

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de L'Enseignement Supérieur et de la Recherche Scientifique



**UNIVERSITÉ FERHAT ABBAS - SETIF1**  
**FACULTÉ DES SCIENCES**

**THÈSE**

**Présentée au Département d'informatique**

**Pour l'obtention du diplôme de**

**DOCTORAT**

**Domaine : Mathématiques et Informatique**

**Filière : Informatique**

**Option : Interaction Homme-Machine Avancée**

**Par**

**IZOUNTAR Yousra**

**THÈME**

**Contribution à la conception et à l'implémentation de collecticiels malléables pour les Environnements Virtuels/Augmentés collaboratifs (EVC/EAC)**

**Soutenue le 14/02/2024 devant le Jury:**

<b>Sadik BESSOU</b>	<b>M.C.A</b>	<b>Univ. Ferhat Abbas Sétif 1</b>	<b>Président</b>
<b>Fateh SEGHIR</b>	<b>M.C.A.</b>	<b>Univ. Ferhat Abbas Sétif 1</b>	<b>Examineur</b>
<b>Mahmoud BELHOCINE</b>	<b>DR</b>	<b>CDTA</b>	<b>Examineur</b>
<b>Samir OTMANE</b>	<b>Professeur</b>	<b>Université d'Evry Paris-Saclay</b>	<b>Directeur de thèse</b>
<b>Samir BENBELKACEM</b>	<b>M.R.A.</b>	<b>CDTA</b>	<b>Co-Directeur</b>
<b>Abdallah KHABABA</b>	<b>Professeur</b>	<b>Univ. Ferhat Abbas Sétif 1</b>	<b>invité</b>
<b>Nadia ZENATI-HENDA</b>	<b>DR</b>	<b>CDTA</b>	<b>Invité</b>

# DEDICATION

Je dédie ce modeste travail, comme preuve de respect, de gratitude et de reconnaissance :

À Mes chers parents, qui ont été une source constante d'encouragement, de conseils et d'aide tout au long de mon parcours. Leur amour inconditionnel, leur compréhension et leur patience ont été des piliers essentiels dans ma vie. Leur soutien moral et matériel a été inestimable, et il est impossible de véritablement exprimer les sentiments que j'éprouve à leur égard.

À Ma belle mère, Naziha, Dieu ait son âme.

À Mon cher Mari, Mohamed Zine.

À Mon fils, Adem.

À Toute ma famille.

À **vous tous,**

Je dédie ce travail.

# REMERCIEMENT

-Je remercie Dieu car il m'a permis d'accomplir ce travail par sa grâce, à lui la louange avant tout.

-Je souhaite aussi remercier mon encadreur, Samir BENBELKACEM, pour le temps précieux qu'il a consacré à me transmettre les outils méthodologiques essentiels à la réalisation de cette recherche.

-Je souhaite exprimer ma sincère reconnaissance envers mon directeur de thèse, Monsieur Samir OTMANE, pour sa patience inébranlable et ses précieux conseils qui ont nourri ma réflexion tout au long de ce parcours.

-Je souhaite exprimer ma sincère gratitude envers Monsieur Abdallah KHABABA.

-Je tiens à remercier le président Maître de conférence A Sadik BESSOU, ainsi que les examinateurs Maître de conférence A Fateh SEGHIR, et le directeur de recherche Mahmoud BELHOCINE, les invités, le professeur Abdallah KHABABA et le directeur de recherche Nadia ZENATI-HENDA, pour leur précieuse contribution et leur participation active lors de cette évaluation. Leur présence et leurs commentaires ont grandement enrichi cette expérience.

-Je désire remercier les membres de l'équipe IRVA de CDTA, notamment le chef d'équipe, Madame Nadia ZENATI-HENDA, qui m'ont fourni les outils nécessaires et m'ont soutenu moralement et intellectuellement tout au long de mon parcours.

-J'exprime ma gratitude envers mes parents bien-aimés, Rekia et Foudil, qui ont toujours été présents à mes côtés.

-Je remercie mon cher Mari, Mohamed zine BOUZIANE, qui m'a toujours soutenue tout au long de ce parcours.

-Je souhaite remercier mon oncle Ali, ma tante Taklit, ainsi que mes cousins et cousines pour leur soutien précieux. Un grand merci à ma cousine Dalila et à son mari Toufik.

-Je remercie également mes sœurs, Badiàa, Chahra et Mounira, mes frères, Ferhat, Ousama et Slimane. Je veux remercier également, mon beau-père, Amer, mes beaux-frères, Abderrahman, Hacem, Aimen, Fouad et Ali, mes belles-sœurs, Zina, Lamia, Warda, Sarah et Siham, pour leurs soutiens.

# RÉSUMÉ

La thèse s'inscrit dans le domaine de l'interface homme-machine (IHM) et plus précisément dans celui des technologies immersives telles que la réalité virtuelle et augmentée.

L'objectif est de développer des collecticiels immersifs malléables qui prennent en compte les différents contextes des utilisateurs et leurs environnements d'usage. Trois axes de recherche se sont dégagés, qui sont, les environnements virtuels collaboratifs (EVC), les collecticiels, et la malléabilité des EVC. Les travaux ont conduit à deux contributions majeures. La première est une architecture logicielle pour une plateforme virtuelle malléable d'assistance à la rééducation fonctionnelle, qui se compose de trois modules : un module de capture et d'interprétation des données, un module d'analyse des données, et un module d'adaptation du jeu d'exercice en fonction de l'émotion du patient. Des tests utilisateurs ont montré que cette plateforme virtuelle était malléable en fonction de l'état émotionnel du patient. La deuxième contribution consiste à proposer une étude complète et détaillée d'un système visant à développer des Environnements Virtuels Collaboratifs Malléables (EVCM), comprenant une définition détaillée de celui-ci et de ses composants essentiels, qui pourraient être l'objectif de toute recherche visant à développer un système collaboratif malléable ; composé de trois parties principales : (1) gestionnaire du travail collectif, (2) gestionnaire de l'adaptation et (3) gestionnaire du contexte. Ceux-ci ont été assemblés dans un nouveau modèle appelé MVC-3DCM. De plus, nous avons présenté une étude de cas d'un jeu coopératif sérieux pour illustrer comment mettre en œuvre le modèle proposé.

**Mots Clés :** Réalité virtuelle (RV), Environnements virtuels/augmentés collaboratifs (EVC/EAC), Travail collaboratif assisté par ordinateur(TCAO), Malléabilité, Sensibilité au contexte.



# SUMMARY

The thesis falls within the fields of human-computer interaction (HCI) and more specifically immersive technologies such as virtual and augmented reality.

The objective is to develop flexible immersive groupware that takes into account the different contexts of users and their usage environments. Three research axes emerged, including collaborative virtual environments (CVEs), groupware, and the malleability of CVEs. The research led to two major contributions. The first is a software architecture for a malleable virtual platform to assist in functional rehabilitation, which consists of three modules : a data capture and interpretation module, a data analysis module, and an exercise game adaptation module based on the patient's emotions. User tests showed that this virtual platform was malleable based on the patient's emotional state. The second contribution consists of proposing a comprehensive and detailed study of a system aimed at developing Malleable Collaborative Virtual Environments (EVCM), including a detailed definition of it and its essential components, which could be the objective of any research aiming to develop a malleable collaborative system ; composed of three main parts : (1) collective work manager, (2) adaptation manager, and (3) context manager. These have been assembled into a new model called MVC-3DCM. Additionally, we have presented a case study of a serious cooperative game to illustrate how to implement the proposed model.

**Key Words :** Virtual reality (RV), Collaborative virtual/augmented environments (EVC/EAC), Computer-supported collaborative work (CSCW), Malleability, Context sensitivity.

## ملخص:

أطروحتنا بعنوان "المساهمة في تصميم وتنفيذ البرامج التكيفية للبيئات الافتراضية / المعززة التعاونية ، وترتكز على مجالات واجهة الإنسان والآلة، والعمل التعاوني ، الواقع الافتراضي / الواقع المعزز. الهدف هو تطوير برامج جماعية قابلة للتكيف ديناميكيا تأخذ بعين الاعتبار السياقات المختلفة للمستخدمين وبيئات الاستخدام الخاصة بهم. قمنا بدراسة ثلاثة خطوط بحثية، وهي البيئات الافتراضية التعاونية ، والبرامج الجماعية، وليونة الأنظمة الافتراضية التعاونية. أدى بحثنا إلى مساهمتين رئيسيتين. المساهمة الأولى هي تصميم وإنتاج منصة افتراضية مرنة تكيفية للمساعدة في إعادة التأهيل الوظيفي، والتي تتكون من ثلاث وحدات: وحدة التقاط البيانات وتفسيرها، ووحدة تحليل البيانات، ووحدة التكيف التي تعمل على جعل اللعبة متكيفة وحالة المريض حسب حالته. أما المساهمة الثانية فتتمثل في اقتراح دراسة شاملة ومفصلة للبيئات التعاونية الافتراضية التكيفية، بما في ذلك تقديم تعريف مفصل له وكذا مكوناته الأساسية والتي قد تكون مقصدا لأي بحث يريد تطوير نظام تعاوني تكيفي؛ يتكون من ثلاث أجزاء أساسية (1) المسؤول عن العمل الجماعي، (2) المسؤول عن التكيف و(3) المسؤول عن السياق. كما قدمنا أيضاً دراسة حالة للعبة جادة تعاونية لتوضيح كيفية تطبيق النموذج المقترح.

**الكلمات المفتاحية:** الواقع الافتراضي ، البيئات الافتراضية/المعززة التعاونية، العمل التعاوني ، المرنة، الحساسية للسياق .

# Table des matières

Liste des Figures	i
Liste des Tableaux	iv
Liste des Abréviations	v
Introduction Générale	1
<b>I Etat de l’art</b>	<b>7</b>
<b>1 Les Environnements virtuels collaboratifs (EVC)</b>	<b>8</b>
1.1 Introduction	8
1.2 La réalité virtuelle	9
1.2.1 Bref Historique	9
1.2.2 Définitions	10
1.3 Les Environnements virtuels collaboratifs	14
1.3.1 Définition	14
1.4 Architectures des environnements virtuels collaboratifs	15
1.4.1 Architecture pair-à-pair	16
1.4.2 Architecture client-serveur	17
1.4.3 Architecture hybride	17
1.5 Distribution des données	18
1.5.1 Mode centralisé	19
1.5.2 Mode homogènement répliqué	20
1.5.3 Mode partiellement répliqué	21
1.6 Interactions dans les EVC	23
1.6.1 Les tâches d’interaction 3D	23
1.6.2 Les techniques d’interaction multi-utilisateurs	27
1.6.3 Conclusion intermédiaire	34

---

1.7	Exemples d'environnements virtuels collaboratifs . . . . .	35
1.7.1	FréjusVR : simulateur de réalité virtuelle pour la formation des pompiers . . . . .	36
1.7.2	MS2OT :Multi-Step State-Action-Value-based Optimized Trajectory	37
1.7.3	CollaboVR :méta modèle d'un système de réalité virtuelle pour la collaboration créative . . . . .	38
1.8	Conclusion . . . . .	39
<b>2</b>	<b>La malléabilité des Environnements virtuels collaboratifs</b>	<b>41</b>
2.1	Introduction . . . . .	41
2.2	La malléabilité . . . . .	43
2.3	La plasticité 3D . . . . .	44
2.4	La RV et la sensibilité au contexte . . . . .	46
2.5	Adaptation des environnements virtuels . . . . .	48
2.5.1	Classification des systèmes adaptatifs . . . . .	48
2.5.2	Architecture d'un système de réalité virtuelle adaptatif de formation	50
2.6	Synthèse . . . . .	51
2.6.1	Contraintes et défis relevés . . . . .	51
2.6.2	Nouvelles exigences . . . . .	52
2.6.3	Notre démarche de recherche . . . . .	56
2.7	Les systèmes sensibles au contexte . . . . .	58
2.7.1	Le contexte dans les systèmes classiques . . . . .	58
2.7.2	Le contexte dans les applications RV . . . . .	61
2.7.3	Architecture générale d'un système sensible au contexte . . . . .	69
2.8	Quelque systèmes virtuels collaboratifs sensible au contexte . . . . .	74
2.8.1	vr-UCAM : un module sensible au contexte pour la réalité virtuelle	74
2.8.2	Unified Context-aware Augmented Reality . . . . .	75
2.8.3	Système d'incarnation virtuelle . . . . .	77
2.8.4	Tango :h un systeme de recommandation d'un jeu serieux de rééducation . . . . .	78
2.9	Conclusion . . . . .	80
<b>3</b>	<b>Les collecticiels</b>	<b>82</b>
3.1	Introduction . . . . .	82
3.2	Définition . . . . .	83
3.3	Modèle du trèfle fonctionnel des collecticiels . . . . .	86
3.4	Modèles d'architecture pour les collecticiels . . . . .	87

3.4.1	Modèle ALV . . . . .	87
3.4.2	Modèle Zipper . . . . .	88
3.4.3	Modèle CoPAC . . . . .	89
3.4.4	Modèle PAC* . . . . .	90
3.4.5	Modèle AMF-C . . . . .	91
3.4.6	Méta modèle de Dewan . . . . .	92
3.4.7	Modèle Clock et DragonFly . . . . .	93
3.4.8	Modèle Clover . . . . .	94
3.4.9	Modèle C4 . . . . .	95
3.4.10	Modèle MVC-3DC . . . . .	96
3.5	Malléabilité des collecticiels . . . . .	98
3.6	Approches et méthodologie des collecticiels malléables . . . . .	99
3.6.1	Théorie de l'activité et la coévolution . . . . .	99
3.6.2	Approche à base de workflow . . . . .	101
3.6.3	Approche à base d'objets médiateurs . . . . .	103
3.6.4	Approche à base d'objets composants . . . . .	103
3.6.5	Approche à base de système multi-agent . . . . .	105
3.7	Synthèse . . . . .	107
3.7.1	Analyse des architectures des collecticiels 2D/3D . . . . .	107
3.7.2	Analyse des méthodologie des collecticiels malléables . . . . .	108
3.7.3	Bilan . . . . .	109
3.8	Conclusion . . . . .	110

## II Contributions 112

4	Conception et réalisation d'un Environnement Virtuel Malléable(EVM) <span style="float: right;">113</span>
4.1	Introduction . . . . . <span style="float: right;">113</span>
4.2	Jeux sérieux et rééducation . . . . . <span style="float: right;">114</span>
4.3	L'architecture de la plateforme virtuelle collaborative malléable d'assistance à la rééducation fonctionnelle . . . . . <span style="float: right;">115</span>
4.3.1	Capture et interprétation des données . . . . . <span style="float: right;">116</span>
4.3.2	Analyse des données . . . . . <span style="float: right;">117</span>
4.3.3	L'adaptation . . . . . <span style="float: right;">118</span>
4.4	Un jeux sérieux malléable à base de la reconnaissance faciale . . . . . <span style="float: right;">119</span>
4.4.1	Architecture technique . . . . . <span style="float: right;">120</span>
4.4.2	Description . . . . . <span style="float: right;">121</span>

4.4.3	Test et évaluation . . . . .	122
4.5	Conclusion . . . . .	127
<b>5</b>	<b>Conception et réalisation d'un</b>	
	<b>Environnement Virtuel Collaboratif Malléable(EVCM)</b>	<b>129</b>
5.1	Introduction . . . . .	129
5.2	Notre définition d'un EVCM . . . . .	131
5.3	Architecture de base d'un EVCM . . . . .	131
5.4	La boucle de malléabilité dans un EVCM . . . . .	133
5.5	La décomposition fonctionnelle de notre système . . . . .	135
5.5.1	L'espace motrice . . . . .	135
5.5.2	L'espace de données . . . . .	156
5.6	MVC-3DCM : Modèle d'architecture de notre système . . . . .	162
5.7	Modélisation . . . . .	164
5.7.1	Diagramme de classe de notre système . . . . .	165
5.7.2	Architecture réseau de notre système . . . . .	167
5.7.3	Diagramme d'activité d'un scénario de collaboration . . . . .	168
5.8	Étude de cas : jeu sérieux collaboratif malléable pour la rééducation . . . . .	170
5.8.1	Description du jeu . . . . .	171
5.8.2	Scénario d'exécution . . . . .	175
5.8.3	Le processus de malléabilité . . . . .	178
5.8.4	Simulation . . . . .	185
5.9	Conclusion . . . . .	187
	<b>Conclusion Générale</b>	<b>188</b>
	<b>Bibliographie</b>	<b>192</b>

# Liste des Figures

0.1	<i>Les composants d'un EVCM</i> . . . . .	1
1.1	Le premier système de réalité virtuelle(extrait de [56]). . . . .	10
1.2	<i>Schéma d'interaction entre l'homme et la réalité virtuelle[47].</i> . . . .	13
1.3	<i>Méthodes de communication réseau</i> . . . . .	16
1.4	<i>Architecture pair-à-pair.</i> . . . . .	16
1.5	<i>Architecture client-serveur.</i> . . . . .	17
1.6	Architecture hybride. . . . .	18
1.7	<i>(a) L'exécution et (b)la modification du comportement des objets en mode centralisé(extrait de [46]).</i> . . . . .	19
1.8	<i>(a)L'exécution et (b)la modification du comportement des objets en mode homogènement répliqué(extrait de [46]).</i> . . . . .	20
1.9	<i>Répartition partielle des données chez les utilisateurs(extrait de [46]).</i> . . . .	22
1.10	Classification des techniques de navigation de groupe en réalité virtuelle [147]. . . . .	24
1.11	Aperçu d'un environnement virtuel collaboratif pour la visualisation de molécules (extrait de [11]) . . . . .	28
1.12	Course à deux joueurs sur écran partagé dans Gran Turismo 5 : chaque joueur ne peut agir que sur son véhicule et peut communiquer en direct, sans influence sur le véhicule adverse. . . . .	29
1.13	Application de forces sur un objet virtuel à l'aide de simulations de ressorts virtuels [98]. . . . .	31
1.14	Approches collaboratives pour la manipulation d'objets 3D dans des environnements virtuels(extrait de [40]). . . . .	32
1.15	<i>La technique de manipulation "Bent Pick Ray" (Extrait de [112])</i> . . . . .	32
1.16	<i>Illustration de casque FaceDisplay(extrait de [54]).</i> . . . . .	33
1.17	<i>Illustration de la technique de Manipulation collaborative basée sur la qualité du point de vue(extrait de [146]).</i> . . . . .	34
1.18	<i>Illustration de jeu sérieux FrejusVR de formation(extrait de [25]).</i> . . . . .	36
1.19	<i>Système de redirection de marche en réalité virtuelle(extrait de [84]).</i> . . . .	37
1.20	<i>Quatre exemples d'études de cas utilisant le framework CollaboVR(extrait de [55]).</i> . . . . .	39
2.1	<i>Notre méthodologie de recherche.</i> . . . . .	42
2.2	<i>Les dimensions de la plasticité 3D[80].</i> . . . . .	45
2.3	<i>Architecture pour les systèmes VR adaptatifs de formation. [151].</i> . . . . .	50
2.4	<i>Les trois dimensions de notre système EVCM.</i> . . . . .	56

2.5	<i>Quelques capteurs utilisés dans les environnements virtuels [94].</i>	63
2.6	<i>Exemple d'utilisation des données oculaire dans les environnements VR [1].</i>	65
2.7	<i>HTC VIVE Pro Eye HMD.</i>	66
2.8	<i>Les joueurs élèvent chaque segment en se concentrant sur les commandes push/pull qu'ils ont définies[37].</i>	67
2.9	<i>Quelques capteurs EEG (A)Neurables HTC Vive with Wearable Sensing électrodes , (B), (C) 5-channel Emotiv InsightLooxid Link affixed to HTC Vive, (D)5-channel Muse 2</i>	67
2.10	<i>L'incarnation du visage[37].</i>	68
2.11	<i>Architecture d'un système sensible au contexte.</i>	70
2.12	<i>L'architecture de vr-UCAM[85].</i>	74
2.13	<i>Architecture d'application de réalité augmentée sensible au contexte[120].</i>	76
2.14	<i>visualisation des données contextuelle[94].</i>	77
2.15	<i>Le miroir virtuel dans la scène[94].</i>	77
2.16	<i>L'architecture de la plateforme TANGO :H[51].</i>	79
3.1	<i>Matrice espace-temps pour le travail coopératif assisté par ordinateur(TCAO)[66].</i>	84
3.2	<i>Modèle du trèfle fonctionnel [116]</i>	86
3.3	<i>Modèle ALV[59].</i>	88
3.4	<i>Modèle zipper [103].</i>	89
3.5	<i>Modèle CoPAC[116].</i>	90
3.6	<i>Modèle PAC* [82].</i>	91
3.7	<i>Modèle AMF-C[135].</i>	91
3.8	<i>Modèle Dewan[35].</i>	92
3.9	<i>Modèle clock[53].</i>	93
3.10	<i>Modèle Clover [82].</i>	95
3.11	<i>Modèle C4 [71].</i>	96
3.12	<i>Modèle MVC-3DC [16].</i>	97
3.13	<i>Capture d'écran de DARE[21].</i>	100
3.14	<i>Exemple illustrant l'approche de la malléabilité dans les systèmes de workflow [32].</i>	102
3.15	<i>Approche basée sur des médiateurs[132].</i>	103
3.16	<i>Architecture d'un collecticiel malléable à base de services Web d'agents logiciels[28].</i>	106
4.1	<i>VR :PEER :architecture globale d'un jeu sérieux malléable basé sur la reconnaissance des émotions[63]</i>	116
4.2	<i>L'architecture logique du jeu développé. [63]</i>	120
4.3	<i>Jeu sérieux adaptatif développé sur la base de l'architecture VR-PEER. (a) : interface du participant, (b) : interface utilisée pour tester la reconnaissance des émotions faciales. [63].</i>	122
4.4	<i>La progression de l'état émotionnel. [63]</i>	123
4.5	<i>Valeur de TimeIntervalle à chaque point de contrôle avec l'application et sans l'application de l'adaptation. [63]</i>	126
5.1	<i>Architecture d'un EVC proposé par [99].</i>	132



---

5.2	<i>Notre architecture de base d'un EVCM.</i>	133
5.3	<i>La boucle de malléabilité dans un EVCM.</i>	134
5.4	<i>Les composants d'un EVCM.</i>	135
5.5	<i>Gestionnaire de workflow [38]</i>	137
5.6	<i>La fonction de navigation [38].</i>	138
5.7	<i>La fonction de sélection [38].</i>	138
5.8	<i>La fonction de manipulation [38].</i>	139
5.9	<i>Classification des données.</i>	144
5.10	<i>Gestionnaire de malléabilité.</i>	149
5.11	<i>Illustration de l'espace de données</i>	157
5.12	<i>Exemple illustratif d'utilisation de la fonction Nimbus dans un EVC.</i>	159
5.13	<i>MVC-3DCM : Modèle d'architecture de notre système</i>	163
5.14	<i>Diagramme de classe</i>	166
5.15	<i>Architecture réseau de notre système</i>	168
5.16	<i>Diagramme d'activité</i>	169
5.17	<i>Patient jouant à un jeu sérieux de rééducation avec le Leap Motion</i>	172
5.18	<i>Illustration du Geste 1 :toucher sans force</i>	173
5.19	<i>Illustration du Geste 2 :toucher avec force</i>	173
5.20	<i>Illustration du Geste 3 :saisie et relâcher</i>	174
5.21	<i>Illustration du Geste 4-Zoom-in</i>	175
5.22	<i>Illustration du Geste 4-zoom-out</i>	175
5.23	<i>Vue en plan de jeu avec trois patients jouant ensemble - Champs de vision de chaque patient(focus) à droite</i>	176
5.24	<i>Zoom sur les pommes sélectionnés pour les tests à l'instant <math>t</math></i>	177
5.25	<i>L'évolution des émotions pendant la séance d'entraînement pour les patients 1, 2 et 3.</i>	185

# Liste des Tableaux

2.1	La compatibilité des définitions existantes avec nos exigences. "-" ne prend pas en compte l'exigence, "+" prend partiellement en compte l'exigence, "++" prend fortement en compte l'exigence. . . . .	54
3.1	Analyse des architectures existantes en termes de leur prise en compte des dimensions de l'EVC. . . . .	108
4.1	Exemple des données enregistrées pour chaque sujet pendant l'entraînement.	125
5.1	Exemples des capteurs existants pour les différents types de données. . . . .	148
5.2	Quelques méthode d'interprétation utilisées dans la littérature . . . . .	150
5.3	Nimbus de chaque objet dans l'EVC . . . . .	159
5.4	Nimbus des pommes 0-6 . . . . .	178
5.5	<i>Processus de malléabilité des pommes 0,2,4 dans les points de contrôle 3,6,9</i>	186

# Liste des Abréviations

**RV** Réalité Virtuelle

**RA** Réalité Augmentée

**EVC** Environnement Virtuel Collaboratif

**EAC** Environnement Augmenté Collaboratif

**EVCMM** Environnement Virtuel Collaboratif Malléable

**3D** Trois Dimensions

**TC** Travail Collaboratif

**IHM** Interaction Homme Machine

**TCAO** Travail Collaboratif Assisté par Ordinateur

**MVC-3DCM** modèle-vue-contrôleur 3d collaboratif malléable

**GW** Gestionnaire de workflow

**GC** Gestionnaire de contexte

**GM** Gestionnaire de malléabilité

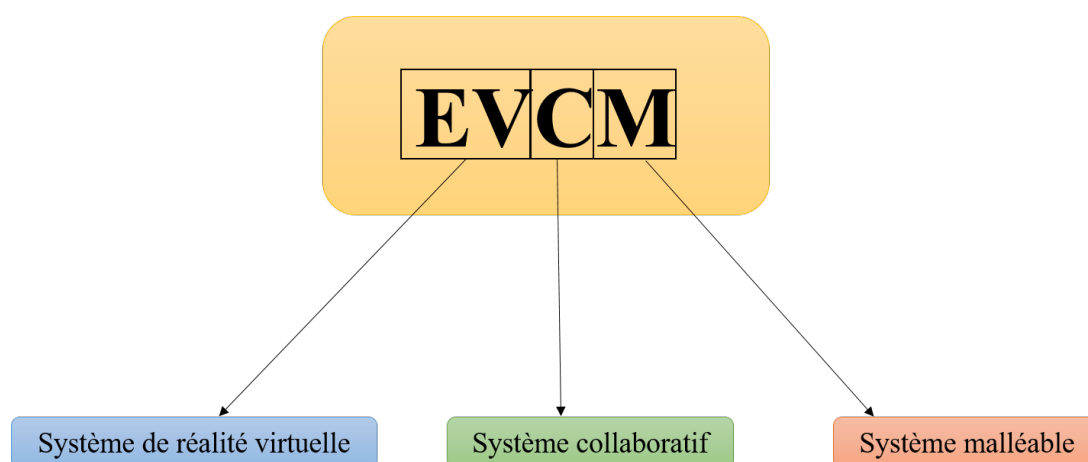
**UML** Unified Modeling Language

**JSC** jeu sérieux collaboratif

# Introduction Générale

## Contexte

Cette thèse présente une approche novatrice qui combine les techniques issues de la réalité virtuelle (RV), de malléabilité, et du travail collaboratif assisté par ordinateur (TCAO) pour concevoir et mettre en oeuvre un environnement virtuel collaboratif malléable (EVCM), comme illustré dans la figure ci-dessous.



