront à créer l'ensemble de données.
- Ensuite, dans les quatre premières lignes, nous importons les modules nécessaires pour extraire les points clés et leurs utilitaires de dessin.
- Dans le code ci-dessus, nous créons une liste temporaire **'data\_aux'** et nous parcourons les images de pose individuellement, en extrayant les points clés à l'aide du modèle de mediapipe\_pose (pose Blaze) les coordonnées respectives x, y pour chacun des 33 points de repère puis stockant dans un tableau temporaire **'data\_aux'**. Une fois l'itération terminée, nous ajoutons cette liste temporaire en tant que nouvel enregistrement dans notre ensemble de données **'data'**.
- Finalement nous pouvons simplement enregistrer ces données dans un fichier pickle **'datapickle'** que nous utiliserons pour créer un modèle d'apprentissage automatique. Le code source de cette opération est comme suit :

```
import os
import pickle
import numpy as np
import mediapipe as mp
import cv2

mp_pose = mp.solutions.pose
mp_draw = mp.solutions.drawing_utils
mp_drawing_styles = mp.solutions.drawing_styles
pose = mp_pose.Pose()
DATA_DIR = './data'
data = []
labels = []
for dir_ in os.listdir(DATA_DIR):
    for img_path in os.listdir(os.path.join(DATA_DIR, dir_)):
        data_aux = []
        x_ = []
        y_ = []
        img = cv2.imread(os.path.join(DATA_DIR, dir_, img_path))
        img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
```

```

results = pose.process(img_rgb)
mp_draw.draw_landmarks()
mp_draw.DrawingSpec((255, 0, 0), 2, 2)
if results.pose_landmarks:
    h, w, c = img.shape
    op_img = np.zeros([h, w, c])
    op_img.fill(0)
    mp_draw.draw_landmarks(op_img, results.pose_landmarks)
    for id, lm in enumerate(results.pose_landmarks.landmark):
        x = int(lm.x * w)
        y = int(lm.y * h)
        x_.append(x)
        y_.append(y)
        data_aux.append(x - min(x_))
        data_aux.append(y - min(y_))
        data.append(data_aux)
        labels.append(dir_)

f = open('data.pickle', 'wb')
pickle.dump({'data': data, 'labels': labels}, f)
f.close()

```

#### 2.6.1.4 Exemple d'extraction des points d'intérêts à l'aide de MediaPipe

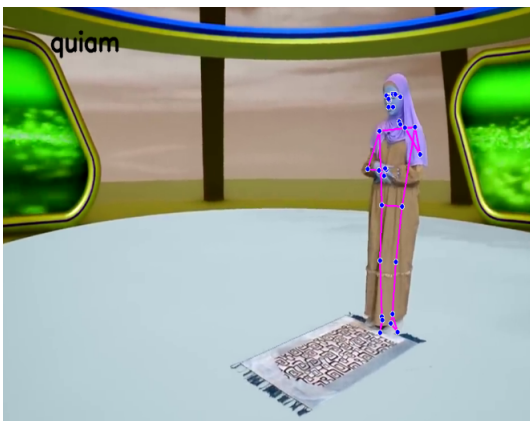


Figure 2.10: Les points clés pour chaque repère de posture quiam

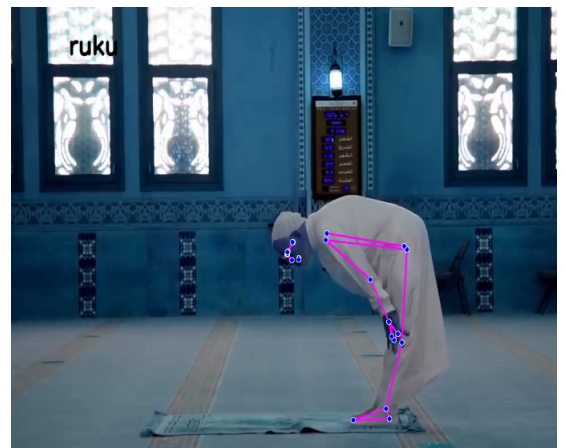


Figure 2.11: Les points clés pour chaque repère de posture ruku

### 2.6.1.5 Création du modèle de détection de pose

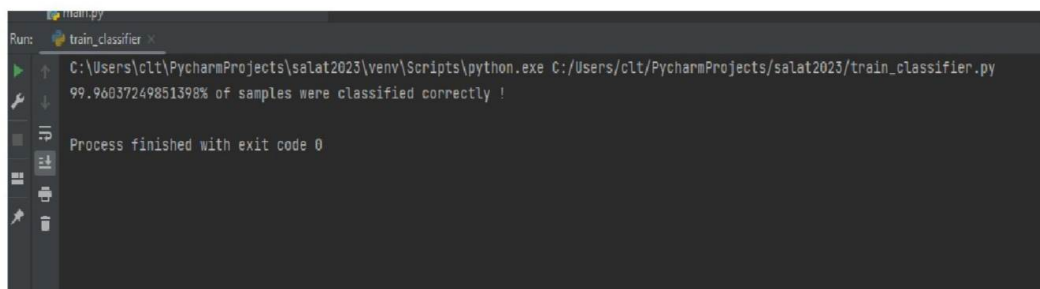
Nous avons utiliser le algorithme de la forêt aléatoire (random forest classifier) ,cette processus passe par les étapes suivantes:

- Importer la bibliothèque
- divisez les données entières en deux ensembles test (un ensemble pour tester les données) et train (un ensemble pour entraîner les données) pour diviser les données, nous pouvons utiliser bibliothèque sklearn, en utilisant la méthode `train_test_split()`,Les données de train représentent 80% de l'ensemble de données et le test représente 20% de l'ensemble de données.
- Créez un jeu d'évaluation et passez-le à la méthode `fit`.
- Enfin, nous prédisons la sortie pour cela, nous utilisons une autre partie des données que nous obtenons de `test_train_split`, c'est-à-dire les données de test. Nous allons l'utiliser et prédire la sortie.

```
y_pred = model.predict(X_test)
print(y_pred)
```

### 2.6.1.6 Evaluation des performances du modèle proposer

Ce modèle a un accuracy score de précision de 99.96% sur les données de ,Cela semble assez impressionnant. Mais on rappelle que la précision n'est pas une bonne mesure des



```
Run: train_classifier
C:\Users\clt\PycharmProjects\salat2023\venv\Scripts\python.exe C:\Users\clt\PycharmProjects\salat2023/train_classifier.py
99.96037249851398% of samples were classified correctly !
Process finished with exit code 0
```

Figure 2.12: Accuracy score du model .

performances du classificateur lorsque les classes sont déséquilibrées . Nous avons besoin de plus d'informations pour comprendre les performances réelles du modèle. On utilise

### 2.6.1.7 la matrice de confusion et rapport de classement

### 2.6.1.8 Matrice de confusion

est un moyen d'exprimer le nombre de prédictions d'un classificateur qui étaient correctes et, lorsqu'elles sont incorrectes, où le classificateur s'est confondu (d'où le nom !). Dans les matrices de confusion ci-dessous, les lignes représentent les vraies étiquettes et les

colonnes représentent les étiquettes prédites. Les valeurs sur la diagonale représentent le nombre (ou le pourcentage, dans une matrice de confusion normalisée) de fois où l'étiquette prédite correspond à l'étiquette réelle. Les valeurs dans les autres cellules représentent les cas où le classificateur a mal étiqueté une observation ; la colonne nous indique ce que le classificateur a prédit, et la ligne nous indique quelle était la bonne étiquette. C'est un moyen pratique de repérer les zones où le modèle peut avoir besoin d'un peu d'entraînement supplémentaire.