**Figure 0.1:** *Les composants d'un EVCM*

La malléabilité se réfère à la capacité d'un objet, d'un système ou d'un environnement à être facilement modifié ou façonné pour s'adapter à des situations changeantes ou pour répondre à des besoins particuliers. Dans les environnements virtuels collaboratifs (EVCs), la malléabilité implique la possibilité de l'environnement en question à s'adapter aux préférences et aux besoins des utilisateurs, en leur offrant des options de personnalisation et/ou en ajustant automatiquement certains aspects en fonction de leur contexte. Le concept de malléabilité des EVCs tire parti des avantages de chaque domaine cité auparavant : (1) la RV offre une expérience immersive, interactive et engageante pour les utilisateurs, (2) le TCAO fournit des outils de collaboration et de communication efficaces pour travailler ensemble à distance. En intégrant la malléabilité, ce concept vise à

permettre une flexibilité et une adaptation fluides aux besoins changeants des utilisateurs.

Étant donné le manque de support théorique approfondi sur l'intégration de la malléabilité dans les EVCs, notre approche méthodologique propose de considérer les fondements théoriques de la malléabilité lors de la conception des EVC. Pour ce faire, nous introduisons le concept d'Environnement Virtuel Collaboratif Malléable (EVCM), un nouvel acronyme créé pour décrire ce type d'environnements. Notre contribution se base sur le concept de sensibilité au contexte, qui représente l'un des aspects importants de la malléabilité dans notre approche. En personnalisant l'expérience de chaque participant en prenant en compte son contexte et son environnement, nous améliorons l'immersion, l'engagement et l'efficacité des utilisateurs dans l'EVCM. La conception et le développement d'un tel système impliquent l'utilisation de techniques et d'approches liées à la sensibilité au contexte, l'adaptation de contenu 3D, les techniques d'interaction et les protocoles de collaboration.

## Problématique

La démocratisation des outils numériques ainsi que les avancées en réalité virtuelle (RV) et réalité augmentée (RA) dans l'industrie (industrie 4.0) et les services à la personne tel que l'hôpital du futur, rendent indispensable le développement d'applications permettant la collaboration entre individus (opérateur-opérateur, médecin-médecin, médecin-patient, etc.). Cette collaboration de groupe, qui peut s'effectuer sur des systèmes appelés collecticiels, permet aux individus d'échanger, de se coordonner et de partager des ressources, des informations et des tâches. Cela contribue grandement à l'accomplissement efficace de tâches complexes et favorise la prise de décisions rapides. Cependant, l'hétérogénéité des profils d'utilisateurs (niveau d'expertise, type de handicap, formateur, apprenant, etc.) ainsi que la nature des tâches à réaliser (simples, modérées et complexes) nécessitent l'introduction de nouveaux concepts théoriques et pratiques pour le développement de systèmes collaboratifs novateurs appelés collecticiels malléables qui s'adaptent aux différents profils d'utilisateurs et aux exigences des tâches à accomplir.

Ainsi, le projet de thèse vise à concevoir et à développer un système collaboratif, également appelé collecticiel, malléable/adaptatif qui évolue dans un environnement virtuel/augmenté interactif. Ce système permettra à plusieurs utilisateurs en situation de collaboration de :

- 1) Partager et échanger des informations (principalement des données 3D) afin de coordonner leurs tâches et réaliser une action commune dans de meilleures conditions.
- 2) S'adapter dynamiquement aux nouveaux besoins sans interruption du système.

Ainsi, les questions de recherche sous-jacentes sont :

1. La malléabilité dans les environnements collaboratifs classiques est-elle la même que la malléabilité dans les environnements augmentés/virtuels collaboratifs ? Si oui, quel type de malléabilité faut-il utiliser pour les EVC/EAC ? Dans le cas contraire, est-il possible de donner une définition de la malléabilité dans les environnements virtuels/augmentés collaboratifs ?
2. Quels sont les spécificités d'un collecticiel malléable dans les EVC/EAC par rapport à un collecticiel classique ? Comment concevoir une architecture logicielle malléable pour les "collecticiels 3D" qui peut s'adapter aux changements des besoins et comportements des utilisateurs en situation collaborative de manière dynamique, sans intervention de l'utilisateur et sans interruption du système ?
3. Les architectures existantes sont-elles appropriées ou inappropriées pour modéliser un collecticiel malléable dans les EVC/EAC ? Si oui, quelle est l'architecture convenable ? Dans le cas contraire, peut-on utiliser une architecture existante et ajouter des aspects propres aux EVC/EAC ? Sinon, faut-il réfléchir à une nouvelle démarche ?
4. Peut-on mettre en place un processus d'adaptation intelligent qui prend en compte les comportements des utilisateurs sur le plan psychologique (émotions, douleurs, etc.), ainsi que leur profil (âge, préférences, maladies, etc.) et le contexte de la tâche en cours ? Ce processus fournirait ensuite une série d'ajustements à appliquer pour rendre l'EVC malléable.

Pour répondre à ces questions, notre recherche se concentre sur les travaux menés dans les domaines suivants :

1. Les architectures sensibles au contexte pour les aspects de malléabilité, en bénéficiant de processus d'adaptation de ce genre de système, notamment la capture, l'analyse, la décision et l'action.
2. Les collecticiels pour les aspects de collaboration grâce aux modèles d'architectures logicielles des collecticiels et du trèfle fonctionnel (communication, coordination et coopération).
3. La réalité virtuelle appliquée aux aspects liés à l'immersion afin de permettre aux utilisateurs de plonger dans un environnement virtuel et de vivre une expérience sensorielle réaliste et engageante, ainsi que les aspects liés aux techniques d'interaction 3D.
4. L'intelligence artificielle est utilisée comme exemple de méthode d'analyse pour aborder les problématiques liées à l'étude du contexte des utilisateurs dans notre étude de cas.

En combinant ces thèmes de recherche, nous développons une approche théorique et pratique novatrices pour concevoir un système de réalité virtuel collaboratif malléable intelligent, capable de s'adapter, de manière automatique aux besoins des utilisateurs en collaboration, et au contexte de la tâche en cours.

Dans notre cas, l'intelligence artificielle (IA) a pour objectif de comprendre et reproduire le fonctionnement de la cognition humaine, et elle est souvent associée à une capacité prédictive. Notre hypothèse repose sur l'utilisation de cette capacité de prédiction dans un jeu sérieux de rééducation fonctionnelle, conçu pour être utilisé par des utilisateurs individuels ou en collaboration. Ce jeu se focalise sur la reconnaissance des émotions, ce qui permet d'adapter dynamiquement les interactions et les stimuli en fonction des réactions émotionnelles des patients. Par le biais de cette expérimentation, nous pourrions évaluer l'efficacité et l'impact de notre modèle d'architecture dans un contexte concret de rééducation fonctionnelle.

Notre thèse porte sur la conception d'un collecticiel malléable pour les environnements virtuels et augmentés collaboratifs (EVCs & EACs). Bien que notre travail soit principalement axé sur les EVCs, notre contribution reste pertinente pour les EACs. En effet, les EACs combinent des environnements virtuels et le monde réel, mais notre contribution se limite à la partie virtuelle des EACs, tout comme elle s'applique aux EVCs.

## Organisation du manuscrit

La thèse est structurée en deux parties : l'état de l'art et les contributions. La première partie comprend trois chapitres. Le premier chapitre offre un aperçu sur les EVCs, en abordant la réalité virtuelle, et les notions connexes. Le deuxième chapitre se focalise spécifiquement sur la malléabilité, en explorant les concepts et les approches associés. Le dernier chapitre de cette partie examine les collecticiels dans le but d'étudier plus en profondeur l'aspect collaboratif.

La partie "contributions" est composée de deux chapitres. Le premier chapitre présente le développement d'un jeu malléable basé sur la reconnaissance des émotions, la conception d'un environnement virtuel (EV) mono-utilisateur malléable. Le chapitre suivant propose, dans sa première partie, une conception détaillée des EVCs, mettant l'accent sur leur architecture et leur fonctionnement. La seconde partie de ce chapitre présente une étude de cas portant sur la simulation d'un jeu sérieux malléable multi-utilisateurs, basée sur la reconnaissance des émotions faciales des patients. Cette partie met en pratique les concepts présentés dans les contributions précédentes.

## **Chapitre 1 : Les Environnements virtuels collaboratifs (EVCs)**

Ce chapitre introduit les EVCs en commençant par une discussion sur la réalité virtuelle, incluant un bref historique et des définitions. Il explore les concepts clés de l'immersion, de la présence et de l'interaction dans les environnements virtuels. Ensuite, il examine les architectures des EVCs, en se concentrant sur l'architecture réseau du système et la distribution de données, ainsi que les interactions dans les EVCs, notamment les tâches d'interaction 3D et les techniques d'interaction multi-utilisateurs. Le chapitre se termine par quelques exemples d'environnements virtuels collaboratifs existants.

## **Chapitre 2 : La malléabilité des Environnements virtuels collaboratifs**

Dans ce chapitre, nous explorons la malléabilité des environnements virtuels collaboratifs. Nous commençons par une définition de la malléabilité et son application aux collecticiels. Toutefois, étant donné le manque de ressources spécifiques sur la malléabilité des EVCs, nous explorons des notions connexes telles que la plasticité 3D, la sensibilité au contexte et l'adaptation. Pour chacune de ces notions, Nous décrivons une synthèse et expliquons comment celles-ci peuvent contribuer à notre travail.

Ensuite, nous avons identifié une série d'exigences pour un environnement virtuel collaboratif malléable. Nous avons évalué chaque notion en termes de sa capacité à répondre à ces exigences et avons conclu que la sensibilité au contexte était la plus proche de nos attentes. Nous avons donc mené une étude approfondie sur les systèmes sensibles au contexte dans les environnements virtuels (EVs).

## **Chapitre 3 : Les collecticiels**

Ce chapitre introduit les collecticiels, en commençant par une définition de ce concept. Nous présentons par la suite le modèle du trèfle fonctionnel des collecticiels, qui met en évidence les différentes dimensions des collecticiels, telles que la communication, la coordination, et la coopération. Ce chapitre examine ensuite différents modèles d'architecture pour les collecticiels, tels que le modèle ALV, le modèle Zipper, le modèle CoPAC, le modèle PAC\*, le modèle AMF-C, le méta modèle de Dewan, le modèle Clock et DragonFly, le modèle Clover, et le modèle C4. Enfin, nous explorons les approches et méthodologies des collecticiels malléables, en mettant l'accent sur la théorie de l'activité, les approches basées sur le workflow, les objets médiateurs, les objets composants et les systèmes multi-agents.



## **Chapitre 4 : Conception et réalisation d'un Environnement Virtuel Malléable**

Dans ce chapitre, nous avons traité de la malléabilité dans les environnements virtuels mono-utilisateur. Nous commençons par présenter notre plateforme malléable d'assistance à la rééducation fonctionnelle, qui comprend la capture et l'interprétation de données, l'analyse de données et l'adaptation. Ensuite, nous présentons un jeu sérieux malléable basé sur la reconnaissance faciale. Cette section contient une description du jeu, l'architecture technique, les tests et l'évaluation[63, 64]. En conclusion, nous mettons en évidence l'importance de la malléabilité dans les jeux sérieux de rééducation.

## **Chapitre 5 : Conception et réalisation d'un Environnement Virtuel Collaboratif Malléable**

Dans ce chapitre, nous présentons une contribution méthodologique pour la conception d'un environnement virtuel collaboratif malléable (EVCM). Nous avons tout d'abord proposé une définition de l'EVCM, ainsi que son architecture. Nous avons ensuite exposé la boucle de malléabilité, qui est le processus permettant à l'EVCM d'être adaptable aux changements dans son environnement.

Pour concrétiser notre contribution, nous avons procédé à une décomposition fonctionnelle de notre système, qui inclut un gestionnaire de workflow, un gestionnaire de contexte, un gestionnaire de malléabilité et un espace de données. Nous avons également défini le modèle architectural de notre système, Le MVC-3DCM, dérivé du modèle MVC (Modèle-Vue-Contrôleur), intégrant les composantes de la réalité virtuelle, les aspects collaboratifs et la malléabilité. Nous avons notre système à travers plusieurs diagrammes, notamment un diagramme de classe, une architecture réseau client-serveur, et un diagramme d'activité pour un scénario de collaboration. Nous avons ensuite présenté une étude de cas de notre système, en développant un jeu sérieux collaboratif malléable pour la rééducation fonctionnelle.

Enfin, nous avons conclu ce chapitre en résumant notre contribution et les résultats de notre étude de cas, qui ont montré la facilité d'utiliser de notre système dans le développement d'environnements virtuels adaptatifs et personnalisables. Nous estimons que notre contribution est une avancée importante dans le domaine de la conception d'environnements virtuels collaboratifs malléables, et elle pourra être utile pour de nombreuses applications telles que l'enseignement, la formation, la santé, etc.

# Première partie

## Etat de l'art

# Chapitre 1

## Les Environnements virtuels collaboratifs (EVC)

### 1.1 Introduction

Les environnements virtuels collaboratifs (EVC) sont devenus extrêmement importants dans les sociétés modernes, offrant de nouvelles opportunités de communication et de collaboration à distance. Ces espaces virtuels partagés permettent aux utilisateurs de se connecter et d'interagir au sein de mondes virtuels, créant ainsi des expériences immersives et collaboratives. Au cœur de notre recherche, notre objectif principal est de concevoir une architecture pour un EVC malléable, capable de s'adapter en temps réel aux besoins et aux comportements des utilisateurs.

Pour atteindre cette finalité, il est primordial de plonger en profondeur dans les fondements des EVC, leur mode de fonctionnement, leurs architectures et leurs possibilités d'interaction. Ainsi, ce chapitre introductif vise à offrir une vue d'ensemble générale des EVC, soulignant leur utilité et leur pertinence dans notre vie quotidienne, tout en mettant en évidence la nécessité d'étudier ces environnements virtuels dans le contexte de notre recherche.

Une compréhension approfondie des EVC constitue la première étape indispensable

pour élaborer une architecture malléable qui réponde aux exigences des utilisateurs et qui puisse s'adapter aux différentes situations. Par conséquent, nous explorerons différentes dimensions des EVC, depuis les définitions fondamentales de la réalité virtuelle jusqu'aux modèles de présence et aux architectures réseau. Nous examinerons également les interactions au sein des EVC, en étudiant de près les exemples concrets d'environnements virtuels collaboratifs existants.

## 1.2 La réalité virtuelle

### 1.2.1 Bref Historique

La réalité virtuelle a commencé à être explorée dès les années 1950, mais c'est seulement dans les années 1980 et 1990 qu'elle a commencé à prendre forme en tant que technologie [123]. Les premiers casques de réalité virtuelle (RV) ont fait leur apparition à la fin des années 1980, mais ils étaient peu pratiques et peu accessibles pour le grand public [23]. Au début des années 2000, les smartphones et la puissance de traitement accrue ont permis une révolution de la RV, offrant une expérience plus immersive et accessible pour les utilisateurs [125]. Depuis, la RV a continué à évoluer et est devenue un outil couramment utilisé dans divers domaines tels que les jeux vidéo, la formation, la simulation, la médecine, etc.

Les systèmes de réalité virtuelle ont fait leur apparition dans les années 50, bien avant même l'émergence du concept de réalité virtuelle [47]. En 1956, Morton Heiling a présenté le système Sensorama [56], offrant aux utilisateurs la possibilité de s'immerger auditivement et visuellement dans une scène préenregistrée (Figure 1.1). En 1964, Ivan Sutherland a proposé un autre système appelé Sketchpad, considéré comme l'un des premiers à contribuer au développement des techniques de dessin et de conception assistés par ordinateur (CAO) [130]. Plus tard, en collaboration avec son étudiant Bob Sproull, Ivan a développé le premier casque de réalité virtuelle, connu sous le nom de "Ultimate Display" [131]. Dans

les années 80, Jaron Lanier a introduit l'expression anglaise "Virtual Reality". Cette expression a été créée à des fins de marketing et de publicité, mais elle est maintenant couramment utilisée [139].



Figure 1.1: Le premier système de réalité virtuelle(extrait de [56]).

## 1.2.2 Définitions

La réalité virtuelle (RV) est un domaine de recherche qui englobe plusieurs disciplines telles que les sciences humaines, les sciences et techniques, l'informatique, l'infographie, la robotique et les interactions homme-machine (IHM). Au fil des années, la RV s'est également diversifiée dans d'autres domaines tels que la médecine, le militaire et l'éducation, ce qui lui permet d'établir sa légitimité en tant que domaine de recherche à part entière bénéficiant de cette multidisciplinarité.

D'un point de vue fonctionnel, la réalité virtuelle (RV) peut être définie comme une simulation d'un environnement 3D spécifique dans lequel les utilisateurs peuvent interagir avec des objets virtuels. Ci-dessous, nous présentons différentes utilisations de la RV :

- Une simulation réaliste d'un monde virtuel qui permet aux utilisateurs d'effectuer des actions impossibles à réaliser dans la réalité, comme le pilotage d'avions. Elle peut également être utilisée pour des situations dangereuses qui nécessitent une formation préalable avant de les réaliser dans la réalité, comme les opérations chirurgicales.
- Des applications de RV utilisées pour comprendre la réaction sensorielle des êtres humains face à des phénomènes réels, en étudiant la situation mentale de l'utilisateur.
- Un monde artificiel imaginé par des artistes, offrant de nouvelles expériences qui n'existent pas dans la réalité, telles que le traitement avec des personnages imaginaires.
- Un exemple concret de l'utilisation de la RV dans le domaine de la rééducation est l'utilisation de simulations virtuelles pour permettre aux patients de s'entraîner dans des environnements virtuels et motivants, afin d'améliorer leurs capacités physiques et de faciliter leur rééducation. Ces simulations permettent de reproduire des situations spécifiques et d'offrir des possibilités d'interaction qui ne seraient pas réalisables dans un environnement réel, contribuant ainsi à accélérer et optimiser le processus de rééducation.

D'un point de vue technique, la RV est une application 3D qui permet à l'utilisateur de se sentir immergé dans un monde virtuel et d'interagir avec celui-ci. La définition de la RV repose sur trois notions clés : l'immersion, la présence, et l'interaction.

### 1.2.2.1 L'immersion

L'immersion en réalité virtuelle (RV) se réfère au degré de présence et de sensation d'être physiquement présent dans un monde virtuel. Cela est rendu possible grâce à l'utilisation de dispositifs de RV tels que des casques, des gants ou des contrôleurs qui permettent à l'utilisateur d'interagir avec l'environnement virtuel. L'objectif de la RV est de créer une illusion d'être réellement présent dans un monde virtuel, offrant ainsi une

expérience de présence et d'immersion qui ne peut être reproduite par d'autres formes de médias. Plusieurs facteurs influencent le degré d'immersion en RV, notamment la qualité de l'affichage, la précision des interactions gestuelles, la qualité du son et l'ergonomie de l'interface utilisateur. Plus l'immersion est forte, plus l'utilisateur ressent sa présence dans le monde virtuel et plus l'expérience est immersive.

### 1.2.2.2 La présence

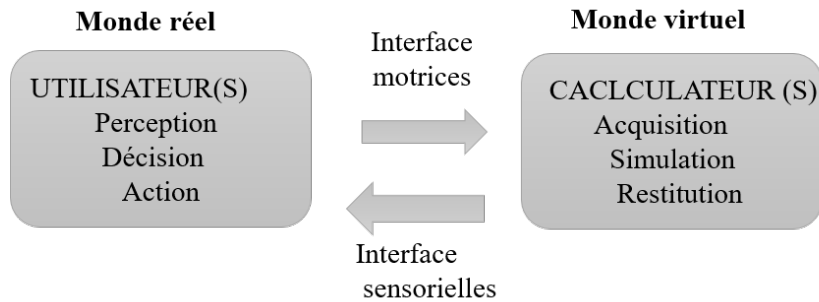
Contrairement à l'immersion, qui est principalement liée aux aspects techniques du système, le sentiment de présence se réfère à l'expérience subjective de l'utilisateur au sein de l'environnement virtuel (EV). Selon Slater et al. [123], le sentiment de présence est un phénomène subjectif qui englobe la sensation d'être réellement présent et "là" dans l'EV. La présence, également connue sous le nom de téléprésence, désigne les caractéristiques de l'état psychologique de l'expérience de l'utilisateur dans l'environnement virtuel (EV).

La présence dans un environnement virtuel (EV) dépend de processus cognitifs, de la perception de l'environnement et des émotions de l'utilisateur lors de son interaction. L'intensité de la présence influence la sensation d'immersion de l'utilisateur dans le monde virtuel, ce qui peut avoir un impact positif sur l'expérience globale. La présence est largement reconnue comme un aspect crucial dans des domaines tels que les jeux, la formation, la simulation et la thérapie en réalité virtuelle. Cette notion de présence dans les environnements virtuels collaboratifs sera explorée plus en détail dans la section suivante.

### 1.2.2.3 L'interaction

L'interaction 3D joue un rôle essentiel en tant que langage de communication entre l'homme et la machine. Ce langage permet à l'utilisateur de modifier son environnement virtuel en utilisant différentes fonctionnalités et d'obtenir des retours grâce à des interfaces sensorielles, motrices et diverses techniques d'interaction [47]. Selon Fuch [47], l'utilisateur agit sur l'environnement virtuel en utilisant des interfaces motrices qui captent ses actions telles que les gestes et la voix. Ces actions sont transmises à un ordinateur qui

interprète les modifications de l'environnement virtuel en fonction des actions de l'utilisateur. L'ordinateur évalue les transformations et les restitutions à apporter à l'utilisateur via les interfaces sensorielles (voir la figure 1.2).



**Figure 1.2:** Schéma d'interaction entre l'homme et la réalité virtuelle[47].

L'utilisateur devient alors un acteur actif du système plutôt qu'un simple spectateur. L'interaction joue un rôle crucial dans la différence entre un système de réalité virtuelle et une animation. Le temps est un facteur essentiel dans la conception des techniques d'interaction. Pour que l'interaction soit efficace et confortable, il est primordial que l'utilisateur puisse interagir en temps réel avec le système de réalité virtuelle sans ressentir de délai entre ses actions réelles et les répercussions dans le monde virtuel via les interfaces sensorielles. Réduire au maximum le décalage entre les actions réelles de l'utilisateur et les retours sensoriels est une contrainte complexe à respecter. Cependant, les chercheurs accordent une grande importance à cette contrainte lors du développement des environnements virtuels, car son non-respect peut influencer la qualité de l'immersion des utilisateurs dans l'environnement virtuel [24].

L'interaction en réalité virtuelle fait référence à la capacité d'un utilisateur d'agir et d'interagir avec les objets et les environnements virtuels de manière à générer une réponse réaliste et cohérente. Cela englobe des actions telles que le déplacement d'objets, l'exécution d'actions comme la saisie d'objets, la navigation et la communication avec d'autres utilisateurs. L'interaction peut être réalisée à l'aide d'équipements tels que des contrôleurs VR, des casques VR, des gants VR, etc. Une interaction réussie en réalité virtuelle peut



renforcer l'immersion et la sensation de présence de l'utilisateur dans le monde virtuel. De plus, l'interaction peut contribuer à offrir une expérience plus authentique et captivante pour les utilisateurs. Les développeurs en réalité virtuelle cherchent souvent à améliorer les niveaux d'interaction afin de renforcer l'expérience utilisateur.

## 1.3 Les Environnements virtuels collaboratifs

### 1.3.1 Définition

L'apparition des environnements virtuels collaboratifs (EVC) est une réponse au constat que les environnements virtuels destinés à un seul utilisateur ne peuvent pas répondre aux situations nécessitant une collaboration entre plusieurs utilisateurs, tels que les systèmes de réalité virtuelle utilisés dans la formation où un apprenant interagit avec un formateur. La recherche sur les environnements virtuels et le travail collaboratif assisté par ordinateur (TCAO) est étroitement liée aux EVC, car les chercheurs s'efforcent d'appliquer les concepts du TCAO et d'intégrer les principes fondamentaux de la réalité virtuelle afin de créer un système de réalité virtuelle qui favorise la collaboration entre les individus. Un EVC dispose d'un espace partagé entre les utilisateurs, leur permettant de communiquer, d'interagir entre eux et avec les éléments de l'environnement virtuel, de se toucher et de collaborer, même s'ils se trouvent à distance géographique [30]. Les EVC se trouvent à la convergence de différentes disciplines, telles que la réalité virtuelle, la psychologie cognitive, les interfaces homme-machine (IHM), l'ergonomie, la sociologie, le design, l'intelligence artificielle, la robotique, etc.

Un environnement virtuel collaboratif (EVC) permet à plusieurs utilisateurs de partager un même espace virtuel. En plus des interactions homme-machine des environnements virtuels, telles que la navigation, la sélection, la manipulation et le contrôle des applications, un EVC offre la possibilité aux utilisateurs de communiquer et d'interagir de manière synchronisée et collaborative, que ce soit en étant physiquement proches ou

à distance. Dans un EVC, les actions de chaque utilisateur sont instantanément transmises aux autres utilisateurs, chacun possède son propre environnement virtuel et peut être représenté visuellement par des avatars.

Cela est souvent nécessaire pour des tâches qui ne peuvent être accomplies individuellement, comme la co-manipulation ou la co-planification, par exemple. Ainsi, toutes les actions d'un utilisateur dans l'environnement virtuel sont transmises aux autres utilisateurs qui les perçoivent instantanément. Cela permet à chaque utilisateur de voir ses partenaires et de suivre leurs actions dans l'environnement virtuel. L'objectif fondamental de la conception des EVC est d'améliorer les interactions collaboratives à travers les outils informatiques. Grâce aux nouvelles possibilités d'interaction offertes par ces systèmes, la recherche dans ce domaine a connu une forte expansion tant au sein des institutions académiques que dans l'industrie [68].

## 1.4 Architectures des environnements virtuels collaboratifs

Dans un environnement virtuel collaboratif distribué, on trouve généralement plusieurs ordinateurs interconnectés, souvent géographiquement dispersés, qui sont désignés comme des "nœuds". Ces nœuds communiquent entre eux sur le réseau à l'aide de trois principales méthodes de transmission des données (voir Figure 1.3) :

- *Unicast* : Il s'agit d'une transmission de données d'un nœud à un autre, établissant une connexion point-à-point.
- *Broadcast* : Cette méthode permet à un nœud de transmettre des données à tous les autres nœuds du réseau simultanément.
- *Multicast* : Cette méthode permet à un nœud de transmettre des données à un sous-ensemble spécifique de nœuds du réseau.

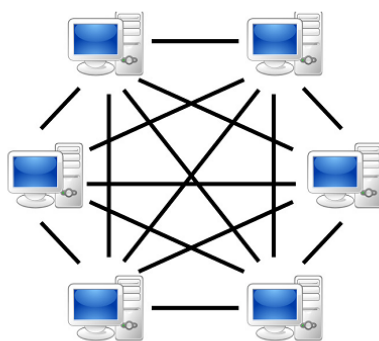


**Figure 1.3:** *Méthodes de communication réseau*

Les auteurs de l'article [34] font une distinction entre deux types d'organisations d'ordinateurs pour les environnements virtuels distribués : les architectures pair-à-pair et les architectures client-serveur. De plus, [83] introduit une troisième catégorie, à savoir les architectures hybrides, qui combinent les deux premières. Cette classification est également adoptée par [46].

#### 1.4.1 Architecture pair-à-pair

Les nœuds des utilisateurs évoluant dans l'environnement virtuel transmettent directement les informations entre eux (voir Figure 1.4). Certains systèmes utilisant cette architecture incluent MRTToolkit, SIMNET, NPSNET et SPIN-3D. Cependant, lorsque le nombre d'utilisateurs augmente considérablement, le nombre de messages à transmettre sur le réseau peut devenir très important, en particulier avec un mode de transmission unicast.



**Figure 1.4:** *Architecture pair-à-pair.*

Par exemple, si le système comprend  $N$  utilisateurs, un nœud doit envoyer  $N-1$  messages pour transmettre une mise à jour. De plus, il peut être difficile d'établir rapidement une communication avec tous les nœuds participant à la session collaborative, ce qui complique la maintenance de la cohérence entre les différents nœuds.

### 1.4.2 Architecture client-serveur

Dans cette architecture, il existe un accès central auquel tous les utilisateurs se connectent (voir Figure 1.5). Ce serveur central joue le rôle de médiateur pour la communication entre les utilisateurs et stocke les données nécessaires. Des exemples d'architectures client-serveur comprennent les systèmes VISTEL et ShareX3D. Cependant, lorsque le nombre d'utilisateurs augmente considérablement, un goulot d'étranglement peut se former au niveau du serveur en raison du grand nombre de requêtes entrantes [46].

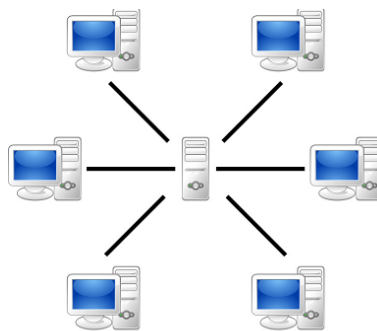
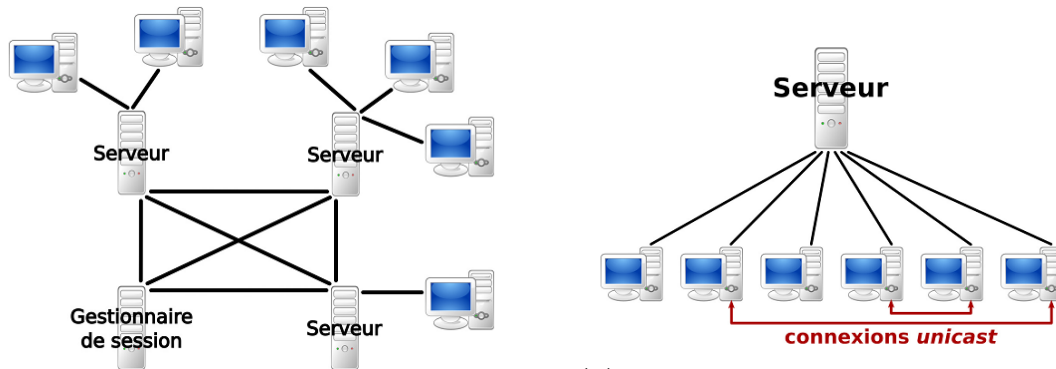


Figure 1.5: *Architecture client-serveur.*

### 1.4.3 Architecture hybride

Afin de surmonter les limitations des deux types d'architectures mentionnés précédemment, de plus en plus de systèmes optent pour des architectures hybrides qui combinent à la fois des communications entre les nœuds et des communications avec un serveur. Cette approche permet, par exemple, d'accélérer les communications entre les utilisateurs tout en facilitant le maintien de la cohérence de l'environnement virtuel grâce à l'interven-

tion du serveur. Les Figures (Figure 1.6a et 1.6b) illustrent deux exemples d'architectures hybrides. La première présente plusieurs serveurs qui répartissent les flux d'informations en utilisant des connexions unicast entre les serveurs, tandis que les utilisateurs sont connectés à un seul serveur qui gère uniquement les connexions. Cette architecture permet d'éviter les goulots d'étranglement rencontrés dans l'architecture client-serveur, tout en permettant de connecter des utilisateurs ayant des besoins spécifiques.



(a) Plusieurs serveurs connectent un environnement virtuel.

(b) Plusieurs connexions unicast supplémentaires sont établies entre les utilisateurs qui collaborent ensemble.

**Figure 1.6:** Architecture hybride.

La deuxième figure propose une autre architecture hybride conçue pour faciliter la collaboration entre des utilisateurs se trouvant à proximité les uns des autres dans le monde virtuel. Tous les utilisateurs sont connectés au même serveur. Cependant, lorsque des utilisateurs se trouvent à proximité dans le monde virtuel, des connexions unicast sont établies entre eux afin d'améliorer la cohérence de l'environnement virtuel entre ces utilisateurs, facilitant ainsi leur collaboration [46].

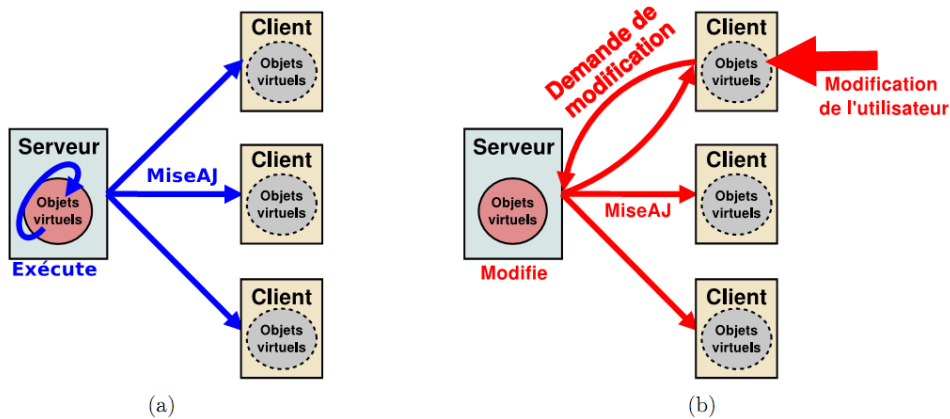
## 1.5 Distribution des données

Selon [46], lors de la conception d'un environnement virtuel collaboratif, une décision cruciale concerne la localisation des données liées à l'environnement virtuel et à ses objets. Il est essentiel de déterminer quels nœuds doivent stocker les données (telles que les

données géométriques, les textures, etc.) et quels nœuds doivent exécuter les processus associés à chaque objet. Trois modes de distribution des données sont généralement identifiés : le mode centralisé, le mode homogène répliqué et le mode partiellement répliqué.

### 1.5.1 Mode centralisé

Certains systèmes, comme VISTEL [150], adoptent une approche centralisée où toutes les données de l'environnement virtuel collaboratif sont stockées sur un serveur central, et tous les traitements liés aux objets sont exécutés sur ce même serveur (voir Figure 1.7(a)).



**Figure 1.7:** (a) L'exécution et (b) la modification du comportement des objets en mode centralisé (extrait de [46]).

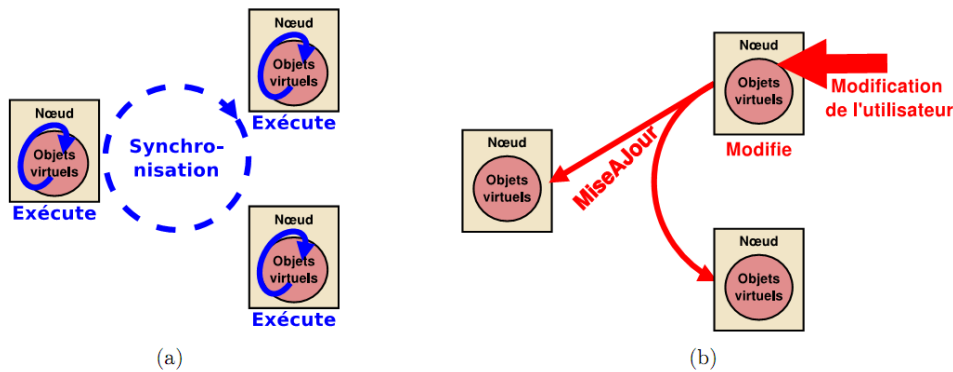
Les utilisateurs se connectent ensuite à ce serveur pour participer à la session collaborative, ce qui nécessite une architecture réseau de type client-serveur. Lorsqu'un utilisateur souhaite modifier l'état d'un objet, il envoie une requête au serveur, qui traite alors la demande de modification avant de transmettre une mise à jour à tous les nœuds, y compris celui qui a initié la requête (voir Figure 1.7(b)). Cette approche garantit une forte cohérence de l'environnement virtuel entre les différents nœuds et évite la duplication des données.

Cependant, cette méthode présente certains inconvénients majeurs. Tout d'abord, elle peut entraîner une latence importante entre les clients et le serveur, ce qui se traduit par un délai entre les actions des utilisateurs et les modifications effectives de l'environnement

virtuel par le serveur. De plus, le serveur risque de devenir un goulot d'étranglement car il doit envoyer des messages de mise à jour à tous les nœuds simultanément. Cette charge de travail intense peut affecter les performances du serveur et causer des retards dans la transmission des mises à jour aux utilisateurs.

### 1.5.2 Mode homogènement répliqué

La méthode la plus couramment utilisée dans les environnements virtuels collaboratifs consiste à initialiser l'état de chaque nœud participant à la session en utilisant une base de données commune contenant toutes les informations sur le monde virtuel, telles que le terrain, les modèles géométriques, les textures, les comportements des objets, etc. Cette approche est adoptée par des systèmes tels que MRTToolkit ou SPIN-3D. Les données peuvent être présentes sur le nœud de l'utilisateur lorsqu'il se connecte à la session (comme c'est le cas dans de nombreux jeux vidéo) ou être répliquées à partir des autres nœuds déjà présents dans la session ou à partir d'un serveur central. Pendant la session, chaque base de données évolue indépendamment sur les nœuds des utilisateurs, et tous les traitements relatifs aux objets sont exécutés localement (voir Figure 1.8(a)).



**Figure 1.8:** (a) L'exécution et (b) la modification du comportement des objets en mode homogènement répliqué (extrait de [46]).

Afin de maintenir la cohérence, seuls les changements d'état des objets et parfois des événements spécifiques tels que des collisions entre objets sont transmis sur le réseau, permettant à tous les utilisateurs de mettre à jour leurs bases de données respectives

(voir Figure 1.8(b)). Cette méthode présente deux avantages principaux : le nombre et la taille des messages transitant sur le réseau sont relativement faibles car seules les mises à jour sont envoyées, et la latence est très faible lors des interactions des utilisateurs car les objets manipulés sont traités localement sur leur propre nœud.

Cependant, cette méthode présente également quelques inconvénients. Tout d'abord, si le volume de données de l'environnement virtuel augmente, le volume de données de chaque nœud augmente également [7, 46]. De plus, cette approche est relativement inflexible, notamment si les utilisateurs souhaitent ajouter de nouveaux objets qui ne sont pas prévus dans la base de données initiale. La gestion des droits d'accès des utilisateurs aux objets peut également poser des difficultés, et l'environnement virtuel peut devenir incohérent en raison de la latence ou de la perte de certaines données lors des mises à jour sur le réseau. Enfin, cette méthode n'est pas adaptée à la co-manipulation, où plusieurs utilisateurs interagissent simultanément avec un même objet.

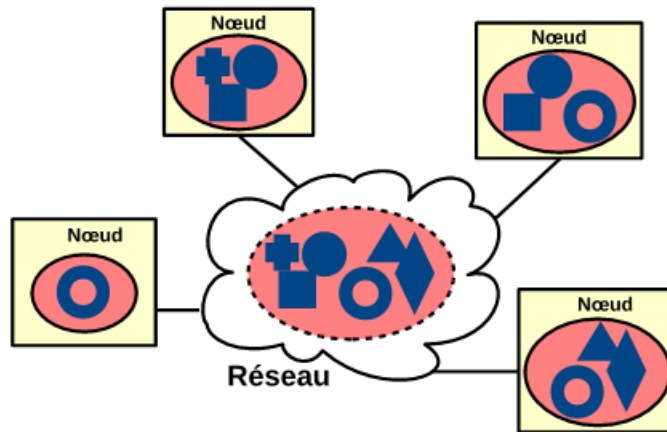
### 1.5.3 Mode partiellement répliqué

De nombreux systèmes de réalité virtuelle adoptent des approches hybrides qui combinent partiellement les avantages des modes précédemment présentés tout en atténuant certains de leurs inconvénients. Ces approches consistent à répartir les données ou les processus entre les différents nœuds, ce qui permet d'éviter une multiplication excessive des ressources requises et facilite le maintien de la cohérence de l'environnement virtuel. En utilisant cette approche hybride, il est possible d'optimiser l'utilisation des ressources et d'améliorer les performances globales du système.

Pour éviter la duplication de toutes les données sur les nœuds de chaque utilisateur, il est possible de répartir les données entre ces différents nœuds. Par exemple, dans le système DIVE, seuls les objets nécessaires à l'utilisateur sont copiés sur son nœud, tandis que les autres données restent sur le serveur (voir Figure 1.9). Cependant, si l'utilisateur a besoin d'autres objets au cours de la session, il devra les télécharger à nouveau de manière



dynamique. DIVE [26] utilise des connexions unicast pour gérer les communications entre les utilisateurs, permettant le transfert de messages ou d'objets lorsque cela est nécessaire. De plus, un serveur est utilisé pour suivre toutes les données présentes chez les différents utilisateurs, ce qui permet de sauvegarder l'état de l'environnement virtuel en cas de déconnexion brutale de certains utilisateurs ou de reprendre la session là où elle s'est arrêtée. Cette approche hybride permet d'optimiser l'utilisation des ressources tout en maintenant la cohérence de l'environnement virtuel.



**Figure 1.9:** Répartition partielle des données chez les utilisateurs (extrait de [46]).

Cette méthode présente l'avantage de pouvoir gérer un grand nombre d'utilisateurs ou un volume de données important sans avoir à les dupliquer pour chaque utilisateur. Cependant, elle présente également des inconvénients, tels que le coût élevé de communication pour transmettre les données supplémentaires et assurer la cohérence. Lorsqu'un objet change d'état, il est nécessaire de transmettre des messages de mise à jour à tous les autres utilisateurs, même s'ils n'ont pas cet objet. De plus, si le nombre d'utilisateurs est élevé, un utilisateur peut ne pas savoir quels autres utilisateurs possèdent l'objet qu'il vient de modifier [7, 46]. Ces aspects peuvent entraîner une surcharge du réseau et une complexité accrue dans la gestion de la cohérence de l'environnement virtuel.

## 1.6 Interactions dans les EVC

En Réalité Virtuelle, les utilisateurs ont la possibilité d'évoluer dans des environnements imaginaires ou simulés, qu'ils soient basés sur le monde réel ou complètement fictifs. L'interaction joue un rôle essentiel dans cette expérience, permettant aux utilisateurs d'accomplir différentes tâches et de modifier les événements qui se déroulent au fil du temps. Dans le contexte de la Réalité Virtuelle, une tâche peut être définie comme une action spécifique effectuée par l'utilisateur à un moment précis, et qui entraîne des changements dans l'état du monde virtuel. L'interaction en Réalité Virtuelle consiste donc à traduire les actions de l'utilisateur dans le monde réel en actions spécifiques qui ont un impact sur l'espace virtuel dans lequel il évolue.

### 1.6.1 Les tâches d'interaction 3D

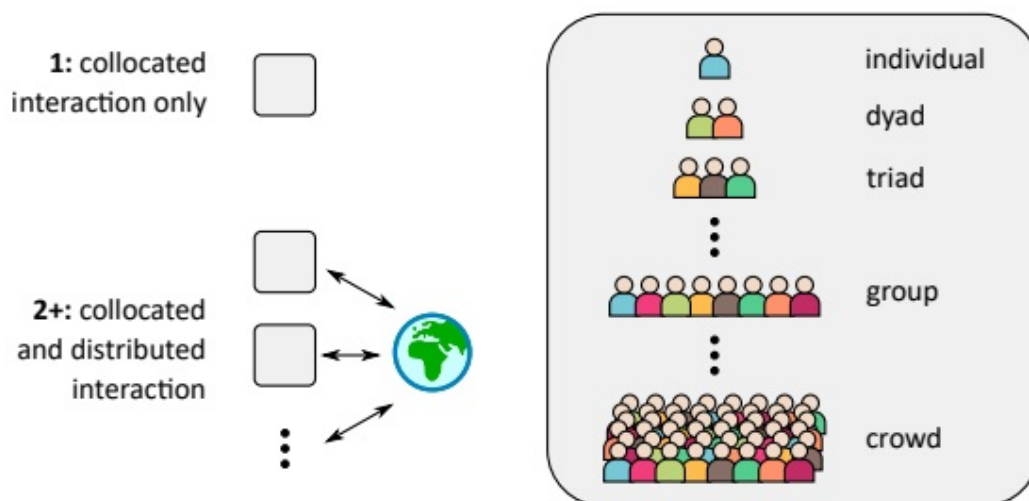
En 1999, Bowman a proposé une classification des techniques d'interaction en réalité virtuelle, basée sur quatre tâches principales : la navigation, la sélection, la manipulation et le contrôle d'application [22]. Cette classification permet de regrouper les différentes actions que les utilisateurs effectuent lors de leur interaction avec un environnement virtuel.

#### 1. La navigation

La navigation dans un environnement virtuel collaboratif implique le déplacement et l'exploration d'un espace virtuel partagé par plusieurs utilisateurs. Elle englobe différentes actions telles que se déplacer d'un endroit à un autre, explorer différentes zones, changer d'orientation et de position au sein de l'environnement virtuel. Pour effectuer la navigation, les utilisateurs peuvent utiliser des commandes de déplacement telles que la marche, la course ou le saut, ou utiliser des véhicules virtuels tels que des voitures, des avions, des bateaux, etc. De plus, des dispositifs de saisie spécifiques tels que des manettes, des joysticks, des contrôleurs de mouvement ou des casques de réalité virtuelle peuvent être

utilisés pour interagir avec l'environnement et se déplacer de manière immersive.

En 2021, Weissker et ses collègues ont proposé une classification des techniques de navigation en groupe dans la réalité virtuelle multi-utilisateurs [147]. Selon leur proposition, ces techniques sont classées en fonction du nombre d'espaces de travail distribués impliqués et du nombre d'utilisateurs colocalisés situés dans chacun de ces espaces, comme illustré dans la figure 1.10. Cette classification permet d'organiser et de catégoriser les différentes approches utilisées pour la navigation en groupe dans les environnements de réalité virtuelle, offrant ainsi un aperçu clair des possibilités et des variations de ces techniques en fonction de la configuration spatiale et de la distribution des utilisateurs.



**Figure 1.10:** Classification des techniques de navigation de groupe en réalité virtuelle [147].

- Interaction colocalisée : Cette catégorie comprend les situations où tous les utilisateurs sont présents dans le même espace de travail et interagissent uniquement les uns avec les autres dans cet espace. Il n'y a pas d'interaction entre les utilisateurs situés dans d'autres espaces de travail distribués ;
- Interaction distribuée : Cette catégorie implique plusieurs espaces de travail distribués où les utilisateurs interagissent uniquement dans leur espace respectif. Il n'y a pas d'interaction directe entre les utilisateurs situés dans différents espaces de travail distribués ;

- Interaction colocalisée et distribuée(hybride) : Dans cette catégorie, il y a un seul espace de travail principal où les utilisateurs colocalisés interagissent les uns avec les autres. Cependant, ces utilisateurs peuvent également interagir avec des utilisateurs situés dans d'autres espaces de travail distribués. Il y a donc une combinaison d'interaction colocalisée et d'interaction distribuée ;

## 2. La sélection

Dans les environnements virtuels, les utilisateurs interagissent avec les objets en utilisant leurs sens tels que la vue, le toucher et l'ouïe. L'une des actions les plus importantes est la sélection, qui consiste à choisir un ou plusieurs objets pour une utilisation ultérieure, notamment pour la manipulation. La tâche de sélection peut être subdivisée en deux sous-tâches distinctes.

La première sous-tâche est la désignation, qui correspond au choix initial d'un ou plusieurs objets dans le monde virtuel par l'utilisateur. Cela peut être réalisé en utilisant des techniques telles que le pointage, le suivi des mouvements ou l'utilisation d'interfaces tactiles. La deuxième sous-tâche est la validation, qui confirme la sélection effectuée. La validation peut être réalisée de différentes manières, en fonction des dispositifs et des interfaces utilisés. Par exemple, cela peut impliquer de presser un bouton physique ou virtuel, d'effectuer un geste spécifique, d'utiliser une commande vocale ou même de maintenir une certaine position pendant une durée spécifiée. En combinant la désignation et la validation, les utilisateurs sont en mesure de sélectionner les objets souhaités dans l'environnement virtuel et de les utiliser pour des manipulations ultérieures, telles que le déplacement, la modification ou l'interaction avec ces objets.