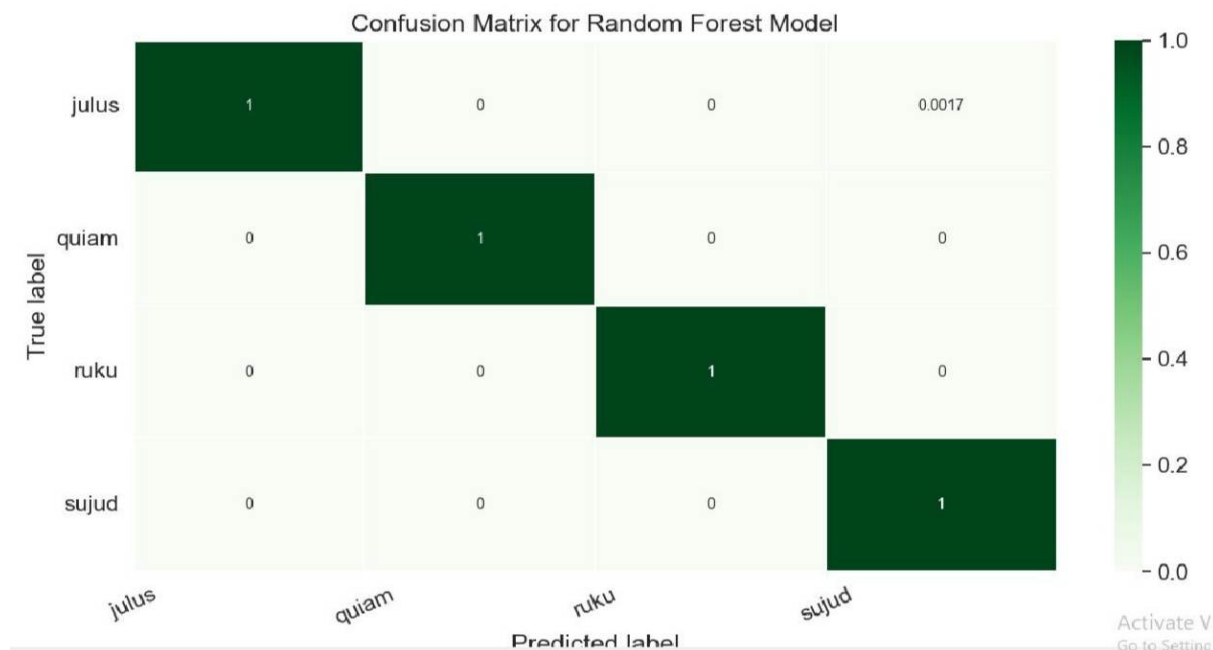


Figure 2.13: La matrice de confusion de model .

### 2.6.1.9 Rapport de classement

pour obtenir encore plus d'informations sur les performances du modèle, nous devons examiner d'autres mesures telles que la précision, le rappel et le score F1.

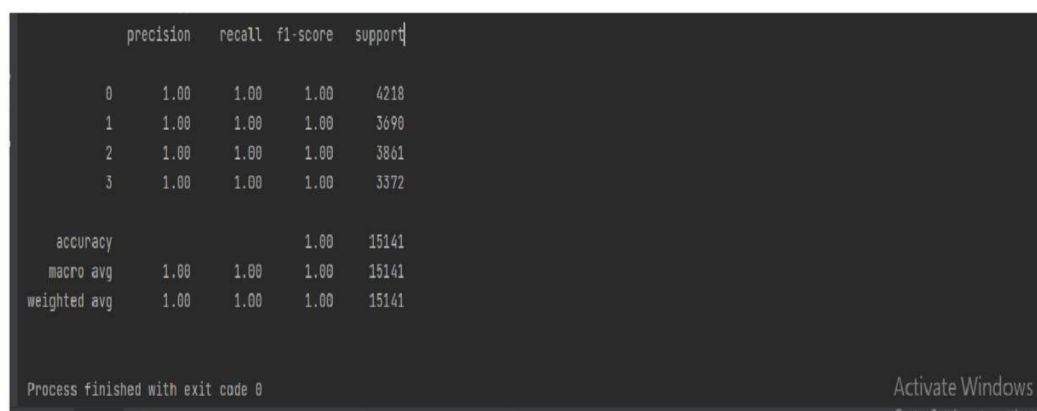


Figure 2.14: Rapport de classement Le score F1 .

## 2.6.2 Algorithmes utilisés

Avant de décider d'utiliser l' algorithme de forêt aléatoire comme un algorithme qu'aide à l'obtention d'un modèle de prédiction dans notre étude, nous avons fait plusieurs tests sur d'autres algorithmes, à savoir:

1. support Vector Machines(SVM) est un modele d'apprentissage automatique supervisé considéré pour la recherche est la machine à vecteurs de support. L'objectif d'une machine à vecteurs de support est de trouver un plan qui classe distinctement les points de données lorsque tracé dans un espace de caractéristiques
2. L'algorithme des k plus proches voisins est un algorithme de classification supervisée qui n'est pas paramétrique et utilise la distance entre les points de données pour prédire la classification d'un point de données individuel.
3. VGG-16 est VGG16 est une architecture CNN (Convolutional Neural Network) qui est largement considérée comme l'un des meilleurs modèles de vision par ordinateur disponibles aujourd'hui. Les concepteurs de ce modèle ont examiné les réseaux et amélioré la profondeur à l'aide d'une architecture de filtre à convolution compacte ( $3 \times 3$ ) qui dépassait considérablement les paramètres de l'art antérieur. Le 16 dans VGG16 fait référence au fait qu'il a 16 couches qui ont du poids. C'est un très grand réseau avec 138 millions de paramètres[3]

dans le tableau ci-dessous, on résume les résultats obtenus:

N°	type de l'apprentissage	nom d' algorithme	Taux de accuracy
01	apprentissage automatique	Forêt aléatoire	99.96 %
02	apprentissage automatique	Support Vector Machines	99.93%
03	apprentissage automatique	k-plus proches voisins (KNN)	99.94%
04	apprentissage profond	VGG-16	82.89 %

Tableau 2.3: Comparision les résultat entre les deffirents algorithmes

## 2.6.3 Phase de prédiction

Puisque l'algorithme Forêt aléatoire donne une accuracy on va utiliser le modèle de l'apprentissage car il a été formé avec succès, le code ci-dessous peut être utilisé pour tester le modèle en affichant divers points clés et leurs prédictions respectives. les poses de salut ont été détectées à partir un vidéo contient un personne exécutant l'une des cinq prières quotidiennes à l'aide de l'objet OpenCV VideoCapture. les points de repère sont extraits à l'aide du modèle mediapipe pose (Blazepose), puis transmis au préformé Classificateur Foret aléatoire , pour faire la prédiction pour l'ensembles d' actions lorsque la salut terminée nous avons obtenu une List nommé: **séquence de prédiction** contient la séquence de pose exécute.