## 3. La manipulation

La manipulation d'un objet est une tâche courante dans le monde réel, et elle revêt également une grande importance dans les environnements virtuels (EV). Dans notre vie

quotidienne, nous effectuons des gestes complexes de manipulation d'objets sans y penser, comme maintenir un objet en équilibre ou le déplacer d'un endroit à un autre. Dans les EV, la manipulation joue un rôle crucial, car elle permet aux utilisateurs de modifier les propriétés d'un ou plusieurs objets virtuels.

La manipulation dans un EV permet aux utilisateurs d'interagir avec les objets de manière virtuelle, en leur donnant la possibilité de changer la position, l'orientation, la couleur, la taille ou d'autres caractéristiques des objets. Cela offre une plus grande flexibilité et une plus grande liberté pour créer et modifier des scènes virtuelles. Il convient de noter que la manipulation est étroitement liée à la sélection. En effet, avant de pouvoir manipuler un objet dans un EV, il est nécessaire de le sélectionner au préalable. La sélection permet à l'utilisateur de choisir l'objet sur lequel il souhaite effectuer des actions de manipulation. Une fois sélectionné, l'utilisateur peut ensuite appliquer des gestes ou des commandes spécifiques pour manipuler l'objet de la manière souhaitée.

#### **4. Le contrôle d'application**

Le contrôle d'application dans les environnements virtuels peut être réalisé de diverses manières en utilisant différentes techniques d'interaction. Parmi ces techniques, on retrouve la commande gestuelle, les menus graphiques 3D, la reconnaissance vocale ou de gestes, ainsi que l'utilisation d'outils de commande.

Les menus graphiques 3D sont une adaptation tridimensionnelle des menus 2D traditionnels. Ils peuvent prendre la forme d'un menu 2D transformé ou d'un menu entièrement en 3D. Les commandes peuvent être manipulées en utilisant des techniques de sélection et de manipulation 3D, permettant ainsi aux utilisateurs d'interagir avec les différentes options disponibles. La reconnaissance vocale offre la possibilité d'associer des mots ou des phrases à des commandes spécifiques. Cela permet aux utilisateurs de contrôler l'application ou l'environnement virtuel en utilisant leur voix. Cependant, la reconnaissance vocale peut nécessiter une phase d'apprentissage et ne garantit pas une efficacité optimale,

en particulier dans des environnements bruyants ou pour les utilisateurs ayant des accents ou des variations de prononciation. La reconnaissance de gestes permet aux utilisateurs de lancer des commandes en effectuant des mouvements naturels avec leur corps ou leurs membres. Cela permet une interaction intuitive et immersive avec l'environnement virtuel. Cependant, certaines commandes peuvent être difficiles à réaliser de manière précise et constante, nécessitant une certaine coordination et une pratique appropriée. L'utilisation de gants de données constitue une autre approche permettant de mesurer la flexion des doigts à l'aide de capteurs ou de boules réfléchissantes analysées par une caméra. Cette technique permet aux utilisateurs d'effectuer des gestes précis et détaillés avec leurs mains pour contrôler l'application ou l'environnement virtuel.

## 1.6.2 Les techniques d'interaction multi-utilisateurs

Les techniques d'interaction multi-utilisateurs dans les environnements virtuels permettent à plusieurs utilisateurs d'interagir simultanément et de manière collaborative. Il existe deux types principaux d'interaction multi-utilisateurs : synchrone et asynchrone.

### 1.6.2.1 Interaction asynchrone

La méthode de sélection/manipulation asynchrone a été utilisée dès les débuts de la recherche dans le domaine de la collaboration en environnement virtuel. Elle consiste à permettre à plusieurs utilisateurs d'interagir avec un objet partagé en utilisant des techniques d'interaction mono-utilisateur de manière séquentielle.

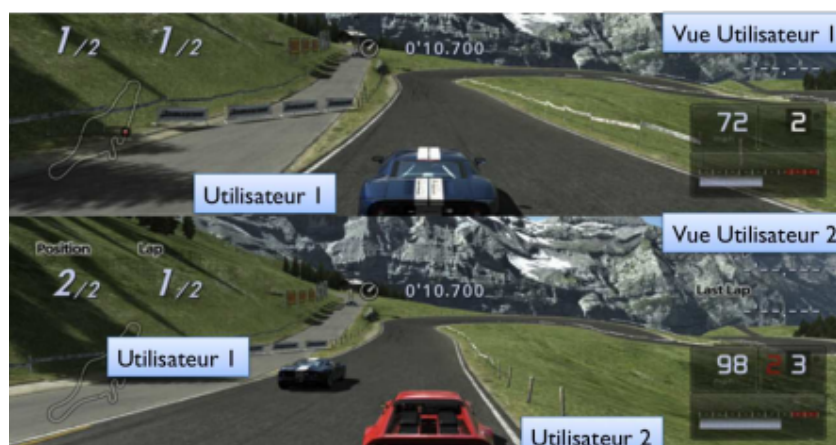
Un exemple de cette approche est le projet PIT (Protein Interactive Theater) [11], qui a permis à deux utilisateurs co-localisés et physiquement proches de partager et de manipuler un environnement virtuel commun pour l'étude des molécules. Chaque utilisateur disposait de sa propre vue de la scène et de ses propres données, car sa tête était suivie par des capteurs positionnés sur les lunettes stéréoscopiques. Cependant, les deux utilisateurs utilisaient un même dispositif d'interaction pour la manipulation des objets,

en utilisant une main virtuelle simple. Les actions sur l'objet étaient réalisées de manière séquentielle par les utilisateurs, qui se coordonnaient en temps réel en communiquant directement pour répartir les tâches. Ainsi, un utilisateur pouvait effectuer des manipulations sur l'objet pendant que l'autre observait et attendait son tour pour effectuer ses propres manipulations. Cette approche asynchrone permettait une collaboration efficace en évitant les conflits d'interaction et en facilitant la coordination entre les utilisateurs.



**Figure 1.11:** Aperçu d'un environnement virtuel collaboratif pour la visualisation de molécules (extrait de [11])

Stewart et al. ont proposé des travaux similaires dans le domaine de la collaboration en environnement virtuel [126]. Dans leur approche, les deux utilisateurs disposent chacun d'un dispositif d'interaction individuel, mais partagent le même dispositif de sortie, généralement un écran. Cette configuration est souvent utilisée dans les jeux vidéo multi-joueurs, où l'écran peut être divisé en deux parties pour que chaque joueur ait sa propre vue, ou bien les joueurs peuvent partager un même écran non divisé et visualiser les actions des autres participants.



**Figure 1.12:** Course à deux joueurs sur écran partagé dans Gran Turismo 5 : chaque joueur ne peut agir que sur son véhicule et peut communiquer en direct, sans influence sur le véhicule adverse.

### 1.6.2.2 Interaction synchrone

Les techniques synchrones permettent à plusieurs utilisateurs d'agir simultanément sur des données partagées de l'objet. Il y a deux catégories de techniques synchrones : la première consiste à répartir les données partagées entre les utilisateurs, en assignant par exemple à un utilisateur les translations et à l'autre les rotations de l'objet, via des techniques mono-utilisateur. La deuxième catégorie permet un accès concurrent aux mêmes données, ce qui permet de combiner les mouvements des différents utilisateurs pour déterminer le mouvement de l'objet. Ces techniques synchrones peuvent exploiter des techniques mono-utilisateur telles que le ray-casting ou la main virtuelle simple, ou des extensions de techniques qui ont pour base les techniques mono-utilisateur.

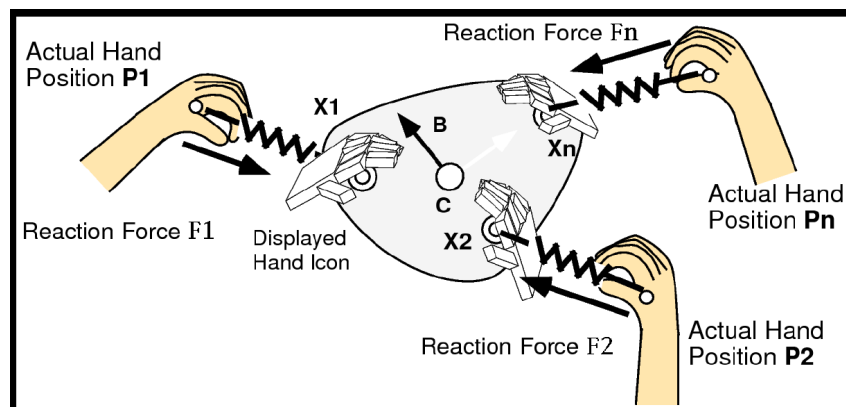
**a. Répartition des données partagées entre les utilisateurs** Dans l'approche, plusieurs utilisateurs peuvent simultanément agir sur un même objet en se voyant attribuer des tâches spécifiques de rotation et de translation. On distingue deux sous-catégories : les métaphores coopératives homogènes, où tous les utilisateurs utilisent la même technique d'interaction, et les métaphores coopératives hétérogènes, où les utilisateurs utilisent différentes techniques. Les chercheurs se concentrent sur le développement de méthodes visant



à maintenir la cohérence des actions tout en favorisant la coordination et la génération d'informations visuelles sur les actions des autres utilisateurs.

**b. Accès concurrent aux données partagées** Dans cette approche synchronisée de l'interaction multi-utilisateur, tous les utilisateurs ont un accès simultané aux données partagées, telles que les degrés de liberté, de l'objet. Lors de la manipulation de l'objet, son mouvement est déterminé en combinant les mouvements physiques des différents utilisateurs. Deux approches sont possibles : la première consiste à déterminer le mouvement de l'objet manipulé par deux utilisateurs en additionnant les vecteurs de position et d'orientation associés à chacun. La seconde approche consiste à prendre la moyenne de ces deux vecteurs.

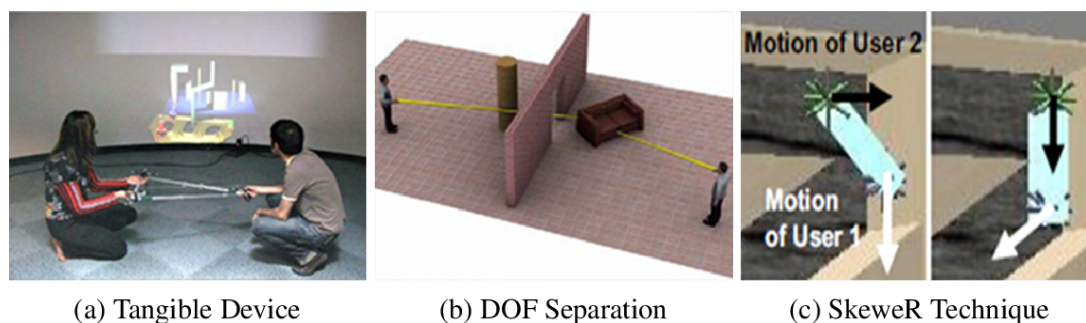
Noma et ses collègues ont présenté des travaux sur la manipulation d'objets par plusieurs utilisateurs [98]. Ces derniers interagissent avec l'objet partagé à travers des bras à retour d'efforts, comme illustré sur la figure 1.13. Chaque utilisateur est représenté par une main virtuelle simple, et le mouvement final de l'objet est le résultat de l'équilibre des forces appliquées par les utilisateurs. Cependant, cela peut engendrer des incohérences entre la position de la main virtuelle et la main réelle. En effet, lorsqu'un utilisateur applique une force à l'objet pour le déplacer, les mains virtuelles des autres participants attachées à l'objet suivent également le mouvement de l'objet, ce qui peut sembler incohérent pour les autres utilisateurs n'ayant pas actionné leur bras haptique. Différentes techniques ont été proposées pour permettre la manipulation simultanée d'un objet par plusieurs utilisateurs en environnement virtuel collaboratif. Parmi celles-ci, la technique des "3 Mains Virtuelles" [8] permet la détermination du mouvement de l'objet en utilisant trois points d'attache correspondant aux points de manipulation des mains virtuelles. Cependant, cette technique ne prend en compte que les translations.



**Figure 1.13:** Application de forces sur un objet virtuel à l'aide de simulations de ressorts virtuels [98].

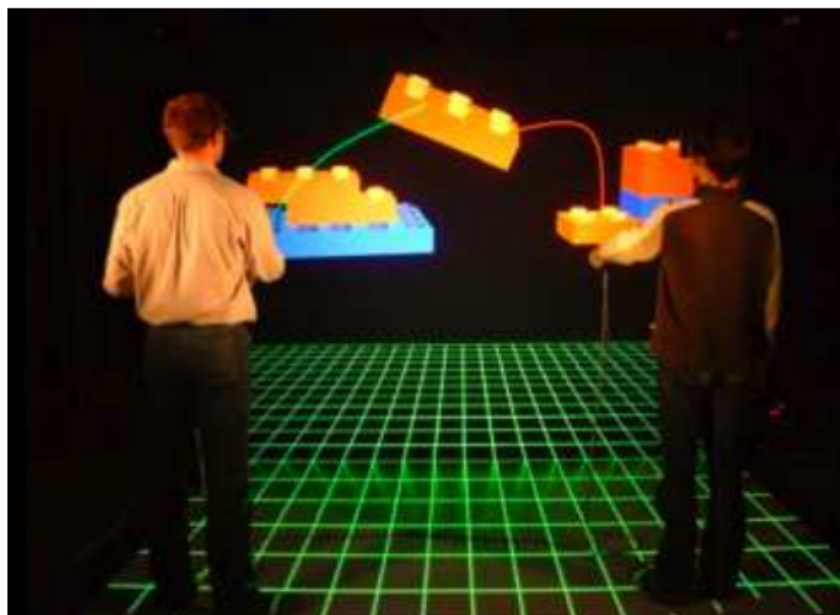
Une autre technique, appelée les rayons 2D/3D [39], permet une interaction 3D entre deux utilisateurs à l'aide d'un dispositif de pointage 2D, tel qu'une souris, sur un simple PC. Chaque utilisateur est représenté par un avatar immergé dans l'environnement virtuel, et l'objet 3D est contrôlé à partir de ce dispositif de pointage. Cette technique permet une collaboration asymétrique entre les utilisateurs.

Le curseur 3D, ou SkeweR [40], est une autre technique qui permet la manipulation simultanée d'un objet par plusieurs utilisateurs tout en conservant un historique des interactions. Chaque utilisateur contrôle un curseur 3D qui agit comme une main virtuelle simple et qui sélectionne automatiquement l'objet partagé lorsqu'il s'en approche. Si un seul utilisateur est disponible pour manipuler l'objet, seules les rotations peuvent être effectuées. Si deux utilisateurs sont disponibles, l'objet est contrôlé comme une "baguette" à partir des mouvements de translation et de rotation, comme illustré sur la figure 1.14.c.



**Figure 1.14:** Approches collaboratives pour la manipulation d'objets 3D dans des environnements virtuels(extrait de [40]).

La technique proposée par Duval et Tenier en 2006 est une adaptation de la méthode RayCasting pour la manipulation d'un objet par deux utilisateurs. Lorsqu'un utilisateur déplace son rayon pour déplacer l'objet partagé, le rayon de l'autre utilisateur attaché à cet objet se déplace également, mais sans déplacer l'effecteur du second utilisateur, ce qui crée une incohérence entre le mouvement réel et le mouvement virtuel. Pour remédier à cela, les auteurs proposent de courber les rayons en fonction de la force appliquée, en utilisant trois formes de rayons : élastique, coudé et déformable.



**Figure 1.15:** La technique de manipulation "Bent Pick Ray" (Extrait de [112])

Le Bent Pick Ray, présenté par Riege et al. [112], est une méthode similaire au rayon

virtuel proposé par Duval. Cette méthode utilise un rayon virtuel droit (pick ray) pour la sélection en mono-utilisateur. Lorsqu'un second utilisateur sélectionne également l'objet, un second rayon apparaît et les deux rayons se courbent (bent pick ray), comme illustré dans la figure 1.15.

Ces dernières années, de nouvelles techniques d'interaction multi-utilisateur ont émergé, comme celle proposée par [54], qui explore le potentiel de l'interaction multi-utilisateur en réalité virtuelle. Les auteurs abordent les défis liés à la création d'une expérience de réalité virtuelle multi-utilisateur et présentent FaceDisplay, un casque de réalité virtuelle modifié qui permet une interaction asymétrique entre les utilisateurs, comme illustré dans la figure 1.16. Ce casque est équipé de trois écrans tactiles et d'une caméra de profondeur qui permettent aux personnes environnantes de percevoir le monde virtuel à travers les écrans et d'interagir avec l'utilisateur du casque, que ce soit par le toucher ou par des gestes. Les auteurs discutent également des applications potentielles de cette technologie dans des domaines tels que les jeux, l'éducation ou les soins de santé.



**Figure 1.16:** illustration de casque FaceDisplay(extrait de [54]).

Une autre méthode de manipulation collaborative a été proposée par Wang et al en 2021 [146], visant à améliorer l'efficacité et la précision de la manipulation d'objets dans les applications de réalité virtuelle avec plusieurs utilisateurs. La méthode se base sur l'identification d'un manipulateur dominant en fonction de la qualité de point de vue, qui prend en compte la visibilité de l'objet à manipuler, la visibilité de la cible, ainsi que la profondeur et la distance de la cible. Un filtre temporel est utilisé pour rendre la détermination du manipulateur dominant plus stable. Cette technique se base sur la

qualité de point de vue . Un scénario est illustrée dans la figure 1.17 où deux utilisateurs manipulent un tuyau pour le faire correspondre à une cible. Les vues de l'utilisateur A et de l'utilisateur B sont représentées respectivement par (a) et (b). La troisième vue (c) est utilisée pour visualiser les positions des utilisateurs A et B. La boîte bleue indique le tuyau bleu en cours de manipulation, tandis que la boîte verte marque l'emplacement cible du tuyau. Grâce à cette méthode, la qualité du point de vue de l'utilisateur B est supérieure à celle de l'utilisateur A (0,69 contre 0,36). Autrement dit, l'utilisateur B est plus adapté pour manipuler les objets dans cette image, et donc, il est désigné comme le manipulateur dominant. Cette approche permet de sélectionner le manipulateur le plus efficace en fonction de la qualité de son point de vue, assurant ainsi une manipulation plus précise et efficiente de l'objet collaborativement.



**Figure 1.17:** Illustration de la technique de Manipulation collaborative basée sur la qualité du point de vue(extrait de [146]).

### 1.6.3 Conclusion intermédiaire

Les techniques d'interaction 3D collaboratives présentent plusieurs défis à relever. L'un des principaux défis est la coordination des actions entre les utilisateurs. Lorsque plusieurs utilisateurs interagissent simultanément dans un environnement virtuel, il est essentiel de garantir une coordination fluide et cohérente des actions. Cela nécessite la gestion des conflits potentiels entre les actions des utilisateurs, ainsi que la synchronisation des informations et des états des objets manipulés. Un autre défi majeur est la représentation et la visualisation des actions des différents utilisateurs. Il est crucial de fournir des indica-

tions claires et compréhensibles sur les actions effectuées par chaque utilisateur, afin de faciliter la collaboration et la coordination entre eux. Cela peut impliquer la conception d'interfaces intuitives, la mise en évidence visuelle des actions en cours, ou encore l'utilisation de représentations graphiques appropriées pour les objets manipulés. La gestion des problèmes de latence constitue également un défi important dans les environnements d'interaction 3D collaboratifs. Étant donné que les utilisateurs interagissent en temps réel, toute latence perçue peut compromettre la fluidité et la réactivité de la collaboration. Il est donc nécessaire de mettre en place des mécanismes de synchronisation efficaces et d'optimiser les performances du système pour minimiser les délais de traitement et de transmission des données. Enfin, la prise en compte des différences entre les utilisateurs, tels que leurs compétences, leurs préférences ou leurs capacités physiques, représente également un défi dans les techniques d'interaction 3D collaboratives. Il est important de développer des approches flexibles et adaptables qui permettent de prendre en compte ces différences afin de garantir une expérience utilisateur optimale pour tous les participants.

En résumé, les principaux défis des techniques d'interaction 3D collaboratives résident dans la coordination des actions, la représentation visuelle, la gestion de la latence et la prise en compte des différences entre les utilisateurs. La recherche et le développement continus dans ces domaines contribueront à améliorer l'efficacité et la convivialité des environnements d'interaction 3D collaboratifs.

## 1.7 Exemples d'environnements virtuels collaboratifs

L'évolution rapide de la technologie a ouvert de nouvelles perspectives passionnantes dans le domaine des environnements virtuels collaboratifs. Ces environnements offrent des possibilités de travail et d'interaction innovantes, permettant aux utilisateurs de collaborer et de communiquer à distance dans des mondes virtuels immersifs. Dans cette partie, nous explorerons quelques exemples d'environnements virtuels collaboratifs, mettant en évidence leurs fonctionnalités uniques et leur utilité dans différents domaines. Que ce soit

pour la formation, la conception collaborative, les jeux en ligne ou la simulation.

### 1.7.1 FréjusVR : simulateur de réalité virtuelle pour la formation des pompiers

FrejusVR est un système de simulation de réalité virtuelle spécialement conçu pour la formation des pompiers à la lutte contre les incendies dans les tunnels routiers (voir figure 1.18). Le simulateur permet aux utilisateurs de jouer différents rôles, tels que conducteur de voiture, passager de voiture, conducteur de camion et pompiers, et de suivre les procédures de sécurité en cas d'incendie. Le simulateur utilise des modèles physiques pour simuler le feu et la fumée, ainsi que des interactions avec les objets dans le tunnel, tels que les extincteurs et les téléphones d'urgence. Les utilisateurs sont évalués sur leur capacité à suivre les procédures de sécurité et à effectuer les tâches assignées à leur rôle. Le simulateur est conçu pour être évolutif et peut être adapté pour inclure de nouveaux scénarios et rôles.

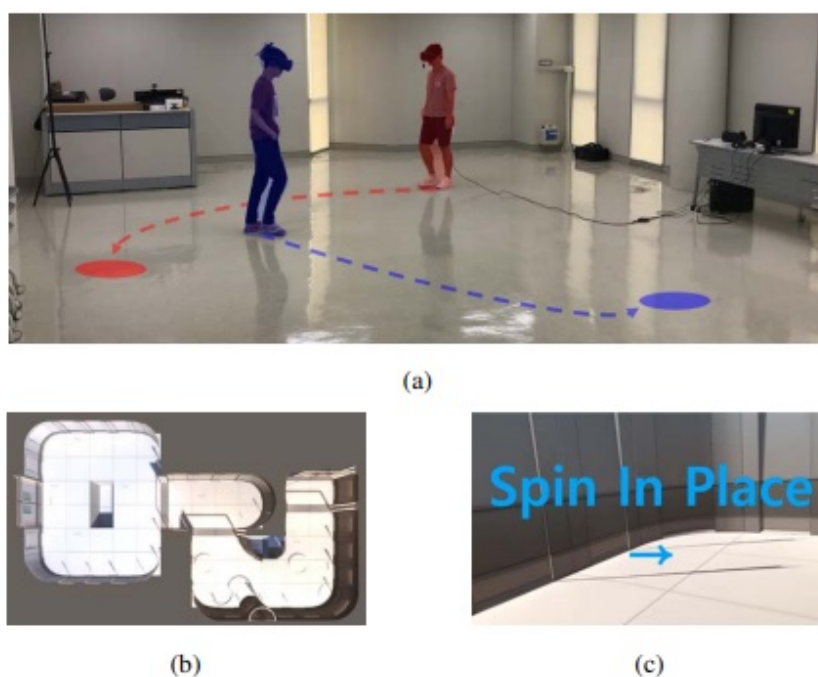


Figure 1.18: Illustration de jeu sérieux FrejusVR de formation(extrait de [25]).



## 1.7.2 MS2OT : Multi-Step State-Action-Value-based Optimized Trajectory

Le travail présenté par [84] présente une méthode d'apprentissage par renforcement pour la marche en réalité virtuelle appelée MS2OT (Multi-Step State-Action-Value-based Optimized Trajectory). Cette méthode utilise un réseau de neurones convolutifs en 3D pour extraire des caractéristiques de l'état actuel de l'utilisateur et prédire les actions de marche appropriées pour atteindre la cible virtuelle. La méthode utilise également des techniques de redirection pour éviter les collisions avec les murs physiques de l'environnement.