Les figure ci-dessus montre un instantané de détection vidéo de quatre poses différentes ont été effectuées avec précision Les prédictions sont générées et affichées sur les frame.

### 1. Exemples de detection posture de quiam



Figure 2.15: Posture quiam



Figure 2.16: Posture quiam

### 2. Exemples de detection posture de ruku :

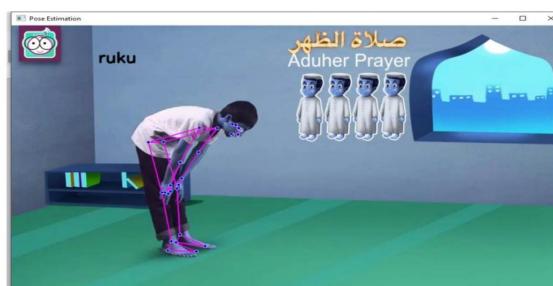


Figure 2.17: Posture de ruku

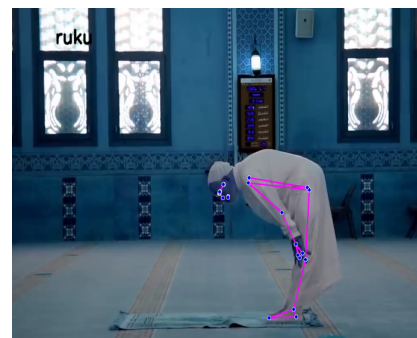


Figure 2.18: Posture de ruku

### 3. Exemples de detection posture de sujud

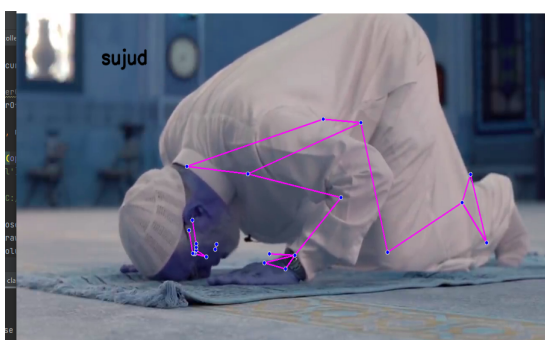


Figure 2.19: Posture de sujud

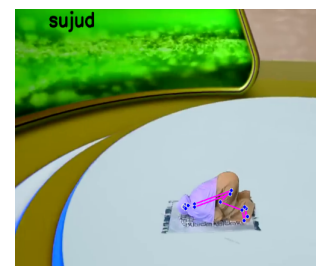


Figure 2.20: Posture de sujud

### 4. Exemples de detection posture de julus:



Figure 2.21: Posture de julus

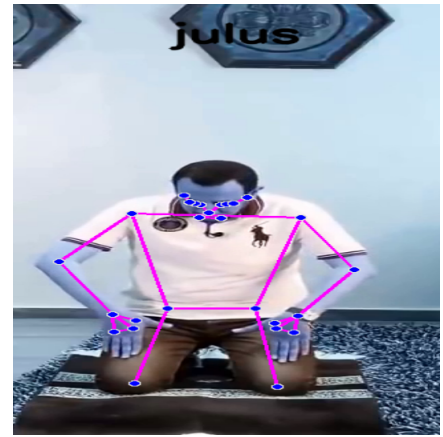


Figure 2.22: Posture de julus

## 2.6.4 Vérifier la correction de prière

dans cette etaps on fait une comparaison entre la list détecte par la model de la classification et la liste 'référenceséquence' qui contient la séquence de posture de tout les raka3ats

- Si les deux list sont identiques alors la salat est: **correct**.
- Sinon alors on donne des remarques sur la salat:
  1. le cas de l'ajout pendent la prière il faut deux prostrations (sujud) après les salutation
  2. le cas de La diminution pendent la prière il faut deux prostrations (sujud) avant les salutation

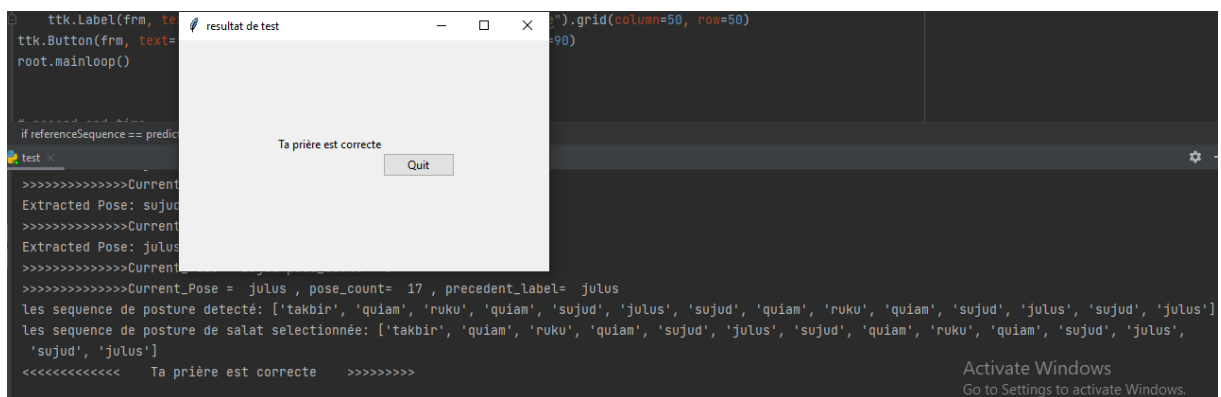


Figure 2.23: Resultat de systeme de verification .

## Conclusion

Dans ce chapitre, nous allons voir en détail les différents modules de conception du système. Ce dernier commence par le choix de la prière afin de déterminer la séquence de référence. Par la suite, génère une autre séquence à partir de la séquence de la prière. Finalement, en comparant les deux séquences peut donner des remarques de corrections (au cours de la prière) ainsi qu'un rapport sur la prière à la fin.

# Tests et Résultats

## Introduction

Nous avons vu dans le chapitre précédent la conception du système de façon détaillée. Dans ce chapitre, nous allons présenter l'environnement de développement avec les différentes bibliothèques importantes utilisées, ainsi que les résultats obtenus en testant le système.

### 3.1 Environnement matériels(Hardware)

**Matériel :** Le matériel réalisé est un PC personnel Lenovo I5 caractérisé par les caractéristiques suivantes :

- La capacité de mémoire 8 GB
- Un processeur Intel Core(TM) i3-4005U CPU @ 1.70GHz
- Windows 10 Pro , service pack 1 ,64 bit type système.

### 3.2 Environnement logiciel (Software)

**PyCharm** est un environnement de développement intégré utilisé pour programmer en Python. Il permet l'analyse de code et contient un débogueur graphique. Il permet également la gestion des tests unitaires, l'intégration de logiciels de gestion de versions, et supporte le développement web . Développé par l'entreprise russe JetBrains, c'est un logiciel multi-plateforme qui fonctionne sous Windows, Mac OS X et Linux. Il est décliné en édition professionnelle, diffusé sous licence propriétaire, et en édition communautaire diffusé sous licence Apache [7] .

### 3.3 Langage de programmation et bibliothèques

**Python** est devenu ces dernières années le langage de programmation le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages [16]. Dans ce travail nous avons utilisé les bibliothèques suivantes:

- **NumPy** nous avons utilisé cette bibliothèque pour adapter les types d'entrée selon la configuration du modèle utilisés, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux. Nous avons utilisé cette bibliothèque exactement dans le cas de balayage de l'image et l'extraction des fenêtres [10].
- **Matplotlib** est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques, nous avons utilisé cette bibliothèque pour visualiser nos images sous formes de graphiques [9].
- **Pandas** est une bibliothèque Python pour l'analyse de données. Lancée par Wes McKinney en 2008 pour répondre au besoin d'un outil d'analyse quantitative puissant et flexible, Pandas est devenue l'une des bibliothèques Python les plus populaires. Elle compte une communauté extrêmement active de contributeurs. La bibliothèque se positionne donc comme une bibliothèque généraliste dans le domaine du traitement d'images, et ne se démarque pas dans un domaine spécifique [11].
- **OpenCV** est une énorme bibliothèque open-source pour la vision par ordinateur, l'apprentissage automatique et le traitement d'images. OpenCV supporte une grande variété de langages de programmation comme Python, C++, Java, etc. Il peut traiter des images et des vidéos pour identifier des objets, des visages ou même l'écriture d'un être humain [1].
- **Os** Le système d'exploitation fait partie des modules utilitaires standard de Python. Ce module fournit un moyen portable d'utiliser les fonctionnalités dépendantes du système d'exploitation.
- **MediaPipe** fournit des modèles d'apprentissage automatique de base pour des tâches courantes comme le pistage des yeux, éliminant ainsi le même goulot d'étranglement de développement qui existe pour une multitude d'applications d'apprentissage automatique. Ces modèles, ainsi que leurs API excessivement faciles à utiliser, rationalisent à leur tour le processus de développement et réduisent la

durée de vie du projet pour de nombreuses applications qui dépendent de la vision par ordinateur .