**Figure 1.19:** Système de redirection de marche en réalité virtuelle(extrait de [84]).

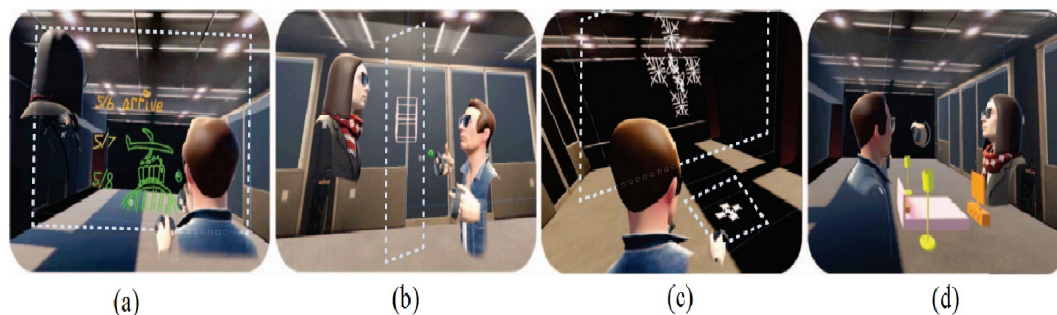
Les expériences menées ont comparé les performances de MS2OT avec celles de trois autres algorithmes de redirection de marche en réalité virtuelle. Les expériences ont été menées à la fois avec des utilisateurs réels et des utilisateurs simulés. Les résultats ont montré que MS2OT était plus efficace que les autres algorithmes pour la marche en réalité virtuelle avec plusieurs utilisateurs dans un espace physique. Cette approche de réedu-



cation malléable multi-utilisateurs représente une idée novatrice pour les travaux futurs dans le domaine. En adaptant les environnements virtuels et en développant des algorithmes de redirection de marche spécifiquement conçus pour la rééducation post-AVC, il est possible de créer un système interactif où plusieurs patients peuvent participer simultanément. Cette approche permettrait non seulement de favoriser la coopération et la socialisation entre les patients, mais aussi de recueillir des données précieuses sur les interactions entre les utilisateurs et d'identifier des stratégies de mouvement efficaces. En analysant ces données, il serait alors possible d'améliorer et d'optimiser les programmes de rééducation, offrant ainsi des résultats plus personnalisés et efficaces pour chaque patient. En explorant cette perspective de rééducation malléable multi-utilisateurs, cette approche pourrait ouvrir de nouvelles voies dans le domaine de la réadaptation post-AVC et contribuer à l'amélioration de la qualité de vie des patients.

### **1.7.3 CollaboVR :méta modèle d'un système de réalité virtuelle pour la collaboration créative**

CollaboVR représente une plateforme de réalité virtuelle reconfigurable qui combine différentes fonctionnalités telles que le dessin animé, l'édition collaborative de scènes et la communication multi-utilisateurs en temps réel. Quatre cas d'utilisation sont présentés avec des mises en page personnalisées : (a) une mise en page intégrée pour des réunions d'affaires(1.20.a), (b) une mise en page en miroir pour des présentations de cours de mathématiques(1.20.b), (c) une perspective à la troisième personne pour une mise en page projective permettant à l'utilisateur de dessiner avec ses mains et de projeter ses croquis aux participants distants sur un tableau interactif partagé(1.20.c), et (d) une mise en page montrant deux colocataires discutant de la conception d'un appartement. Pour des démonstrations en direct, veuillez vous référer à la vidéo supplémentaire(1.20.d).



**Figure 1.20:** Quatre exemples d'études de cas utilisant le framework CollaboVR(extrait de [55]).

Le système CollaboVR présenté dans l'article n'est pas initialement conçu pour être malléable, mais il existe des opportunités d'adaptation pour répondre aux besoins spécifiques de chaque étude de cas présentée. Dans le cas de la mise en page intégrée d'une réunion d'affaires, une amélioration potentielle pour la malléabilité serait de permettre aux utilisateurs de personnaliser la disposition des éléments sur l'écran en fonction de leurs préférences et de leur style de réunion spécifique. Par exemple, ils pourraient ajuster la taille et la position des fenêtres de discussion, des présentations ou des tableaux blancs virtuels pour optimiser leur flux de travail. Pour la mise en page en miroir d'une présentation de cours de mathématiques, une adaptation possible serait de fournir des options permettant aux enseignants et aux élèves de choisir le niveau de détail et les outils d'annotation disponibles pendant la présentation. Certains utilisateurs peuvent préférer une vue plus simplifiée pour faciliter la compréhension globale, tandis que d'autres peuvent avoir besoin d'un niveau de détail plus approfondi pour résoudre des problèmes mathématiques complexes.

## 1.8 Conclusion

D'après notre étude, il est évident que les recherches sur les environnements virtuels collaboratifs (EVC) se concentrent principalement sur la mise en œuvre des systèmes plutôt que sur leur conception. Il existe un manque d'efforts pour élaborer une architecture

ou un modèle de développement spécifiquement adapté à ce type d'environnement. De plus, les systèmes actuels sont souvent statiques, ce qui signifie qu'ils ne sont pas conçus pour s'ajuster de manière dynamique aux comportements des utilisateurs pendant leur utilisation. Même lorsqu'ils envisagent des systèmes personnalisés, ces derniers se fondent principalement sur le profil des utilisateurs, sans prendre en compte la situation de l'utilisateur pendant l'utilisation du système.

Dans la continuation de notre étude, nous examinerons de manière approfondie le concept de malléabilité et les termes associés, aussi bien dans les systèmes classiques que dans les systèmes en 3D. Notre objectif est de renforcer notre hypothèse selon laquelle les développeurs d'environnements virtuels collaboratifs (EVC) accordent peu d'attention aux aspects architecturaux et que le terme "malléabilité" n'est pas largement utilisé dans ce domaine. À cet égard, nous présenterons plusieurs outils et concepts qui fourniront une base solide pour établir un ensemble d'exigences à considérer lors de la conception d'une architecture pour un collectif dans un environnement virtuel collaboratif. Ainsi, nous pourrions envisager la création d'un environnement virtuel collaboratif malléable, capable de s'adapter en temps réel au contexte des utilisateurs.

# Chapitre 2

## La malléabilité des Environnements virtuels collaboratifs

### 2.1 Introduction

Nous avons mené cette recherche en constatant le manque d'études sur le concept de malléabilité dans le domaine de la réalité virtuelle, ce qui a suscité notre motivation à approfondir ce sujet. Pour combler cette lacune, nous avons entrepris une recherche approfondie en utilisant une combinaison de termes et de mots-clés provenant de différents domaines, tels que la réalité virtuelle (VR), les environnements virtuels collaboratifs (EVC), les collecticiels, les interfaces 3D, ainsi que les concepts de malléabilité, d'adaptation, de sensibilité au contexte et de plasticité 3D. Ces termes ont été soigneusement sélectionnés après une analyse préliminaire de la littérature existante. La figure 2.1 illustre notre méthodologie de recherche, dont l'objectif est d'explorer les concepts de malléabilité, d'adaptabilité et de sensibilité au contexte dans le cadre des Environnements Virtuels (EV) et des Environnements Virtuels Collaboratifs (EVC), ainsi que dans les collecticiels et les interfaces 3D. Nous incluons également le concept de plasticité 3D. En combinant ces termes, notre but est d'identifier des travaux de recherche pertinents traitant de la conception et du développement d'environnements virtuels et collaboratifs capables de

s'adapter dynamiquement aux besoins des utilisateurs et à leur contexte d'utilisation. Les combinaisons de termes en gras dans la figure 2.1 représentent ceux qui ont été identifiés comme pertinents dans la littérature.

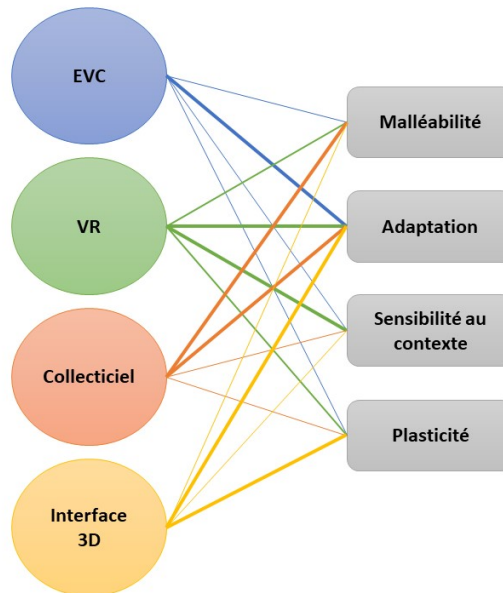


Figure 2.1: Notre méthodologie de recherche.

La figure 2.1 peut être représentée par la requête suivante :

```
"(EV OR EVC OR collectifiel OR interface3D) AND
(malléable OR malléables OR malléabilité) AND
(adapte OR adapter OR adaptation OR adaptive OR adaptable)AND
(sensibilité au contexte OR sensible au contexte) AND
(plasticité OR plastique OR plasticité 3D)"
```

Dans ce chapitre, nous avons commencé par examiner les définitions des concepts de malléabilité, de plasticité 3D, d'adaptation et de sensibilité au contexte. En combinant ces définitions avec les exigences de la réalité virtuelle et le modèle de trèfle fonctionnel de la collaboration, nous avons défini un ensemble d'exigences pour la malléabilité dans les EVC. Ainsi, nous avons identifié trois dimensions essentielles d'un environnement virtuel

collaboratif malléable : la réalité virtuelle, la malléabilité et la collaboration. Ensuite, nous nous sommes intéressés aux systèmes sensibles au contexte dans le contexte des systèmes de réalité virtuelle classiques, étant donné que nous avons constaté qu'ils répondaient le mieux à nos exigences. Nous avons examiné en détail ces systèmes sensibles au contexte et exploré leur pertinence pour notre étude.

## 2.2 La malléabilité

Selon Pinkwart et al. [107], la malléabilité des collecticiels peut prendre différentes formes, telles que la personnalisation, la composition et l'intégration. Cette capacité de modification pour répondre aux changements contextuels des utilisateurs en collaboration est une caractéristique essentielle d'un collecticiel malléable. Kahler [69] identifie trois autres caractéristiques importantes pour un tel logiciel : la prise en compte de diversités multidimensionnelles pour supporter différents usages, la capacité à s'adapter aux changements dynamiques du travail individuel et organisationnel, et la nécessité d'utiliser des méthodes alternatives pour accomplir les tâches en raison de l'incertitude et de l'ambiguïté du travail collaboratif. La malléabilité est la capacité à intégrer ou composer de nouveaux services sans interruption du système en cours d'exécution [17, 28]. Preguiça [109] souligne également l'importance de la malléabilité pour les collecticiels modernes et flexibles, qui doivent être capables de s'adapter et de se configurer de manière dynamique.

Bien que le terme "malléabilité" soit peu fréquent dans les études des vingt dernières années, il est utilisé dans le contexte des logiciels collaboratifs reposant sur les fondements théoriques du travail de groupe assisté par ordinateur-TCAO (Computer Supported Cooperative Work - CSCW). Cependant, peu d'études ont été menées sur l'utilisation de ce terme avec les interfaces 3D, les environnements virtuels (EV) et les environnements virtuels collaboratifs (EVC) [60].

Il est observé que la notion de "malléabilité" est abondamment traitée dans la littérature concernant les groupwares. Les groupwares, également appelés collecticiels, désignent

des logiciels permettant à des groupes de personnes de partager des documents à distance afin de faciliter la collaboration. En résumé, la malléabilité est une caractéristique essentielle pour les collecticiels, et étant donné que les environnements virtuels collaboratifs (EVCs) sont considérés comme des collecticiels 3D, la malléabilité revêt également une importance capitale dans ces systèmes.

## 2.3 La plasticité 3D

Les premiers à avoir introduit la notion de plasticité dans les systèmes interactifs sont Thevenin et Coutaz. Ils ont défini un système interactif plastique comme étant capable de résister aux divers changements des propriétés physiques du système et de l'environnement pendant son utilisation [138]. Selon eux, l'interopérabilité est une condition nécessaire, mais pas suffisante pour la plasticité. Cela signifie qu'un système doit être capable de s'exécuter sur toutes les plateformes considérées.

En 2007, Jameson a classé la plasticité comme un type d'adaptation, en distinguant deux types d'adaptation : l'adaptation au profil de l'utilisateur et la plasticité, qui consiste à s'adapter à l'environnement dans lequel l'interaction se situe [65]. Selon Sottet [124], la plasticité est une solution idéale pour développer des interfaces utilisateur (UI) capables de s'adapter dynamiquement à une variété de plates-formes telles que les PDA, les PC, les téléphones, etc. Auparavant, les développeurs devaient développer des UI pour chaque plate-forme, tandis qu'aujourd'hui, les UI plastiques peuvent facilement être redistribuées d'un dispositif à un autre. Taleb [134] considère que la plasticité est le noyau de la sensibilisation au contexte au niveau du système, qui maintient le système utilisable dans des scénarios divers. Lacoche [79] affirme qu'une interface 3D plastique est une interface qui peut s'adapter automatiquement à une série de changements et de contraintes du système et de l'environnement.

D'après Laroche [80], une interface-utilisateur plastique peut être classifiée selon l'une des dimensions suivantes : cibles, temps ou sources, et est ainsi considérée comme une

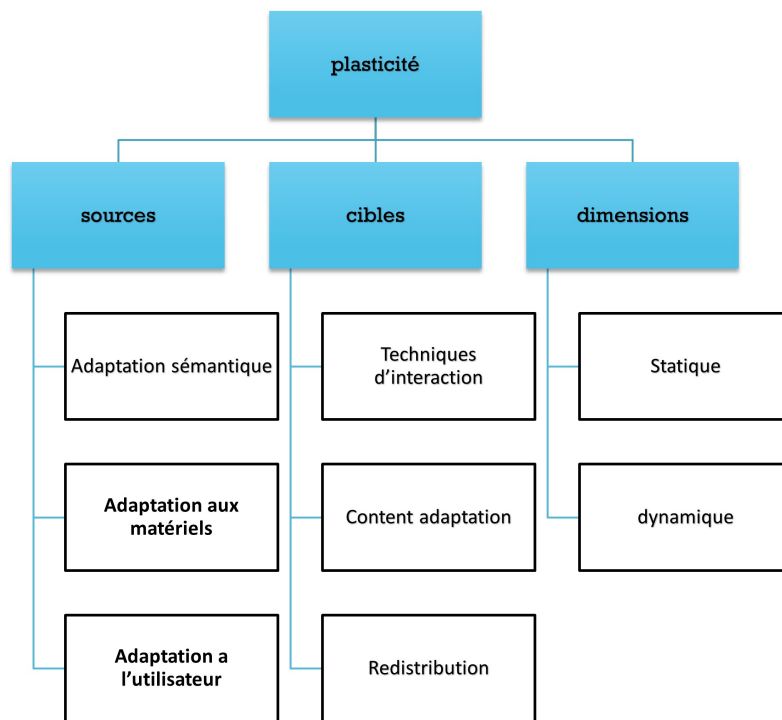


Figure 2.2: Les dimensions de la plasticité 3D[80].

interface-utilisateur plastique.

**a. Les sources d'adaptation :** Les sources d'adaptation font référence aux entités ciblées par le processus d'adaptation. Trois exemples courants de ces entités sont les caractéristiques de l'utilisateur, de l'environnement et du matériel. Ces entités sont définies en fonction du contexte, qui peut être considéré comme toute information pertinente pour caractériser la situation impliquant l'interaction entre un utilisateur et une application, qu'il s'agisse d'une personne, d'un lieu ou d'un objet.

**b. Les cibles d'adaptation :** Les cibles d'adaptation se rapportent aux différents changements effectués par le système et les composants logiciels impliqués dans le processus d'adaptation. Ils peuvent concerner le contenu de l'application ou les techniques d'interaction, par exemple. En somme, un mécanisme d'adaptation prend en compte des sources pour produire une adaptation sur des cibles spécifiques au sein du système. Ces cibles sont modifiées de manière à répondre aux exigences du mécanisme de plasticité.



**b. La dimension temporelle de l'adaptation :** Le mécanisme d'adaptation peut être divisé en deux catégories : statique et dynamique. L'adaptation statique se produit avant le déploiement de l'application, ce qui nécessite l'arrêt de l'application et sa reconfiguration. Cependant, ce type d'adaptation est limité en raison de la variabilité des contraintes d'adaptation. Dans un environnement ubiquitaire où les situations changent fréquemment, il est souvent impossible d'arrêter et de reconfigurer l'application à chaque adaptation. En revanche, l'adaptation dynamique permet la modification du comportement d'un logiciel pendant son exécution. Pour cela, l'application doit être construite sur une plateforme capable de supporter ce type d'adaptation. L'adaptation dynamique peut entraîner un blocage plus ou moins important de l'application. Dans les cas extrêmes, l'application peut être temporairement indisponible pendant la phase d'adaptation, ce qui peut être problématique pour le bon fonctionnement du système. Il est donc préférable de bloquer uniquement les parties spécifiques ciblées par l'adaptation.

Nous constatons que le terme "plasticité" est rarement abordé dans les interfaces classiques, ainsi que dans les interfaces axées sur l'adaptation des interfaces 3D et des environnements virtuels, malgré son importance en tant que caractéristique des systèmes adaptatifs. De plus, la plasticité est souvent associée à l'adaptation de l'interface en fonction des dispositifs et des situations de l'utilisateur, dans le but de créer une interface adaptative capable de s'ajuster à différentes plateformes existantes.

## 2.4 La RV et la sensibilité au contexte

Moon [94] souligne que la sensibilité au contexte dans les systèmes VR consiste à transférer des informations du monde réel de l'utilisateur vers l'ordinateur sous forme d'informations contextuelles, afin d'utiliser ces données pour fournir des informations ou des services qui répondent aux besoins des utilisateurs. Selon cet auteur, la création d'opportunités de personnalisation des expériences de réalité virtuelle exige un degré plus élevé de sensibilité au contexte. Par exemple, les jeux sérieux de réalité virtuelle pour

la rééducation nécessitent une surveillance et un suivi précis des mouvements du patient pour évaluer s'il exécute correctement les mouvements requis [149].

En raison du manque d'études théoriques sur la sensibilité au contexte dans la réalité virtuelle, voire la plupart des applications de réalité virtuelle sensibles au contexte se concentrent davantage sur le développement que sur la conception. Par conséquent, nous avons examiné les systèmes classiques sensibles au contexte. Selon Paspallis et al. [102], un système sensible au contexte est un domaine qui met l'accent sur les outils et les méthodes utilisés pour la capture, la modélisation et enfin l'exploitation des informations contextuelles. Ces informations peuvent inclure toute information qui concerne l'interaction d'un utilisateur avec un système, comme l'adresse GPS de l'utilisateur, le temps, l'activité de l'utilisateur, les conditions de bruit ou d'éclairage, etc. Dey[36] a introduit la notion de sensibilité au contexte comme étant un système qui utilise le contexte pour offrir des informations et/ou des services pertinents à l'utilisateur, dont l'adaptation dépend de la tâche de l'utilisateur. Ryan et al. [50] soulignent que la sensibilité au contexte est la capacité de capturer, interpréter et répondre aux caractéristiques de l'environnement, de l'utilisateur et de l'interface. Selon Abowd et al. et Dey [6, 33], un système est sensible au contexte s'il passe par ces quatre étapes : acquisition des informations contextuelles, modélisation du contexte, raisonnement et distribution, afin de permettre à un système d'offrir des services et des informations adaptables à l'utilisateur. Miraoui et Tadj [92] considèrent qu'un système est sensible au contexte s'il peut changer automatiquement ces formes ou déclencher un service comme réponse au changement de la valeur d'une information ou d'un ensemble d'informations qui caractérisent l'utilisateur et l'environnement. Selon Rohn [113], une application sensible au contexte se distingue d'une application classique en raison de ses caractéristiques suivantes :

- Adaptative : l'application s'adapte aux préférences des utilisateurs ;
- Réactive : l'application anticipe les besoins de l'utilisateur dans un environnement instable ;
- Pro-active : l'application est dirigée par des objectifs, capables de prendre l'initia-

tive plutôt que de simplement réagir à l'environnement ;

- Autonome : l'application est indépendante et peut agir sans intervention humaine.

En outre, Pascoe [101] définit une série de capacités de base qui peuvent servir de vocabulaire pour décrire et identifier un système sensible au contexte :

- Détection contextuelle : un système sensible au contexte doit être capable d'identifier les différents états environnementaux et de les présenter aux utilisateurs ;
- Adaptation contextuelle : dans ce type de système, en plus de l'intérêt de l'utilisateur pour les données contextuelles, les applications doivent également utiliser ces informations contextuelles pour adapter leur comportement afin de s'intégrer de manière plus transparente dans l'environnement de l'utilisateur ;
- Détection de ressources contextuelles : par l'exploitation de ces données, le système peut détecter, dans le même contexte, d'autres ressources que lui et utiliser ces ressources tout en restant dans le même contexte ;
- Augmentation contextuelle : l'environnement est renforcé avec des informations complémentaires en étendant la capacité du système à détecter, intervenir et interagir avec l'environnement.

## 2.5 Adaptation des environnements virtuels

Un système adaptable est un système capable de s'ajuster et de s'adapter automatiquement aux changements de son environnement, aux préférences et aux besoins de ses utilisateurs, afin de fournir une expérience utilisateur optimale. Les systèmes adaptatifs peuvent être divisés en deux catégories : les systèmes avec interfaces-utilisateurs adaptables et les systèmes avec auto-configuration des interfaces-utilisateurs [87].

### 2.5.1 Classification des systèmes adaptatifs

La classification des systèmes adaptatifs, présentée par Lindt [87], repose sur deux critères : le temps de compilation et le temps réel. En ce qui concerne le temps de compilation,

deux types d'interfaces-utilisateurs sont identifiés : les interfaces-utilisateurs configurables et les interfaces-utilisateurs auto-configurables. Les interfaces-utilisateurs configurables permettent à l'utilisateur d'ajuster les paramètres de l'interface avant d'utiliser le système, tandis que les interfaces-utilisateurs auto-configurables adaptent automatiquement l'interface-utilisateur en fonction des préférences et des besoins de l'utilisateur. Pour ce qui est du temps réel, il existe également deux types d'interfaces-utilisateurs : les interfaces-utilisateurs adaptables et les interfaces-utilisateurs adaptifs. Les interfaces-utilisateurs adaptables permettent à l'utilisateur de modifier l'interface pendant son utilisation, tandis que les interfaces-utilisateurs adaptifs s'adaptent automatiquement aux besoins et aux préférences de l'utilisateur en temps réel. Cette classification permet de mieux comprendre les différents types de systèmes adaptatifs et de choisir celui qui convient le mieux aux besoins de l'utilisateur.

Dans le domaine des environnements virtuels de formation, l'adaptabilité et la personnalisation sont devenues des sujets clés en raison de la diversité des utilisateurs et des domaines d'application. Les chercheurs se sont intéressés à rendre ces environnements plus flexibles en offrant aux utilisateurs la possibilité d'apporter des modifications pour répondre à leurs besoins. Différentes définitions de l'adaptation ont été proposées. Selon Streicher et al. [128], l'adaptation se réfère à la personnalisation des expériences en fonction des besoins individuels des utilisateurs et de leur contexte de travail actuel. Zahabi et al. [151] ont identifié trois composants essentiels pour qu'un système de formation soit adaptatif : les mesures de performance du stagiaire, les variables adaptatives et la logique d'adaptation. Kelley [70] a classifié les fonctionnalités adaptables en deux catégories : le contenu de simulation adaptatif et la rétroaction adaptative, en se focalisant sur l'adaptation du contenu de la simulation, du stress, des caractéristiques physiques, des éléments contrôlés, du contrôle du stagiaire, des caractéristiques d'affichage, de la difficulté du scénario d'entraînement et de la charge de la tâche secondaire. Gerbaud et al. [48] ont proposé une classification en deux catégories : le paramétrage, qui concerne l'adaptation des paramètres de la simulation avant l'exécution de la session de formation,

et l'adaptation dynamique, qui se produit pendant l'exécution de la formation et où le contenu s'adapte dynamiquement en fonction des mesures de performance en temps réel. Ces classifications mettent en évidence l'importance de l'adaptation du contenu et de la rétroaction pour améliorer l'apprentissage dans les environnements virtuels de formation.

### 2.5.2 Architecture d'un système de réalité virtuelle adaptatif de formation

Zahabi et al. [151] ont adopté la même classification que Gerbaud [48], mais ils ont souligné l'importance d'adapter également la rétroaction en fonction des différentes mesures de performance pour optimiser l'apprentissage. Dans leur travail [151], les auteurs ont

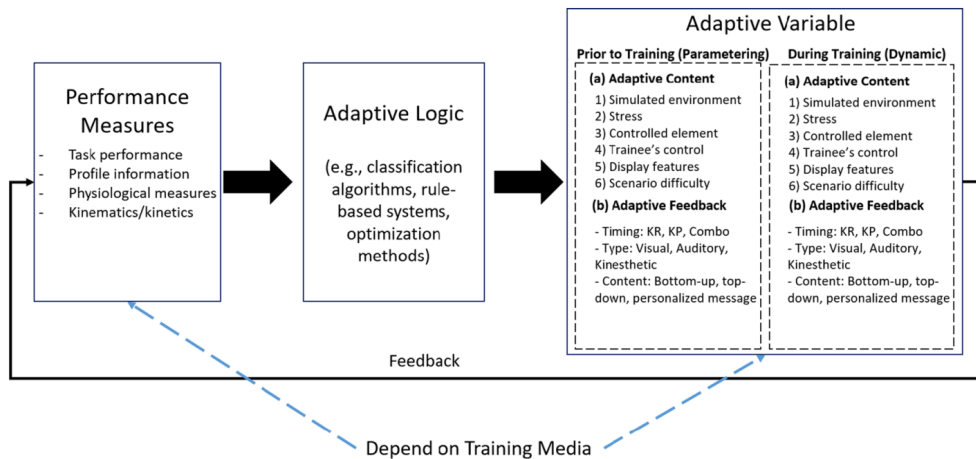


Figure 2.3: Architecture pour les systèmes VR adaptatifs de formation. [151].

proposé un cadre pour la formation adaptative basée sur la réalité virtuelle. Ils ont souligné l'importance de collecter des mesures de performance, qui ont été obtenues à l'aide d'une combinaison de mesures hors ligne et en temps réel. Ces mesures comprenaient des informations sur le profil de l'apprenant, des mesures physiologiques, des mesures de performance de tâches et des mesures cinématiques.

Les auteurs ont également mentionné que le module de logique adaptative utiliserait des algorithmes de classification, des approches d'optimisation ou des systèmes basés sur

des règles pour ajuster le contenu de la simulation et la rétroaction en fonction des mesures de performance de l'apprenant. Ainsi, des adaptations peuvent être apportées avant ou pendant la session de formation, en se concentrant sur différents aspects tels que le contenu de la formation et la rétroaction. Il convient de noter que bien que ce cadre ait été conçu pour la formation basée sur la simulation en réalité virtuelle, sa structure générale peut également être appliquée à d'autres types de formations basées sur la simulation, comme la formation à la conduite. Cependant, il est possible que certaines adaptations ne soient pas pertinentes pour d'autres supports de formation.

En résumé, cette architecture la formation adaptative basée sur la réalité virtuelle est un outil précieux pour les instructeurs et les apprenants, car il permet d'adapter la formation de manière dynamique en fonction des besoins individuels de l'apprenant. En utilisant des mesures de performance et des algorithmes de logique adaptative, ce cadre offre un moyen efficace d'améliorer l'efficacité de la formation et de garantir que les apprenants ont la meilleure expérience possible. Cependant, cette architecture spécifique à un type particulier de système de réalité et ne tient pas compte de l'aspect collaboratif.

## 2.6 Synthèse

### 2.6.1 Contraintes et défis relevés

Le présent travail se concentre sur la conception de l'architecture d'un environnement virtuel collaboratif malléable (EVCM). Cependant, avant d'aborder cette tâche, il est crucial de fournir une définition claire de l'EVCM, ce qui s'est avéré difficile pour plusieurs raisons :

- Tout d'abord, il existe peu de recherches spécifiquement axées sur les EVCM, et les auteurs n'emploient pas fréquemment le terme "malléabilité" dans les environnements virtuels en général, ni dans les environnements virtuels collaboratifs en particulier. Ils préfèrent généralement les termes "adaptation" ou "personnalis-

tion".

- Ensuite, malgré nos recherches approfondies sur le terme "malléabilité" dans les systèmes classiques afin de l'adapter aux EVCM, nous n'avons pas trouvé de définition unifiée. Chaque auteur propose sa propre définition en fonction de ses besoins spécifiques.
- Enfin, nous avons également rencontré d'autres termes similaires à la malléabilité, tels que la sensibilité au contexte en RV, la plasticité 3D et l'adaptation. Toutefois, la difficulté de parvenir à une définition précise persiste, et nous n'avons pas réussi à trouver une définition satisfaisante répondant aux exigences du système que nous souhaitons développer.

Le sujet de thèse proposé aborde les aspects collaboratifs d'un EVCM ainsi que les problématiques liées à son adaptation en temps réel. Notre objectif principal est de proposer une plateforme de collaboration RV "malléable" capable de s'ajuster aux changements et aux diversités des besoins des utilisateurs. Par conséquent, le système requiert des exigences élevées.

### 2.6.2 Nouvelles exigences

Dans cette partie, nous présenterons les nouvelles exigences d'un système de réalité virtuelle collaboratif malléable, que nous considérons comme indispensables dans ce type de système. Ces exigences ont été établies à partir des études menées dans ce chapitre

1. Offrir une expérience immersive en réalité virtuelle : Le système de réalité virtuelle collaboratif malléable doit être compatible avec les dispositifs de réalité virtuelle tels que les casques VR, les contrôleurs de mouvement et les accessoires associés. Il doit permettre aux utilisateurs de s'immerger pleinement ou partiellement dans l'environnement virtuel, d'interagir de manière réaliste et intuitive avec les objets virtuels, favorisant ainsi la collaboration, l'engagement et l'interaction entre les utilisateurs.

2. Détecter tous les changements dans le système en identifiant les sources d'informations relatives à l'utilisateur et à l'environnement. Le système doit savoir où trouver ces informations en fonction des besoins.
3. Organiser, interpréter et exploiter les données capturées grâce à un gestionnaire d'adaptation dédié.
4. Déterminer le moment opportun pour s'adapter en mettant en place des déclencheurs automatiques après un événement spécifié.
5. Définir les variables adaptatives qui déterminent les aspects du système pouvant être modifiés pour assurer la malléabilité.
6. Contrôler les modifications proposées par le gestionnaire d'adaptation en déterminant qui a l'autorité pour les exécuter : le système, l'utilisateur ou les deux.
7. Représenter les informations relatives à la collaboration des utilisateurs, telles que leur rôle, leur groupe de travail, leur tâche en cours, et gérer les interactions 3D entre eux en intégrant de nouvelles techniques ou en combinant celles existantes. Un workflow dédié à la communication, à la production et à la coordination est nécessaire pour gérer ces aspects. Le système doit être capable de suivre l'état des autres utilisateurs, leurs actions, leurs interactions et leurs difficultés.

La plupart des termes utilisés dans la littérature ne définissent pas clairement ces exigences, et même ceux qui le font ne satisfont pas pleinement à toutes les exigences décrites ci-dessus, comme le montre le tableau comparatif présenté dans notre étude (voir Tableau 4.5). Dans ce tableau, nous avons recueilli plusieurs travaux qui ont défini ces termes, mais chacun les a définis selon son propre contexte. Afin de remédier à cette situation, notre étude se concentre sur la malléabilité des collecticiels et vise à fournir une référence complète pour l'adaptation, la sensibilité au contexte et la plasticité dans les environnements virtuels et les environnements virtuels collaboratifs (EVC). Ainsi, chaque utilisateur pourra utiliser cette contribution en fonction de ses besoins, en exploitant toutes les exigences ou seulement certaines d'entre elles.



Notion	Référence	Exg1	Exg2	Exg3	Exg4	Exg5	Exg6	Exg7
<i>Malléabilité</i>	[107]	-	-	+	-	-	-	+
	[69]	-	-	-	+	-	-	+
	[17]	+	-	-	+	+	-	+
	[28]	-	-	+	+	+	-	++
EV/EVC adaptatifs	[48]	+	+	-	+	+	-	-
	[128]	+	+	-	+	+	-	-
	[151]	+	+	++	+	++	-	-
VR sensible au contexte	[149]	+	+	+	-	-	-	-
	[117]	+	++	-	+	+	-	-
Plasticité 3D	[80]	+	+	-	+	+	+	-

**Tableau 2.1:** La compatibilité des définitions existantes avec nos exigences. "-" ne prend pas en compte l'exigence, "+" prend partiellement en compte l'exigence, "++" prend fortement en compte l'exigence.

Laroche et al.[80] ont défini un ensemble de dimensions de malléabilité, comme illustré dans la figure 2.2. Nous suivrons le même principe en décomposant nos exigences en trois classes. Chaque classe correspondra à une dimension majeure, à savoir la réalité virtuelle, la malléabilité et la collaboration. Ces dimensions constituent les principaux aspects à prendre en compte lors de la conception et de la réalisation d'un tel environnement.

### 1. Dimension de la collaboration :

- **Communication** : L'environnement virtuel collaboratif doit permettre la communication entre les utilisateurs, en prenant en charge les interactions vocales, écrites et visuelles. Il doit faciliter la transmission efficace des informations et la coordination des activités.

- **Coordination** : L'environnement virtuel doit faciliter la coordination des tâches et des actions entre les utilisateurs, en offrant des fonctionnalités telles que la planification, la répartition des rôles et la synchronisation des activités.

- **Production** : L'environnement virtuel collaboratif doit permettre la création et la manipulation d'objets virtuels, ainsi que la gestion des données et des ressources associées aux projets collaboratifs.

### 2. Dimension de la réalité virtuelle :

-**L'immersion** : Elle se réfère à la capacité de plonger les utilisateurs dans un

environnement virtuel de manière à ce qu'ils se sentent totalement immergés et engagés. Cela peut être réalisé grâce à des graphismes de haute qualité, des effets sonores immersifs, des dispositifs de suivi des mouvements précis et d'autres techniques qui contribuent à créer une illusion de présence dans le monde virtuel.

**-La présence :** Concerne la sensation subjective d'être réellement présent dans l'environnement virtuel. C'est cette impression que l'utilisateur ressent d'être physiquement présent et d'interagir avec l'environnement virtuel de manière naturelle. Une bonne présence permet de renforcer l'immersion et de rendre l'expérience plus réaliste et captivante.

**-L'interaction :** Un aspect crucial de la réalité virtuelle. Elle permet aux utilisateurs d'interagir avec les objets virtuels, de manipuler des éléments de l'environnement virtuel et d'effectuer des actions qui ont un impact sur leur expérience. L'interaction peut être réalisée à l'aide de dispositifs de contrôle spécifiques tels que des contrôleurs de mouvement, des gants haptiques ou d'autres interfaces utilisateur adaptées à la réalité virtuelle.

### 3. Dimension de la malléabilité :

- **Adaptation en temps réel :** L'environnement virtuel collaboratif doit être capable de détecter les changements dans le système et de s'adapter en conséquence. Il doit avoir des mécanismes pour déterminer quand s'adapter (**déclencheur**), où trouver les données nécessaires (**sources de données**), comment s'adapter (**l'approche d'adaptation**), quoi adapter (**Les variable d'adaptation**) et qui est chargé de prendre les décisions d'adaptation (**Le contrôleur**). Ces composants seront présentés en détail dans le chapitre 4.5.

Les dimensions mentionnées sont présentées dans la figure ci-dessous (figure 2.4). Dans notre contribution, ces trois dimensions sont cruciales pour la conception et le développement d'un environnement virtuel collaboratif malléable, afin de fournir une expérience

utilisateur immersive et adaptative. Tout au long du reste du manuscrit, nous les considérerons comme des dimensions de référence en nous efforçant de répondre à plusieurs questions essentielles. Comment pouvons-nous les interconnecter de manière efficace ? Quels sont les détails spécifiques de chaque composant de malléabilité ? De plus, nous présenterons une étude de cas concrète pour démontrer que l'architecture proposée est réalisable et applicable. Dans notre étude, nous nous intéresserons tant au volet conceptuel qu'à la mise en œuvre pratique, afin de ne pas rester dans un niveau abstrait.

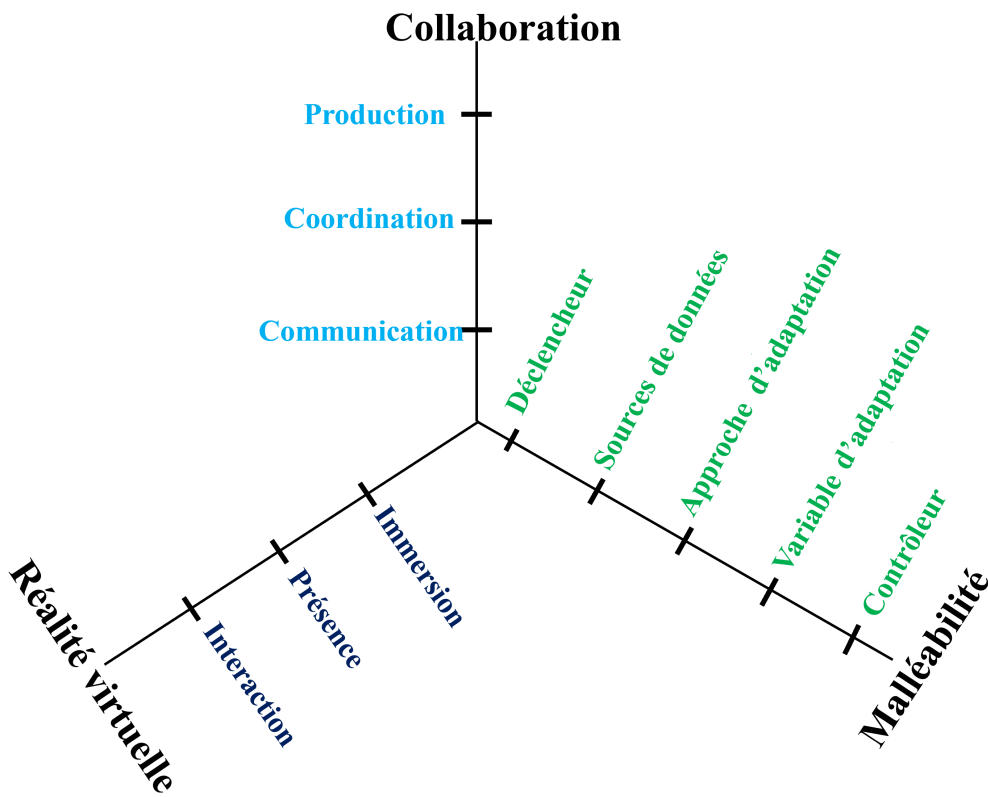


Figure 2.4: Les trois dimensions de notre système EVCM.

### 2.6.3 Notre démarche de recherche

Dans ce qui précède, nous avons effectué une analyse des termes et de leurs définitions existantes afin d'évaluer dans quelle mesure ils répondent à nos exigences. Dans la suite de notre recherche, nous approfondirons cette analyse en cherchant une architecture qui

pourra répondre, au moins partiellement, à nos attentes.

la plupart des travaux sur les systèmes de réalité virtuelle sensibles au contexte, adaptatifs ou plastiques négligent l'importance de l'aspect collaboratif. Pourtant, dans un environnement virtuel collaboratif, la collaboration entre utilisateurs joue un rôle essentiel. Afin de combler cette lacune, il est nécessaire d'approfondir l'étude des architectures des collecticiels afin de concevoir et mettre en œuvre des outils qui complètent les solutions existantes et répondent de manière adaptée à nos besoins conceptuels et pratiques. En intégrant de manière efficace les dimensions de collaboration, de la malléabilité, nous serons en mesure de créer des environnements virtuels collaboratifs offrant une expérience immersive et adaptative, tout en facilitant une interaction fluide entre les utilisateurs. L'étude des systèmes sensibles au contexte et des environnements virtuels collaboratifs (EVC) adaptatifs sont deux domaines de recherche distincts mais complémentaires. Cependant, notre étude sur les systèmes sensibles au contexte peut apporter des éléments nouveaux ou originaux par rapport aux EVC malléables, de plusieurs façons :

1. **Approche centrée sur l'utilisateur** : Les systèmes sensibles au contexte ont une approche centrée sur l'utilisateur, en prenant en compte les besoins, les préférences et les caractéristiques de chaque utilisateur. Cela permet de concevoir des systèmes plus personnalisés et adaptés aux besoins de chaque utilisateur, ce qui peut améliorer leur expérience et leur productivité. Cette approche peut compléter l'approche des EVC adaptatifs, qui se concentrent souvent sur l'adaptation de l'environnement en fonction des données d'utilisation, mais sans nécessairement prendre en compte les caractéristiques individuelles des utilisateurs.
2. **Prise en compte du contexte** : Les systèmes sensibles au contexte prennent en compte le contexte de l'utilisateur, c'est-à-dire les informations environnementales et situationnelles qui peuvent influencer son expérience d'utilisation. Cette prise en compte du contexte peut permettre de mieux comprendre les besoins de l'utilisateur et d'adapter le système en conséquence. Cette approche peut être utile pour

améliorer les EVC adaptatifs, en permettant une adaptation plus fine en fonction du contexte de l'utilisateur.

- 3. Intégration de différentes sources de données :** Les systèmes sensibles au contexte intègrent souvent différentes sources de données, telles que les données provenant de capteurs, de réseaux sociaux ou de dispositifs mobiles, pour mieux comprendre les besoins et les préférences des utilisateurs. Cette intégration de données peut également améliorer les EVC adaptatifs, en fournissant des informations plus précises sur l'utilisation du système et en permettant une adaptation plus fine en temps réel.

Fondamentalement, l'analyse des systèmes sensibles au contexte peut apporter des perspectives nouvelles et innovantes par rapport aux EVC adaptatifs. Cela permet une approche centrée sur l'utilisateur, une considération du contexte et une intégration de diverses sources de données. Une telle approche peut favoriser la conception de systèmes plus performants et personnalisés pour les utilisateurs d'EVC. Par la suite, nous approfondirons notre étude sur les systèmes sensibles au contexte, suivi d'un chapitre dédié aux collecticiels. Dans ce dernier chapitre de l'état de l'art, nous explorerons les concepts fondamentaux des collecticiels, leurs architectures, ainsi que les diverses approches existantes en matière de malléabilité des collecticiels.

## 2.7 Les systèmes sensibles au contexte

### 2.7.1 Le contexte dans les systèmes classiques

Dans le domaine de l'informatique, de nombreux chercheurs ont proposé des définitions du contexte. La première définition de contexte a été introduite en 1994 par Schilit et Theimer [118]. Selon eux, le contexte comprend l'emplacement et l'identité des entités (personnes ou objets) à proximité, ainsi que les changements qui peuvent affecter ces entités au fil du temps.

Chen [29] a défini le contexte comme étant l'ensemble des paramètres environnementaux susceptibles de déterminer le comportement de l'application ou dans lesquels un événement d'application se produit et est intéressant pour l'utilisateur.

Dans le domaine de l'informatique ubiquitaire, Miraoui [93] a défini le contexte comme étant toute information dont le changement de valeur déclenche un service ou modifie la qualité d'un service. Selon Chaari [27], le contexte englobe les paramètres externes à l'application qui sont en mesure d'influencer le comportement du système, en apportant de nouveaux aperçus et perspectives sur ses services et ses données. Enfin, Ghayam [50] a défini le contexte comme toute information qui identifie l'environnement dans lequel l'action se déroule à un moment précis. Il affirme que le contexte doit être lié à une action et doit être capable de contraindre la construction et/ou l'exécution d'une action.

Cependant, la définition la plus complète du contexte a été proposée par Dey [36], qui est adoptée par la majorité des chercheurs. Selon lui, le contexte est : *"toute information qui peut être utilisée pour caractériser la situation d'une entité. Une entité est une personne, un endroit ou un objet qui peut être considéré comme significatif pour l'interaction entre l'utilisateur et l'application, incluant l'utilisateur et l'application eux-mêmes."*

### 2.7.1.1 Caractéristiques d'un contexte

Dans ce qui suit, nous présentons les caractéristiques techniques de l'information contextuelle, telles que définies par [50] :

- L'information contextuelle est changeable avec le temps : la valeur de l'information de contexte se caractérise par le changement continu avec le temps. Par exemple dans un jeu sérieux de rééducation, la fatigue de l'utilisateur implique que le contexte situation psychologique (émotion, douleurs,...) change de valeur.
- L'information contextuelle est hétérogène : différents captures physique (Web Camera, la Kinect, ...) sont installés afin de collecter les différents informations contextuelles de l'environnement virtuelles, la variété des source de contexte a conduit à fournir des informations hétérogène.

- L'information contextuelle est imparfaite : la qualité et les caractéristiques de source d'information contextuelle détermine la qualité de l'information fournie. Ainsi, le contexte peut être incorrecte ou bien imprécis ou même inconnu.
- L'information contextuelle est interdépendante : Les informations contextuelles peuvent dépendre d'autres informations de contexte. La modification de la valeur des informations de contexte peut affecter une autre valeur de contexte.

### 2.7.1.2 Classification du contexte

Afin de faciliter la collecte et la présentation des informations contextuelles dans un système d'adaptation, différentes catégorisations du contexte ont été proposées par plusieurs chercheurs. Dey [33] classe le contexte en deux catégories :

- **Contexte primaire** qui englobe les informations sur l'identité, la localisation, l'activité et le temps ;
- **Contexte secondaire** qui peut être déduit du contexte primaire.

Une autre classification est présentée par Chen et al. [29], qui définissent deux types de contexte :

- **Contexte actif**, qui a une influence sur les comportements du système ;
- **Contexte passif** qui est pertinent mais n'a pas d'incidence sur l'application.

Razzaque [110] définit six catégories de contexte :

- **Contexte utilisateur** : représente le profil de l'utilisateur (identifications, listes de tâches, relations avec les autres, etc.).
- **Contexte physique** : définit l'environnement physique (température, humidité, éclairage, niveau de bruit, etc.).
- **Contexte du réseau** : répertorie l'environnement du réseau (connectivité, bande passante, protocole, etc.).
- **Contexte d'activité** : décrit le déroulement des événements dans l'environnement (entrée d'une personne, sortie, etc.).
- **Contexte matériel** : présente les caractéristiques et les activités des appareils

(résolution, précision, emplacement, durée de vie de la batterie, etc.).

- **Contexte de service** : définit les informations sur les fonctionnalités du système (affichage, format de fichier, etc.).

Le contexte est divisé selon Ghayam et al. [50] en quatre classes :

1. **Contexte utilisateur** : englobe les préférences de l'utilisateur (la résolution d'écran souhaitée, la langue, etc.).
2. **Contexte physique** : définit les paramètres physiques de l'environnement, du terminal (mémoire disponible, vitesse du processeur, etc.) et du réseau (bande passante, mode de connexion, etc.). De plus, il représente la mesure de l'environnement physique (humidité, température, niveau de bruit, etc.).
3. **Contexte spatio-temporel** : définit les informations de l'emplacement de l'utilisateur et de son voisinage. De plus, il garde une référence temporelle des actions courantes de l'utilisateur et en garde la trace.
4. **Contexte organisationnel** : représenter les informations relatives à la collaboration dans laquelle un utilisateur est engagé ( le groupe auquel il fait partie, le rôle, la tâche en cours, etc).

### 2.7.2 Le contexte dans les applications RV

La réalité virtuelle (RV) est un domaine où l'utilisation du contexte peut être bénéfique pour améliorer l'expérience utilisateur. En effet, le contexte peut aider à déterminer les actions et réactions de l'environnement virtuel en fonction des interactions de l'utilisateur, ce qui peut rendre l'expérience plus réaliste et immersive. Les capteurs peuvent détecter les mouvements de l'utilisateur et les traduire en actions dans l'environnement virtuel, tandis que les algorithmes de traitement du contexte peuvent prendre en compte des données telles que l'heure, la localisation géographique, les préférences de l'utilisateur, etc. pour personnaliser l'expérience.



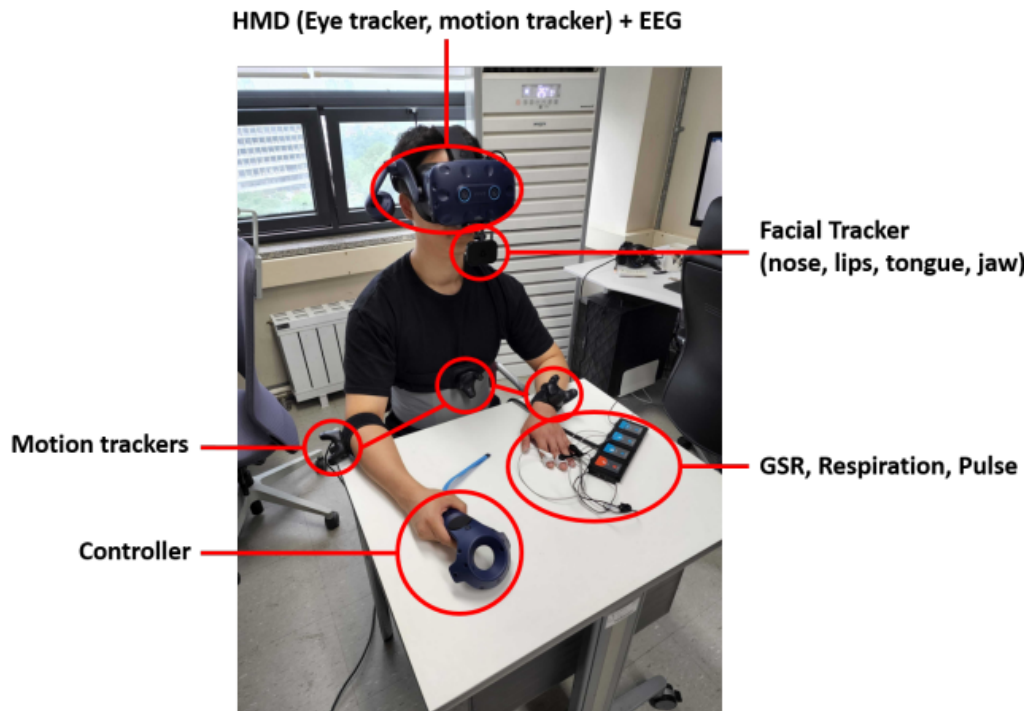
Cependant, il est important de souligner que l'utilisation du contexte dans les applications VR doit être effectuée de manière éthique et respectueuse de la vie privée de l'utilisateur. Les développeurs doivent veiller à ne pas utiliser la VR pour collecter ou suivre de manière abusive les données personnelles de l'utilisateur. Il est crucial de prendre en compte les implications éthiques de la collecte et de l'utilisation des données contextuelles pour garantir que les applications VR soient utilisées de manière responsable.