- **Pickle** est une bibliothèque Python standard pour la sérialisation et la désérialisation d'objets Python vers et depuis des flux de données binaires. Cela vous permet d'enregistrer des structures de données complexes telles que des listes, des dictionnaires et des classes personnalisées dans un fichier ou de les envoyer sur le réseau à un autre programme. C'est une fonctionnalité puissante qui vous permet de sauvegarder l'état d'un programme et de le reprendre plus tard [15].
- **Sklearn (Scikit-learn)** est une bibliothèque en Python qui fournit de nombreux algorithmes d'apprentissage supervisé et non supervisé. Elle s'appuie sur certaines technologies qui vous sont peut-être déjà familières, comme NumPy, pandas et Matplotlib ! Les fonctionnalités offertes par scikit-learn sont les suivantes [14] :
  - La régression, notamment la régression linéaire et logistique.
  - Classification, y compris K-Nearest Neighbors (voisins les plus proches) .
  - Le clustering, y compris K-Means et K-Means++.
  - Sélection de modèles .
  - Prétraitement, y compris la normalisation Min-Max .

### 3.4 Résultats et discussion

Pour évaluer notre système de vérification de la séquence de posture de prière, dix (10) volontaires ont participé à l'expérience. Nous avons demandé aux participants d'enregistrer des vidéos des cinq prières quotidiennes en utilisant des portables avec une résolution acceptable.

Les Tableaux ci-dessus contient les résultats de détection chaque posture :

Tableau 3.1: Taux détection les postures de quiam.

N°raka3at	Rak3a 1	Rak3a 2	Rak3a 3	Rak3a 4
Taux de detection				
Taux de validation (%)	100	100	100	100
Taux de erreur (%)	00	00	00	00

Tableau 3.2: Taux détection les postures de quiam.

N°raka3at	Rak3a 1	Rak3a 2	Rak3a 3	Rak3a 4
Taux de detection				
Taux de validation (%)	100	70	70	70
Taux de erreur (%)	00	30	30	30

Tableau 3.3: Taux détection les Postures de ruku.

	N°raka3at	Rak3a 1	Rak3a 2	Rak3a 3	Rak3a 4
Taux de detection					
Taux de validation (%)		70	70	70	70
Taux de erreur (%)		30	30	30	30

Tableau 3.4: Taux détection les postures de sujud.

	N°raka3at	Rak3a 1	Rak3a 2	Rak3a 3	Rak3a 4
Taux de detection					
Taux de validation (%)		100	100	80	100
Taux de erreur (%)		00	00	20	00

Tableau 3.5: Taux détection les postures de julus.

N°de volontaire	Nom de priere	Num de raka3at	Taux de succès	Observation
01	Isha	04	70 %	If faut vérifier la séquence de prière
02	Asr	04	100 %	prière valide
03	Dhuhr	04	70 %	prière valide
04	Maghrib	03	80 %	If faut vérifier la séquence de prière
05	Maghrib	03	100 %	prière valide
06	Maghrib	03	80 %	prière valide
07	Fajr	02	100 %	prière valide
08	Fajr	02	90 %	If faut vérifier la séquence de prière
09	Fajr	02	100 %	prière valide
10	Fajr	02	100 %	prière valide

Tableau 3.6: Taux détection les postures de julus.

Afin de minimiser les erreurs du système durant la classification ou la prédiction nous avons utilisé la variable TA (Taux d'Acceptation) qui représente le nombre minimale pour une étiquette pour qu'elle soit pris en compte. Malheureusement, dans certain cas surtout si la personne fait la prière de façon très rapide cette configuration peut causée l'effet inverse en obtenant des erreurs de classification. Pour cela, il est nécessaire de rendre la valeur de TA variable (fixée différemment selon chaque personne).

## Conclusion

Dans ce chapitre, Nous avons décrit les étapes de la classification de on utiliser le model obtenu par la model mediapipe pose pour extrait les point cles puis l'algorithme foret pour détection posture de salat. Donc nos objectifs principaux sont atteints par l'obtention de bons résultats de la classification des différentes des posture de salat puis d 'assurer de la validité de la prière ou non .

# Conclusion et perspectives

Dans ce travail, nous avons proposé un système de vérification de la séquence de prière (Salat). Durant sa réalisation nous avons utilisé une dataset qui contient des différents principaux postures (poses) de la prière: quiam, ruku, sujud, julus. En plus, la création du modèle de décision est faite en utilisant quatre classificateur: KNN, SVM, VGG16. Le système utilise la bibliothèque mediapipe afin de créer une représentation avec des points d'intérêts de la personne dans les deux phase: apprentissage et prédiction. par la suite, il fait la classification des contenu des frame vidéo. Les résultats obtenus sont très satisfaisante et encourageante, mais il est possible de faire d'autres améliorations comme suit:

- La position du caméra peut être devant la personne à une distance de 2 mètres afin de donner des des frames de bonne qualité.
- L'ajout d'autres classes comme: takbir, tachahoud, taslim.
- L'ajout d'un estimateur de rythme de prière afin de bien fixé les variables (telle que TA) selon la personne qui utilise l'application.



# Bibliographie

- [1] “About - OpenCV — opencv.org” – <https://opencv.org/about/>, [Accessed 24-May-2023].
- [2] “Chapitre II: Préparation des données — projeduc.github.io” – [https://projeduc.github.io/intro\\_apprentissage\\_automatique/preparation.html](https://projeduc.github.io/intro_apprentissage_automatique/preparation.html), [Accessed 05-Jun-2023].
- [3] “Code Studio — codingninjas.com” – <https://www.codingninjas.com/codestudio/library/vgg-16---cnn-model>, [Accessed 14-Jun-2023].
- [4] “Deep Learning ou Apprentissage Profond : qu’est-ce que c’est ? — datascientest.com” – <https://datascientest.com/deep-learning-definition>, [Accessed 12-Jun-2023].
- [5] “Estimation de mouvement — Wikipédia — fr.wikipedia.org” – [https://fr.wikipedia.org/wiki/Estimation\\_de\\_mouvement](https://fr.wikipedia.org/wiki/Estimation_de_mouvement), [Accessed 05-Jun-2023].
- [6] “How to Pray in Islam (with Pictures) - wikiHow — wikihow.com” – <https://www.wikihow.com/Pray-in-Islam>, [Accessed 01-Jun-2023].
- [7] “JetBrains Marketplace — plugins.jetbrains.com” – <https://plugins.jetbrains.com/pycharm>, [Accessed 24-May-2023].
- [8] “Machine Learning: What it is and why it matters — sas.com” – [https://www.sas.com/en\\_us/insights/analytics/machine-learning.html](https://www.sas.com/en_us/insights/analytics/machine-learning.html), [Accessed 01-Jun-2023].
- [9] “Matplotlib 2014; Visualization with Python — matplotlib.org” – <https://matplotlib.org/>, [Accessed 24-May-2023].
- [10] “NumPy — numpy.org” – <https://numpy.org/>, [Accessed 24-May-2023].
- [11] “pandas - Python Data Analysis Library — pandas.pydata.org” – <https://pandas.pydata.org/>, [Accessed 24-May-2023].
- [12] “Quels sont les différents types de données | Jedha — jedha.co” – <https://www.jedha.co/formation-analyse-donnee/types-donnees-analyse>, [Accessed 05-Jun-2023].
- [13] “SAP Insights — sap.com” – <https://www.sap.com/insights.html>, [Accessed 01-Jun-2023].
- [14] “scikit-learn: machine learning in Python 2014; scikit-learn 1.2.2 documentation — scikit-learn.org” – <https://scikit-learn.org/stable/index.html>, [Accessed 24-May-2023].
- [15] “What is Pickle in Python? How to do Pickling? (with code) — favtutor.com” – <https://favtutor.com/blogs/pickle-python>, [Accessed 24-May-2023].
- [16] “What is Python? Executive Summary — python.org” – <https://www.python.org/doc/essays/blurb/>, [Accessed 24-May-2023].
- [17] C. ABDELATIF – “Reconnaissance automatique des fruits à partir d’images naturelles en utilisant l’apprentissage automatique”.
- [18] N. AHMAD, L. HAN, K. IQBAL, R. AHMAD, M. A. ABID et N. IQBAL – “Sarm: Salah activities recognition model based on smartphone”, *Electronics* **8** (2019), no. 8, p. 881.
- [19] R. AL-GHANNAM et H. AL-DOSSARI – “Prayer activity monitoring and recognition using acceleration features with mobile phone”, *Arabian Journal for Science and Engineering* **41** (2016), p. 4967–4979.
- [20] Z. H. AYAD SAWSAN – “Recouvrement de posture 2d”, (2011).