### 2.7.2.1 Capture des données contextuelles

Dans les environnements de réalité virtuelle (RV), les données sont utilisées à des fins d'analyse ou d'adaptation en temps réel. L'analyse fait référence à la capture et à la persistance des données RV pendant une session, qui sont ensuite utilisées pour fournir une expérience adaptative à l'utilisateur. Par exemple, ces données permettent de modifier dynamiquement la position ou la couleur d'objets, ou encore de générer des scénarios imprévus tels que des incendies, de la pluie, de l'obscurité ou des explosions. Pour réagir convenablement à chaque action de l'utilisateur, le système doit observer en temps réel toutes les actions effectuées et les exploiter.

Prenons l'exemple d'un simulateur de conduite de voiture en RV. Les informations sur l'emplacement, la vitesse et les conditions routières sont constamment capturées par le système RV, qui peut générer des accidents ou des dérapages incontrôlés en fonction des données collectées. Dans un autre exemple, un jeu sérieux de rééducation fonctionnelle en RV utilise des données sur l'état psychologique et physique du patient pour contrôler le niveau de difficulté des exercices dans le jeu.

Dans les simulateurs de formation en RV, la détection des mouvements (« body tracking ») est utilisée pour suivre tous les mouvements du corps, de la tête aux pieds. Parmi les dispositifs utilisés, les gants RV permettent de capturer la position exacte et la pression exercée par les doigts. La figure 2.5 montre quelques capteurs de données utilisés dans les environnements de réalité virtuelle.



**Figure 2.5:** Quelques capteurs utilisés dans les environnements virtuels [94].

La collecte de données provenant de différentes sources de capteurs est essentielle pour permettre une prise de conscience contextuelle approfondie de l'utilisateur dans les systèmes de réalité virtuelle sensibles au contexte. Cependant, la diversité des sources de données pose des défis en termes d'organisation, de transformation et d'utilisation de ces données. De nombreux travaux antérieurs se sont concentrés sur le développement de tels systèmes de réalité virtuelle sensibles au contexte, comme le soulignent Emmelkamp et al. [45], Seo et al. [119], Li et al. [86], Marin et al. [90], et Gradl et al. [52]. Cependant, ces travaux intègrent souvent étroitement la méthode de collecte de données à un projet de réalité virtuelle spécifique, ce qui rend difficile son adaptation à d'autres projets.

Pour remédier à ce problème, Moon [94] a proposé un framework de collecte de données contextuelles réutilisable et extensible appelé ManySense VR. Ce framework unifie la collecte de données provenant de différentes sources pour les applications de réalité virtuelle. Il est mis en œuvre à l'aide de gestionnaires de données contextuelles extensibles, qui permettent de collecter des données à partir de sources telles que le suivi oculaire,

l'électroencéphalogramme, le pouls, la respiration, la réponse galvanique de la peau, le suivi facial et la carte météo. L'approche de ManySense VR offre une solution pratique et flexible pour la collecte de données contextuelles dans divers projets de réalité virtuelle, offrant ainsi de nouvelles opportunités pour une compréhension plus approfondie de l'utilisateur et une meilleure adaptation de l'expérience virtuelle.

### 2.7.2.2 Classification du contexte

Moon [94] propose une classification de cinq types de données qui peuvent être capturées et utilisées dans les systèmes de réalité virtuelle sensibles au contexte.

#### 1. Les données oculaires

incluent des informations telles que l'ouverture et la taille de la pupille, le clignement des yeux, la position des yeux, l'expression faciale et la cible du regard. Le suivi oculaire (eye tracking en anglais) permet de suivre en temps réel l'attention visuelle des participants pendant leur expérience en réalité virtuelle. L'utilisation du suivi oculaire dans un environnement de réalité virtuelle permet de mieux comprendre comment les utilisateurs se déplacent et interagissent avec leur environnement virtuel, ainsi que de déterminer ce sur quoi ils focalisent leur attention.

Le suivi oculaire est utilisé dans divers domaines de la réalité virtuelle, notamment les jeux vidéo pour activer des pouvoirs surhumains et des interactions basées sur le regard, la simulation et la formation pour améliorer l'entraînement et l'évaluation des performances grâce aux données de suivi oculaire, ainsi que la télé-présence et la socialisation en réalité virtuelle en permettant des mouvements oculaires expressifs.

Un exemple d'utilisation des données oculaires est fourni par l'entreprise tobiiPro, qui se spécialise dans les technologies de suivi oculaire. En utilisant le suivi oculaire en réalité virtuelle, il est possible de tester rapidement et de manière plus rentable chaque étape du parcours du consommateur dans un magasin virtuel. Le design des emballages de produits,

l'emplacement des produits et la disposition des magasins peuvent être facilement évalués et optimisés grâce aux statistiques et visualisations instantanées du suivi oculaire, qui montrent comment les consommateurs répartissent leur attention, interagissent avec les produits et naviguent dans les magasins.



**Figure 2.6:** Exemple d'utilisation des données oculaire dans les environnements VR [1].

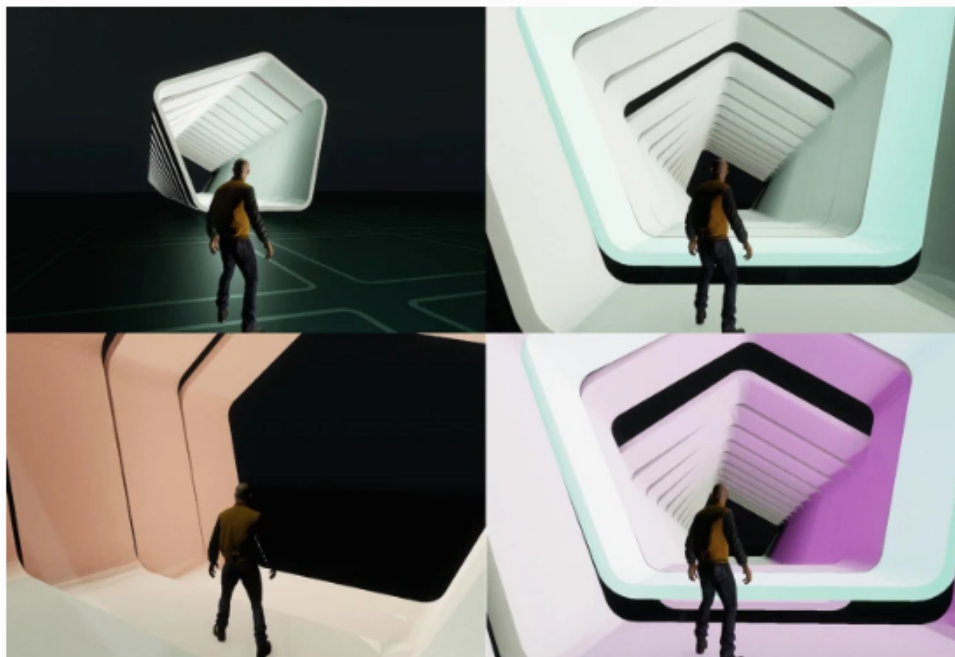
Le système de réalité virtuelle (VR) HTC VIVE Pro Eye HMD est largement utilisé dans les applications qui nécessitent des informations sur le mouvement des yeux (voir figure 2.7). Ce système a été spécialement conçu pour répondre aux exigences les plus élevées des clients commerciaux. Il permet d'améliorer les simulations, les informations et les traitements grâce à une technologie de suivi oculaire de haute précision qui permet aux utilisateurs de voir ce que les utilisateurs voient et d'agir en conséquence.



**Figure 2.7:** *HTC VIVE Pro Eye HMD.*

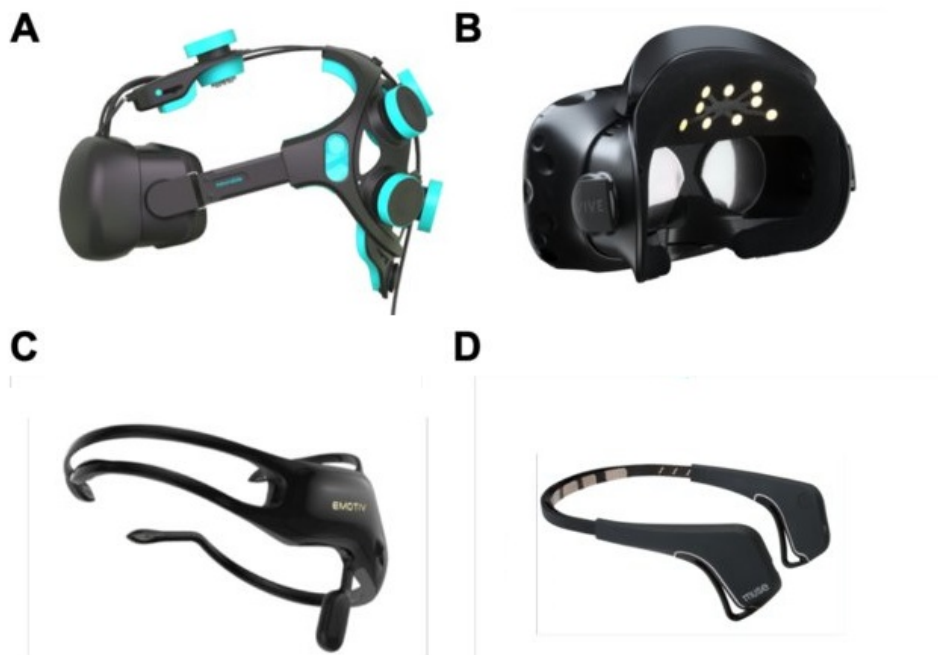
## 2. Les données électroencéphalogramme ou(EEG) :

Les données électriques du cerveau, telles que les types d'ondes cérébrales (delta, thêta, alpha et bêta) ainsi que l'Indice Mental (attention, asymétrie, détente, activité cérébrale), sont couramment utilisées dans la littérature pour diverses applications. Par exemple, Seo et al. ont développé des modèles de classification des émotions en utilisant des données d'électroencéphalogramme (EEG) et des classificateurs d'apprentissage automatique pour les patients atteints de la maladie d'Alzheimer en réalité virtuelle [119]. Doma a également développé un mini-jeu en réalité virtuelle (voir figure 2.8) qui utilise les modèles d'ondes cérébrales définis via l'EEG comme une commande pour interagir avec l'environnement virtuel en temps réel, permettant ainsi des expériences authentiques dans le monde numérique qui seraient impossibles dans le monde physique [37]. L'EEG est donc une entrée alternative pour la réalité virtuelle, qui permet des interactions plus dynamiques et une intégration plus complète avec l'utilisateur.



**Figure 2.8:** Les joueurs élèvent chaque segment en se concentrant sur les commandes push/pull qu'ils ont définies[37].

La figure 2.9 présente quelques dispositifs capable de détecter les ondes cérébrales (les données EEG) lorsque l'utilisateur est immergé dans la réalité virtuelle.



**Figure 2.9:** Quelques capteurs EEG (A)Neurables HTC Vive with Wearable Sensing electrodes , (B), (C) 5-channel Emotiv InsightLooxid Link affixed to HTC Vive, (D)5-channel Muse 2



Avec les données fournies par ces dispositifs, le système peut détecter si l'utilisateur est attentif, stressé, détendu ou si il utilise davantage la partie droite ou gauche de cerveau.

### 3. Les données faciales :

Selon [94] ,cette catégorie de données comprennent les éléments de données liés à l'état de la mâchoire, de la bouche, de la langue, des joues.Les informations de ce type sont utilisés généralement pour synchroniser le visage de l'utilisateur avec le visage de l'avatar virtuel (figure 2.10).



**Figure 2.10:** *L'incarnation du visage*[37].

L'auteur de l'article [37] a utilisé le casque HTC VIVE Pro Eye HMD (Figure 2.7) pour capturer les expressions du haut du visage de l'utilisateur, y compris les yeux et les pupilles, ainsi que le HTC VIVE Facial Tracker (Figure 2.10) pour capturer les expressions du bas du visage, y compris les lèvres, les dents, la mâchoire et les muscles de la langue.

Dans notre contribution, nous incluons également la description des mouvements du

visage en utilisant la méthode Facial Action Coding développée par les psychologues Paul Ekman et Wallace Friesen en 1978 [42]. Cette méthode est utilisée dans plusieurs domaines, tels que :

1. La médecine, pour détecter la douleur et suivre la dépression.
2. Les jeux, pour s'adapter aux émotions des joueurs.
3. La reconnaissance automatique des émotions, pour construire différents appareils tels que des lecteurs de musique qui détectent l'état émotionnel, des robots et des systèmes de communication vidéo.

Nous présenterons dans le chapitre suivant l'utilisation de la méthode FACs dans notre approche.

#### **4. Les données météorologiques :**

Moon a défini un questionnaire de données contextuelles pour récupérer des données météorologiques, en plus des quatre types précédemment cités, afin de démontrer l'extensibilité de ManySense VR. Ce questionnaire permet de récupérer des informations telles que l'emplacement, la température, la pression atmosphérique, l'humidité, les informations sur le vent, la nébulosité et la visibilité. L'auteur utilise un site web [5] pour collecter et traiter les données météorologiques provenant de diverses sources, telles que des modèles météorologiques mondiaux et locaux, des satellites, des radars et un vaste réseau de stations météorologiques, sous les formats JSON, XML ou HTML [5].

### **2.7.3 Architecture générale d'un système sensible au contexte**

L'architecture des systèmes sensibles au contexte diffère considérablement de celle des systèmes classiques. Selon Dey [36], les applications classiques traitent des données internes ou des informations explicites fournies par les utilisateurs. En revanche, les applications sensibles au contexte doivent également prendre en compte les informations de



contexte. Plusieurs architectures ont été proposées dans la littérature, la plupart s'appuyant sur l'architecture proposée par Dey.

En effet, Dey a été le premier à suggérer la séparation entre la réception des informations contextuelles et leur exploitation. Selon El Ghayam [50], cette séparation vise à garantir la réutilisabilité et l'extensibilité du système. De nombreuses études ont convenu d'adopter une architecture composée de quatre couches principales, comme illustré dans la figure 2.11 : la capture de contexte, l'interprétation de contexte, la gestion de contexte et l'adaptation au contexte.

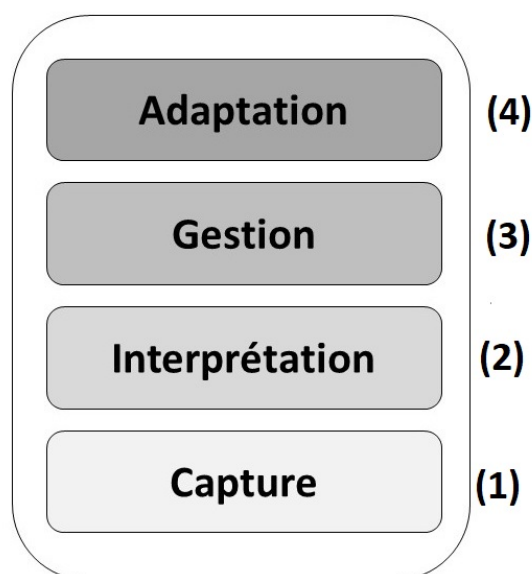


Figure 2.11: Architecture d'un système sensible au contexte.

### 2.7.3.1 La capture de contexte

Les systèmes sensibles au contexte sont conçus pour être attentifs aux changements de leur environnement. La couche de capture est considérée comme la base de tout système sensible au contexte et nécessite l'installation de différents types de capteurs afin de fournir une source d'informations riches. Selon Indulska [62], ces capteurs peuvent être classés en trois catégories :

1. **Capteurs physiques** : Ils transforment l'état d'une grandeur physique observée

en données exploitables. Les appareils modernes sont équipés d'un ensemble de capteurs permettant de mesurer différentes variables caractérisant l'environnement et l'utilisateur, telles que la température, la localisation GPS, le son et l'humidité.

**2. Capteurs virtuels :** Les capteurs virtuels ne possèdent pas de présence physique et ne génèrent pas directement de données de capteurs. Ils fonctionnent en s'appuyant sur des applications et des services pour fournir des informations contextuelles en récupérant des données à partir de diverses sources. Par exemple, ils peuvent extraire des informations à partir du calendrier, du répertoire de contacts, des statuts Twitter, des applications de chat et de messagerie électronique. Les capteurs virtuels utilisent généralement la technologie des services Web pour envoyer et recevoir des données. Ainsi, ils agissent comme des interfaces entre les sources de données contextuelles et les applications, permettant ainsi aux applications d'accéder à des informations contextuelles pertinentes provenant de différentes sources.

**3. Capteurs logiques :** Les capteurs logiques combinent des capteurs physiques et virtuels pour générer des informations plus significatives. Par exemple, un service Web dédié à la fourniture d'informations météorologiques peut être considéré comme un capteur logique. Les stations météorologiques utilisent des capteurs physiques pour collecter des données météorologiques, tandis que des capteurs virtuels tels que des cartes, des calendriers et des données historiques sont également utilisés. Les informations météorologiques sont produites en combinant les données des capteurs physiques et virtuels, ce qui permet de fournir des informations de haut niveau et faciles à manipuler pour les applications.

### 2.7.3.2 L'interprétation du contexte

La couche d'interprétation joue un rôle essentiel dans la transformation et l'interprétation des données contextuelles générées par la couche de capture. Elle analyse les données contextuelles brutes et les transforme en paramètres de plus haut niveau, qui sont plus facilement utilisables par les applications. Par exemple, une adresse postale peut être plus compréhensible que des coordonnées GPS brutes. Cette couche permet égale-

ment l'application d'opérations telles que l'extraction, la quantification, le raisonnement et l'agrégation. Cependant, l'utilisation de multiples sources de contexte peut entraîner des résultats contradictoires ou des situations imprécises. Par conséquent, cette couche doit être équipée de moyens et de techniques pour résoudre ces conflits[27].

### 2.7.3.3 Gestion du contexte

Une fois que le contexte a été capté et interprété dans les couches précédentes, cette couche se charge d'organiser, de stocker et de représenter formellement les informations contextuelles. Selon El Ghayam [50], le stockage peut être centralisé ou distribué. Le stockage centralisé est le plus simple et le plus couramment utilisé, car il permet des mises à jour et des modifications faciles des informations contextuelles. Cependant, il présente des limites en termes de ressources de stockage, notamment pour les appareils et les équipements mobiles utilisés dans la sensibilité au contexte. Le stockage distribué résout le problème de l'espace de stockage des appareils, mais il nécessite des fonctionnalités supplémentaires et complexes liées à la synchronisation et à la mise à jour des valeurs de contexte. Avant de stocker une information contextuelle, il est nécessaire de définir un modèle de représentation. Perera [105] présente les six techniques de modélisation de contexte les plus courantes :

1. Modélisation attribut-valeur : Il s'agit de la forme la plus simple de représentation du contexte parmi toutes les autres techniques. L'attribut représente un élément de contexte, tel que la température, tandis que la valeur est la valeur actuelle de cette information. Ce modèle est facile à gérer lorsqu'il y a une quantité limitée de données.

2. Modélisation par schémas de balisage : Cette technique modélise les données à l'aide de balises. L'utilisation de balises permet une récupération efficace des données. Les schémas de balisage tels que XML sont largement utilisés dans presque tous les domaines d'application pour stocker temporairement des données et transférer des données entre les applications.

3. Modélisation graphique : Cette technique consiste à modéliser les informations

contextuelles sous forme de graphes conceptuels. Elle permet de représenter le contexte avec ses relations. Des exemples de cette technique de modélisation sont le langage de modélisation unifié (UML) et la modélisation de rôle d'objet (ORM).

4. Modélisation basée sur des objets : La représentation basée sur des objets (ou orientée objet) est utilisée pour modéliser les données en utilisant des hiérarchies et des relations de classe. Le paradigme orienté objet favorise l'encapsulation et la réutilisation.

5. Modélisation basée sur la logique : Les faits, les expressions et les règles sont utilisés pour représenter des informations sur le contexte. La modélisation basée sur la logique permet d'extraire de nouvelles informations de contexte de haut niveau à partir d'un contexte de bas niveau. Elle complète d'autres techniques de modélisation de contexte en agissant comme un complément.

6. Modélisation basée sur l'ontologie : Le contexte est organisé en ontologies à l'aide de technologies sémantiques. Différentes normes (RDF, RDFS, OWL) et capacités de raisonnement peuvent être utilisées en fonction des besoins.

#### **2.7.3.4 L'adaptation au contexte**

face aux changements de contexte. Afin d'assurer cette adaptation, il est essentiel de définir un ensemble de règles spécifiques qui peuvent être mises en œuvre en utilisant des langages de programmation traditionnels ou en s'appuyant sur des méthodes d'intelligence artificielle. Selon Chaari [27], l'adaptation des applications au contexte peut être réalisée selon quatre approches différentes :

1. Approche conceptuelle : centrée sur l'aspect architectural des systèmes sensibles au contexte et proposant des moyens de faciliter la collecte, l'interprétation et la transmission des données contextuelles aux parties concernées.

2. Plateformes de services : visant à fournir des services pertinents à l'utilisateur en fonction du contexte. Cela comprend la découverte dynamique de services ainsi que le déploiement dynamique de services adaptatifs pour répondre aux problèmes d'évolutivité, de sécurité et de confidentialité.

## 2.8 Quelques systèmes virtuels collaboratifs sensible au contexte

### 2.8.1 vr-UCAM : un module sensible au contexte pour la réalité virtuelle

L'auteur dans [85] présente un module context-aware pour les environnements de réalité virtuelle (VR) appelé vr-UCAM. Ce module permet une interaction plus personnalisée et adaptée à la situation actuelle de l'utilisateur. Il est basé sur le modèle ubi-UCAM pour l'informatique ubiquitaire (ubiComp) et permet aux capteurs et services de VR de se connecter dynamiquement aux capteurs et services d'ubiComp pour partager des contextes unifiés et fournir des services personnalisés. Le module a été appliqué à un système de visite virtuelle du patrimoine culturel et a montré une amélioration de la qualité de l'interaction et de la mémorisation des sites historiques. Le module peut également prévenir la diminution de la sensation de réalité dans des conditions multi-utilisateurs.

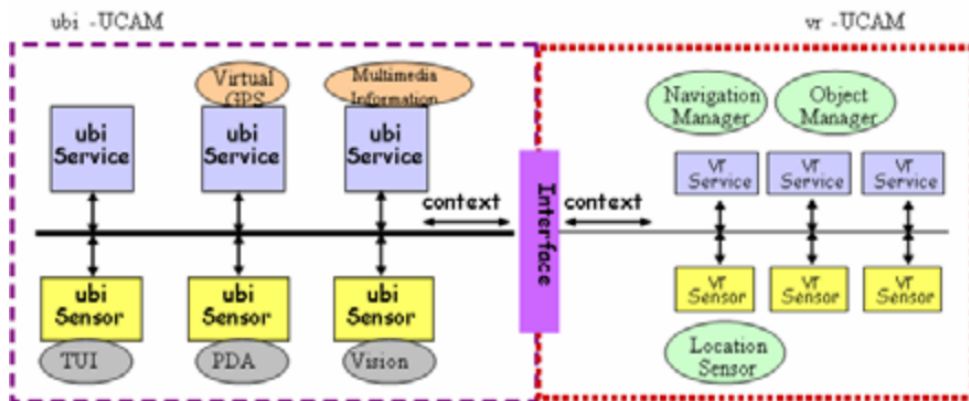


Figure 2.12: L'architecture de vr-UCAM[85].

Le module vr-UCAM se compose de deux modules principaux : vrSensor et vrService. Le module vrSensor comprend trois sous-modules : le module de détection, le module de contexte préliminaire et le module réseau. Le module de détection surveille les activités de l'environnement virtuel et transmet les signaux détectés au module de contexte préli-

minaire. Ce dernier crée un contexte préliminaire en analysant les signaux reçus, exprimé en termes de 5W1H (Qui, Quoi, Où, Quand, Comment, Pourquoi), compréhensible par tous les services. Le module réseau établit des connexions dynamiques avec les services et diffuse le contexte préliminaire aux services connectés.

Le module `vrService` comprend cinq sous-modules principaux : le module réseau, l'intégrateur de contexte, le gestionnaire de contexte, l'interpréteur et le fournisseur de service. Le module réseau gère les connexions dynamiques avec les capteurs. L'intégrateur de contexte collecte les contextes préliminaires sur une période donnée à partir des capteurs connectés aux services vr. Il détermine également le contexte intégré en fusionnant les contextes préliminaires. Ensuite, le gestionnaire de contexte compare le contexte intégré à tous les contextes conditionnels définis par l'utilisateur dans l'interpréteur. S'il trouve une correspondance, il récupère le nom du service associé au contexte conditionnel et transmet le nom et le contexte final correspondant au fournisseur de service. Le contexte final est généré en affinant le contexte intégré avec l'état actuel du service vr.

Cependant, il convient de noter que ce travail se concentre sur la personnalisation des services dans les environnements virtuels individuels. Il ne traite pas spécifiquement des aspects collaboratifs des environnements virtuels collaboratifs (EVC). Par conséquent, il présente des limites par rapport à notre objectif de concevoir une architecture pour un EVC collaboratif. Notre recherche vise à proposer un méta-modèle de référence plus général pour les concepteurs et développeurs qui souhaitent créer des systèmes de réalité virtuelle malléables, en prenant en compte les spécificités de la collaboration entre utilisateurs dans les EVC.