- [21] A. F. BOBICK et J. W. DAVIS – “The recognition of human movement using temporal templates”, *IEEE Transactions on pattern analysis and machine intelligence* **23** (2001), no. 3, p. 257–267.
- [22] K.-M. CHEUNG, S. BAKER et T. KANADE – “Shape-from-silhouette across time part i: Theory and algorithms”, *International Journal of Computer Vision* **62** (2005), p. 221–247.
- [23] G. CLAUDE – “Collecte de données : caractéristiques, techniques et exemples — scribbr.fr”, <https://www.scribbr.fr/methodologie/collecte-de-donnees/>, [Accessed 05-Jun-2023].
- [24] N. DALAL et B. TRIGGS – “Histograms of oriented gradients for human detection”, *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05)*, vol. 1, Ieee, 2005, p. 886–893.
- [25] I. HARITAOGLU, D. HARWOOD et L. S. DAVIS – “W/sup 4: real-time surveillance of people and their activities”, *IEEE Transactions on pattern analysis and machine intelligence* **22** (2000), no. 8, p. 809–830.
- [26] C. HARRIS, M. STEPHENS et al. – “A combined corner and edge detector”, *Alvey vision conference*, vol. 15, Citeseer, 1988, p. 10–5244.
- [27] G. JOHANSSON – “Visual perception of biological motion and a model for its analysis”, *Perception & psychophysics* **14** (1973), p. 201–211.
- [28] K.-P. KARMANN – “Moving object recognition using an adaptive background memory”, *Time-varying image processing and moving object recognition* **2** (1990), p. 289–296.
- [29] Y. KE, R. SUKTHANKAR et M. HEBERT – “Volumetric features for video event detection”, *International journal of computer vision* **88** (2010), p. 339–362.
- [30] A. KLASER, M. MARSZALEK et C. SCHMID – “A spatio-temporal descriptor based on 3d-gradients”, *BMVC 2008-19th British Machine Vision Conference*, British Machine Vision Association, 2008, p. 275–1.
- [31] A. KOUBÂA, A. AMMAR, B. BENJDIRA, A. AL-HADID, B. KAWAF, S. A. AL-YAHRI, A. BABIKER, K. ASSAF et M. B. RAS – “Activity monitoring of islamic prayer (salat) postures using deep learning”, *2020 6th Conference on Data Science and Machine Learning Applications (CDMA)*, IEEE, 2020, p. 106–111.
- [32] I. LAPTEV, B. CAPUTO, C. SCHÜLDT et T. LINDBERG – “Local velocity-adapted motion events for spatio-temporal recognition”, *Computer vision and image understanding* **108** (2007), no. 3, p. 207–229.
- [33] I. LAPTEV et T. LINDBERG – “Interest point detection and scale selection in space-time”, *Scale Space Methods in Computer Vision: 4th International Conference, Scale Space 2003 Isle of Skye, UK, June 10–12, 2003 Proceedings*, Springer, 2003, p. 372–387.
- [34] W.-L. LU et J. J. LITTLE – “Simultaneous tracking and action recognition using the pca-hog descriptor”, *The 3rd Canadian Conference on Computer and Robot Vision (CRV’06)*, IEEE, 2006, p. 6–6.
- [35] Y. LUO, T.-D. WU et J.-N. HWANG – “Object-based analysis and interpretation of human motion in sports video sequences by dynamic bayesian networks”, *Computer Vision and Image Understanding* **92** (2003), no. 2-3, p. 196–216.
- [36] S. H. MOHIUDDIN, T. SYED et B. KHAN – “Salat activity recognition on smartphones using convolutional network”, *2022 International Conference on Emerging Trends in Smart Technologies (ICETST)*, IEEE, 2022, p. 1–5.
- [37] G. MORI, A. BERG, A. EFROS, A. EDEN et J. MALIK – “Video based motion synthesis by splicing and morphing”, *Computer Science* (2004).
- [38] J. OHYA, A. UTSUMI et J. YAMATO – *Analyzing video sequences of multiple humans: Tracking, posture estimation, and behavior recognition*, vol. 3, Springer Science & Business Media, 2002.
- [39] M. M. RAHMAN, R. A. A. ALHARAZI et M. K. I. B. Z. BADRI – “Intelligent system for islamic prayer (salat) posture monitoring”, *IAES International Journal of Artificial Intelligence* **12** (2023), no. 1, p. 220.
- [40] D. RAMANAN et D. FORSYTH – “Automatic annotation of everyday movements”, *Advances in neural information processing systems* **16** (2003).
- [41] S. RUSSELL et P. NORVIG – *Intelligence artificielle: Avec plus de 500 exercices*, Pearson Education France, 2010.

- [42] M. SELMI – “Reconnaissance d’activités humaines à partir de séquences vidéo”, Thèse, Institut National des Télécommunications, 2014.
- [43] L. WANG, W. HU et T. TAN – “Recent developments in human motion analysis”, *Pattern recognition* **36** (2003), no. 3, p. 585–601.
- [44] Z. ZHANG, Y. HU, S. CHAN et L.-T. CHIA – “Motion context: A new representation for human action recognition”, *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part IV 10*, Springer, 2008, p. 817–829.
- [45] Z.-H. ZHOU – *Machine learning*, Springer Nature, 2021.