### **2.8.2 Unified Context-aware Augmented Reality**

Dans l'étude présentée par [120], un cadre d'application de réalité augmentée sensible au contexte unifié est proposé pour le guidage intelligent des utilisateurs lors de visites de musées. Ce modèle permet aux utilisateurs de participer activement à la génération

de contenu lors de la visite du musée en découvrant des artefacts personnels intéressants dans les galeries d'art grâce à la génération de comportement basée sur le contexte. De plus, les utilisateurs ont la possibilité de combiner différents contenus augmentés avec diverses informations afin de personnaliser le contenu en fonction de leurs préférences. Le modèle permet également aux utilisateurs d'étiqueter des objets réels pour y attacher de nouveaux contenus.

Le cadre présenté comprend deux composantes principales : une plateforme auteur et un serveur de contenu. La plateforme de création englobe la sensibilisation au contexte, la reconnaissance et le suivi des objets, l'adaptation du contenu et la visualisation. Le gestionnaire d'espace gère à la fois les contenus générés et les utilisateurs.

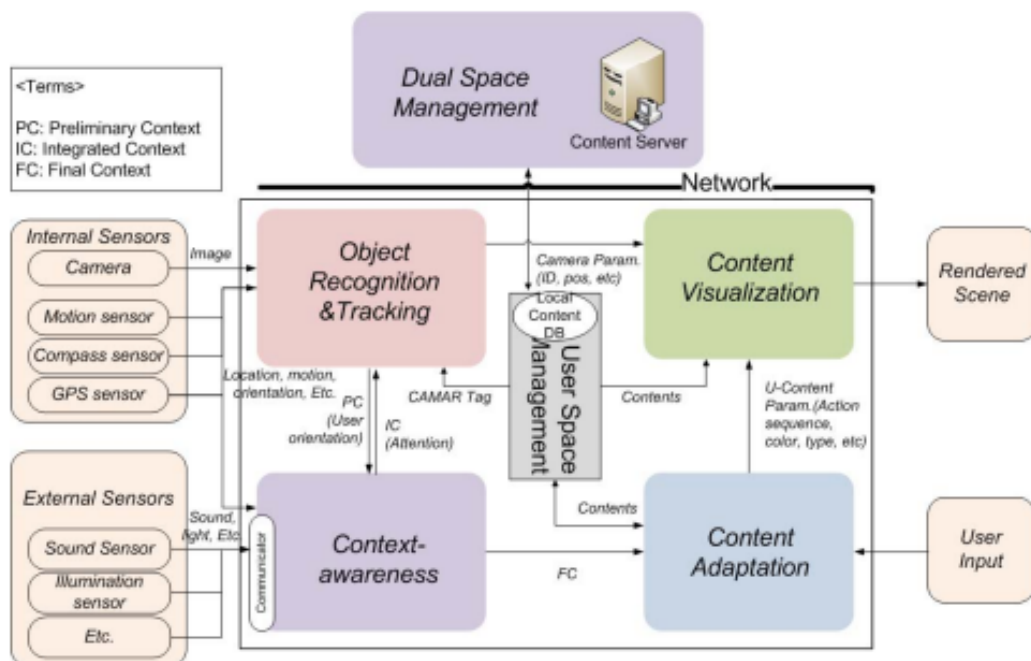


Figure 2.13: Architecture d'application de réalité augmentée sensible au contexte[120].

La Figure 2.13 illustre que la plateforme de création collecte des données sensorielles à partir de capteurs internes et externes, et visualise le contenu sur des objets réels en combinant la sensibilité au contexte, la reconnaissance et le suivi des objets, l'adaptation du contenu et la visualisation. Le composant d'adaptation offre des interfaces pour permettre aux développeurs de créer des applications selon leurs besoins.

### 2.8.3 Système d’incarnation virtuelle

Le travail proposé par Moon en 2022 [94] met en avant la création d’une scène en réalité virtuelle où l’utilisateur peut incarner un avatar en synchronisant ses expressions faciales, mouvements oculaires et gestes corporels. Cette scène se déroule dans un environnement de virtuel et vise à renforcer l’immersion en fournissant un miroir virtuel permettant à l’utilisateur de visualiser son avatar en temps réel comme illustré dans la figure 2.15. De plus, les données physiologiques de l’utilisateur, telles que le pouls et la respiration, sont affichées sur un écran de télévision(voir la figure 2.14). La collecte de ces données est réalisée à l’aide de différents dispositifs, tels qu’un casque HTC VIVE Pro Eye, un contrôleur VIVE, trois trackers VIVE et des capteurs NeuLog, positionnés sur la tête, les mains, les coudes et la taille de l’utilisateur. La Figure 2.5 illustre l’utilisateur portant ces dispositifs, avec des annotations pour chaque type de capteur.



Figure 2.14: *visualisation des données contextuelle*[94].



Figure 2.15: *Le miroir virtuel dans la scène*[94].

Le travail cité nous a fourni des pistes pour identifier les types de données pertinents à utiliser dans notre recherche comme montre dans la partie 5.5.1.2. Bien qu’il ne présente

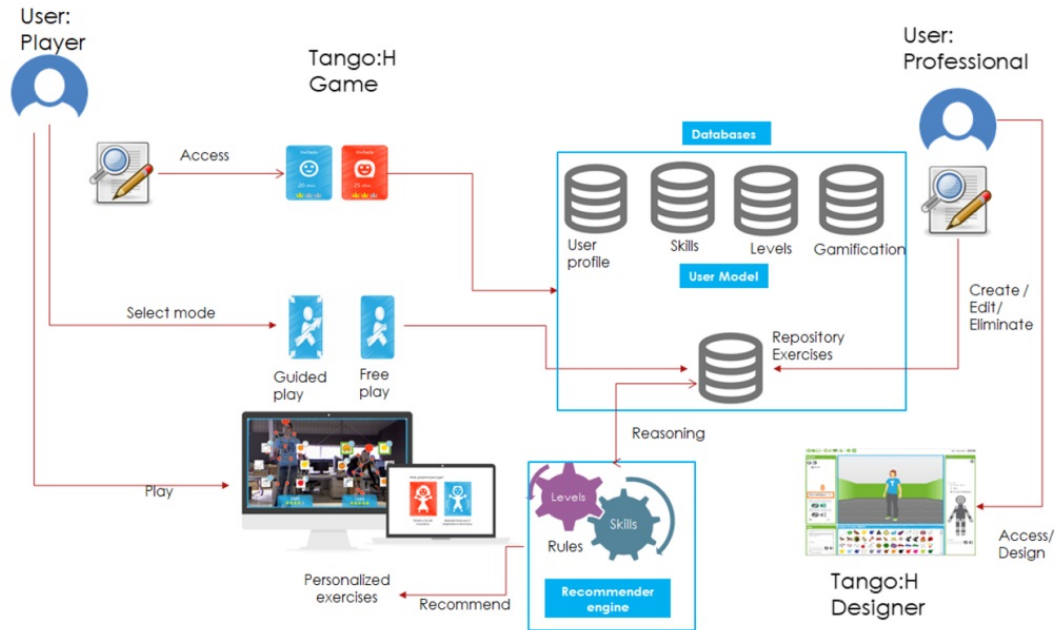


pas une architecture spécifique ni ne réponde aux aspects collaboratifs que nous souhaitons aborder, il a néanmoins été une source d'inspiration technique précieuse pour notre propre travail.

#### **2.8.4 Tango :h un système de recommandation d'un jeu sérieux de rééducation**

TANGO :H est une plateforme de jeux éducatifs accessible basée sur KINECT, qui permet à l'utilisateur d'interagir avec le système en utilisant son corps, sans avoir besoin de contact physique avec des systèmes de contrôle traditionnels. Cette plateforme ouverte et hautement configurable offre la possibilité de créer, personnaliser et adapter des exercices en fonction des besoins et des caractéristiques de chaque utilisateur ou groupe d'utilisateurs.

La plateforme TANGO :H se compose de plusieurs modules principaux (voir la figure 2.16). Tout d'abord, le module TANGO :H Game permet aux joueurs d'interagir avec le système à travers une variété d'exercices. Ensuite, le module TANGO :H Designer est destiné aux professionnels qui souhaitent concevoir différents types d'exercices, définir la gamification des exercices, créer des profils d'utilisateurs en spécifiant leurs caractéristiques et compétences, ainsi que d'attribuer des exercices à des utilisateurs ou des groupes spécifiques.



**Figure 2.16:** *L'architecture de la plateforme TANGO :H[51].*

La plateforme intègre également un modèle utilisateur qui inclut les profils des utilisateurs, leurs compétences, leur niveau, leurs interactions et la gamification. Enfin, le système de recommandation propose des règles de personnalisation et de recommandation qui sont présentées aux joueurs, afin de les guider de manière adaptée et optimale dans leurs expériences de jeu éducatif.

Grâce à TANGO :H, les utilisateurs peuvent bénéficier d'une expérience interactive et personnalisée, en utilisant leur corps comme principal moyen d'interaction. La plateforme offre ainsi de nombreuses possibilités d'apprentissage ludique et adapté aux besoins spécifiques de chaque individu ou groupe. Cependant, il est important de noter que TANGO :H n'est pas une plateforme qui s'adapte en temps réel en fonction des performances de l'utilisateur pendant l'exercice.

Cependant, il est important de souligner que ces scores de compétence ne sont pas toujours un critère fiable pour évaluer la performance d'un patient, car certains patients peuvent ne jamais avoir utilisé les jeux auparavant. Par conséquent, nous suggérons de prendre en compte l'état de santé du patient, y compris la présence de douleurs, ainsi

que sa situation psychologique, en tenant compte de ses émotions. Par exemple, s'il est déprimé, il peut avoir des difficultés à effectuer certains mouvements, même s'ils sont indiqués dans son profil. Dans de tels cas, il peut être judicieux de commencer par lui proposer des exercices plus faciles, puis, lorsque nous détectons qu'il est dans une bonne humeur, nous pouvons lui proposer des exercices plus difficiles. Ainsi, nous adaptons le jeu de manière appropriée en fonction de l'état et des capacités réelles du patient. Par conséquent, il est important de noter que le score bas ne constitue pas nécessairement un indicateur précis de la bonne exécution du geste par le patient.

## 2.9 Conclusion

En résumé, la malléabilité se révèle être un concept crucial à prendre en considération lors de la conception d'environnements virtuels collaboratifs, bien qu'il ne soit pas directement utilisé dans ces contextes. En explorant des termes connexes à la malléabilité, tels que l'adaptation, la sensibilité au contexte et la plasticité 3D, nous n'avons cependant pas trouvé de définitions satisfaisantes pour les environnements virtuels collaboratifs malléables. Cela souligne l'importance de poursuivre la recherche sur la malléabilité et d'explorer de nouvelles définitions et approches afin de concevoir des environnements virtuels collaboratifs malléables répondant aux besoins changeants des utilisateurs et de leur contexte. Il est crucial de développer des approches novatrices pour créer des environnements virtuels permettant une flexibilité et une adaptation en temps réel, tout en favorisant la collaboration et la participation active des utilisateurs.

Notre orientation vers les systèmes de réalité virtuelle (VR) sensibles au contexte s'explique par leur meilleure adéquation à nos attentes de recherche par rapport à d'autres concepts. Au cours de notre exploration, nous avons identifié des contributions significatives ayant influencé la conception de notre propre système. Néanmoins, les travaux existants présentent des limitations, notamment en ce qui concerne les aspects de collaboration, ainsi qu'une absence de travaux portant sur la conception d'un méta-modèle de

référence. Nos prochaines étapes consisteront donc à combler ces lacunes en développant un système plus complet et adapté à nos objectifs de recherche. Nous envisageons d'utiliser une approche de classification des données comme base pour concevoir un module de capture de données, considéré comme l'élément central de notre système. En outre, nous nous appuierons sur l'architecture générale d'un système sensible au contexte pour orienter notre démarche de développement.

En se basant sur les principes de conception dérivés de cette étude, notre objectif ultime est de concevoir un environnement virtuel malléable multi-utilisateur. Cette approche nous permettra de développer un méta-modèle de référence basé sur différentes données contextuelles, pouvant être instancié et adapté selon les besoins spécifiques. Cela fournira un cadre solide pour la conception et le développement d'environnements virtuels collaboratifs malléables, offrant des solutions plus personnalisées et adaptatives. Nous envisageons d'intégrer des éléments tels que la capture et l'analyse des émotions des utilisateurs, ainsi que la personnalisation de l'expérience en fonction de leur état émotionnel.

Le prochain chapitre sera dédié à l'étude des collecticiels afin de répondre à la question suivante : comment intégrer la collaboration dans un environnement virtuel collaboratif malléable (EVCM) ? Nous examinerons s'il existe une architecture classique pouvant répondre aux exigences spécifiques d'un EVCM.

# Chapitre 3

## Les collecticiels

### 3.1 Introduction

La collaboration entre utilisateurs dans les Environnements Virtuels Collaboratifs (EVC) est devenue essentielle dans de nombreux domaines tels que la conception industrielle, la formation, la médecine et les jeux interactifs. Les EVC permettent à plusieurs utilisateurs, géographiquement éloignés, d'interagir et de partager un espace virtuel commun en temps réel, favorisant ainsi la collaboration, la communication et la manipulation d'objets virtuels en vue d'atteindre des objectifs communs.

Dans ce contexte, les collecticiels, également appelés groupwares, jouent un rôle clé en facilitant la collaboration entre utilisateurs. Les collecticiels sont des outils logiciels spécialement conçus pour permettre aux utilisateurs de travailler de manière synchronisée ou asynchrone. Ils offrent des fonctionnalités telles que le partage de fichiers, la communication en temps réel, la coordination des tâches et la gestion de projets, visant à améliorer l'efficacité, la productivité et la qualité des résultats obtenus grâce à la collaboration.

Dans le cadre de cette thèse, l'accent est mis sur la conception et l'implémentation d'un collecticiel adaptatif et flexible dans un environnement virtuel collaboratif. L'étude des collecticiels dans les EVC revêt une importance particulière car elle permet de comprendre les architectures existantes et les approches de malléabilité. L'objectif est d'analyser dans

quelle mesure ces architectures répondent aux trois dimensions essentielles évoquées dans le chapitre précédent, à savoir la réalité virtuelle, la malléabilité et la collaboration.

Ainsi, ce chapitre se propose de définir les collecticiels, d'examiner les architectures existantes des collecticiels, d'explorer les approches de malléabilité, et enfin de synthétiser ces éléments en analysant leur adéquation aux trois dimensions susmentionnées.

## 3.2 Définition

Un collecticiel (également connu sous le nom de groupware ou logiciel collaboratif) est un système informatique qui soutient des groupes de personnes engagées dans une tâche commune et fournit une interface à un environnement partagé [44]] (p. 40). Ces systèmes font partie du travail coopératif assisté par ordinateur (CSCW) et sont spécifiquement conçus pour faciliter la collaboration efficace au sein de groupes ou d'équipes de travail distribués géographiquement et temporellement. Leur objectif principal est d'atteindre un but commun, résoudre des problèmes, partager des données, générer des idées, résoudre des conflits, prendre des décisions ou réaliser des analyses et des conceptions de systèmes. Selon Ellis et al. [44], une conception réussie d'un collecticiel nécessite un équilibre délicat entre les processus et les procédures sociales et la technologie appropriée. Il est crucial de prendre en compte les influences de la technologie sur les personnes, les interactions sociales ainsi que les éléments sur lesquels elles travaillent lors du développement du collecticiel. Par conséquent, il est essentiel de comprendre les besoins et les caractéristiques variés des organisations et des secteurs afin d'intégrer efficacement le collecticiel dans leur travail.

Depuis les années 1980, le terme collecticiel a été utilisé pour désigner des logiciels permettant à des groupes de personnes de collaborer à distance sur un projet commun. Au fil du temps, différentes définitions ont été proposées pour ce concept, reflétant l'évolution des technologies et des pratiques de collaboration. Par exemple, Ellis et al. [44] en 1991 ont défini le collecticiel comme un système informatique qui soutient des groupes de personnes

engagées dans une tâche commune et fournit une interface à un environnement partagé. En 1999, Kline et ses collègues ont proposé une autre définition, précisant que les systèmes de collecticiel sont conçus pour soutenir les groupes de travail ou les équipes distribuées géographiquement et temporellement, afin de favoriser une collaboration efficace pour atteindre un but commun, résoudre des problèmes, partager des données, générer des idées, résoudre des conflits, prendre des décisions ou réaliser des analyses et des conceptions de systèmes [74]. En 2017, d'autres chercheurs ont élargi la définition des logiciels de travail collaboratif, les classifiant en quatre groupes distincts : les logiciels de travail collaboratif à écran unique, à écrans multiples, à appareil unique et à appareils multiples [73]. Une classification largement utilisée des groupwares a été créée par Johansen (citée dans [49]) sous la forme d'une matrice 2x2 de variables temporelles et spatiales, comme le montre la Figure 3.1. Chaque type répond à des besoins différents et se réfère à des technologies différentes. Par exemple, pour la coordination continue des travailleurs séparés à la fois dans le temps et dans l'espace, des outils d'écriture de groupe peuvent être utilisés.

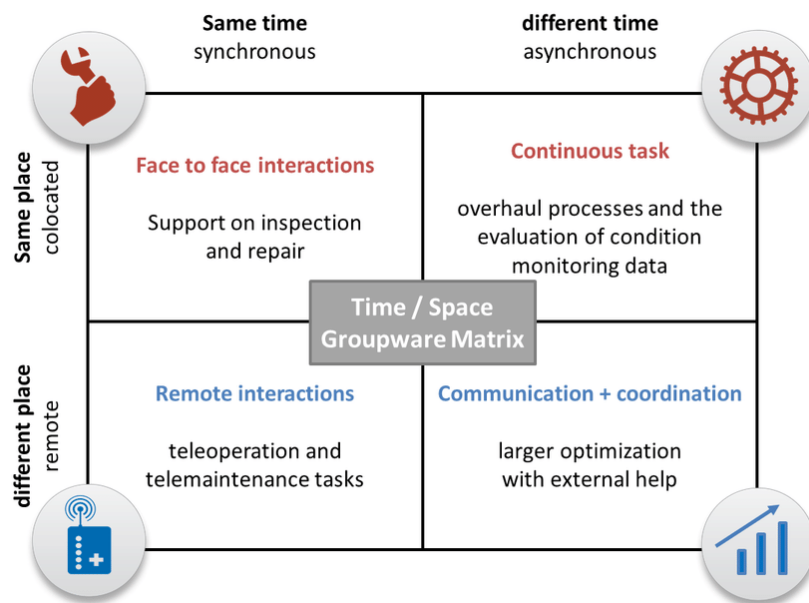


Figure 3.1: Matrice espace-temps pour le travail coopératif assisté par ordinateur (TCAO) [66].

Les jeux sérieux collaboratifs de réalité virtuelle sont des applications informatiques qui combinent à la fois des aspects de jeu et de collaboration pour favoriser la rééduca-

tion. Ces jeux utilisent la technologie de réalité virtuelle pour créer des environnements immersifs où les patients peuvent participer à des activités ludiques tout en travaillant sur leur rééducation de manière collaborative. Ils sont conçus pour soutenir des groupes de personnes engagées dans des tâches communes liées à la rééducation, en fournissant une interface à un environnement virtuel partagé.

Dans le contexte des jeux sérieux collaboratifs de réalité virtuelle, voici des exemples qui correspondent aux quadrants de la matrice CSCW proposé par Johansen [66] :

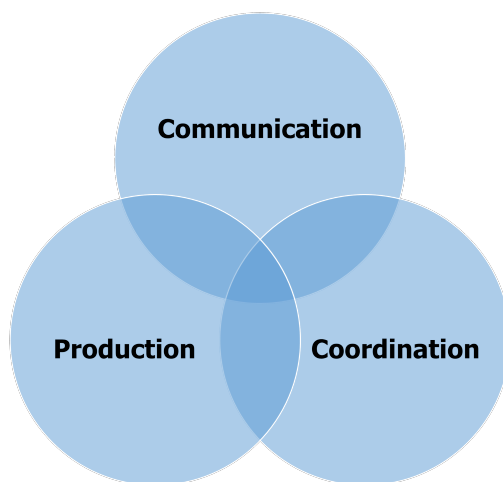
1. **Même temps/même lieu** : Les interactions en face à face sont recréées dans un environnement virtuel partagé. Par exemple, plusieurs patients se trouvent physiquement dans une même salle équipée de casques de réalité virtuelle. Ils peuvent interagir entre eux et avec les éléments virtuels pour effectuer des exercices de groupe, tels que des jeux de coordination motrice ou des simulations de situations de la vie quotidienne nécessitant une rééducation spécifique.
2. **Même temps/différent lieu** : Les interactions à distance sont rendues possibles grâce à la réalité virtuelle. Par exemple, des patients répartis dans différentes localités peuvent se connecter en ligne via des casques de réalité virtuelle et participer à des activités de rééducation en temps réel. Ils peuvent interagir via des avatars ou des représentations virtuelles pour collaborer sur des tâches communes, comme des exercices de rééducation physique ou des jeux de mémoire.
3. **Différent temps/même lieu** : Les tâches continues sont facilitées par les environnements virtuels de rééducation. Par exemple, les patients peuvent avoir des séances individuelles de rééducation à différents moments de la journée dans une salle équipée de dispositifs de réalité virtuelle. Leurs progrès et leurs résultats sont enregistrés et partagés dans un environnement virtuel commun, permettant aux thérapeutes de suivre les performances et d'adapter les plans de rééducation en conséquence.
4. **Différent temps/différent lieu** : La communication avec coordination est ren-



due possible grâce à la réalité virtuelle. Par exemple, les patients peuvent rester en contact même en dehors du centre de rééducation en utilisant des casques de réalité virtuelle et des plateformes en ligne dédiées à la rééducation. Ils peuvent partager leurs expériences, échanger des conseils et se soutenir mutuellement dans leur progression individuelle en utilisant des environnements virtuels interactifs.

### 3.3 Modèle du trèfle fonctionnel des collecticiels

Le modèle du trèfle [116], inspiré du modèle conceptuel d'un collecticiel proposé par C. Ellis [43], offre un cadre conceptuel pratique pour déterminer les exigences fonctionnelles et mener une analyse fonctionnelle. Un collecticiel est divisé en trois espaces fonctionnels différents : la communication, la coordination et la production, comme présenté dans la Figure 3.2.



**Figure 3.2:** *Modèle du trèfle fonctionnel [116]*

1. **L'espace de coordination** : Ce domaine fonctionnel comprend les fonctionnalités qui facilitent l'échange d'informations entre les acteurs du collecticiel. Il offre diverses options de communication, telles que l'audio (téléphone), la vidéo (médiaspace), le texte (messageries) et les gestes (langue des signes).
2. **L'espace de gestion des tâches** : Ce domaine fonctionnel vise à mettre en place

des protocoles qui régissent la gestion des tâches par les groupes d'utilisateurs. Ces protocoles peuvent être purement sociaux, tels que la règle sociale spécifiant qu'une personne doit parler à tour de rôle, ou ils peuvent être formellement spécifiés à travers un système de workflow.

3. **L'espace de production** : Cet espace englobe les éléments partagés qui sont manipulés de manière coopérative pour accomplir les différentes tâches. Cela peut inclure des documents, des fichiers, des ressources ou des objets partagés.

## 3.4 Modèles d'architecture pour les collecticiels

Dans cette section, nous examinons les modèles d'architecture logicielle utilisés pour les collectifs. Les travaux de Laurillau [82] (à partir desquels les Figs de cette section sont tirées) et Khezami [72] ont étudié ces modèles. Ils reposent sur des modèles de systèmes interactifs tels que Arch [13], MVC (Modèle-Vue-Contrôleur) [76] et PAC -Amodeus[96].

Selon Laurillau, chacun de ces modèles représente un style architectural distinct, offrant différentes approches pour organiser et regrouper les fonctionnalités en modules. Le modèle Arch adopte une architecture monolithique, tandis que le modèle MVC repose sur une architecture multi-agents. Enfin, PAC-Amodeus est un modèle hybride combinant les deux styles précédents. Ces modèles mettent l'accent sur la décomposition fonctionnelle en séparant le code central (la logique d'application) de l'interface. Les modèles architecturaux pour les collectifs sont présentés dans l'ordre chronologique de leur apparition : ALV, Zipper, CoPAC, PAC\*, AMF-C, la méta-modèle de Dewan, Clock et DragonFly, Clover, et enfin le modèle C4.

### 3.4.1 Modèle ALV

Le modèle d'architecture ALV est utilisé pour la réalisation de plate-forme Rendez-vous [59], Il se base sur la notion d'objets partagés. Ce modèle compose d'un seul noyau

fonctionnel partagé. Le modèle ALV possède trois facettes qui sont :

- **Une abstraction partagée (shared abstraction) (A)** : qui interprète les objets de domaine circulés entre les utilisateurs en collaboration.
- **Une vues réplique (personal view) (V)** : qui représente l'interface de chaque utilisateur. De plus elle utilise des fonctions dédiées à la modification locale des données afin de traiter les événements générés par l'interaction.
- **Un lien (Link) (L)** : qui assure la communication entre les deux facettes précédentes d'une part, et qui d'autre part, maintient la cohérence entre les données locales de chaque vue et sa représentation abstraite (figure 3.3).

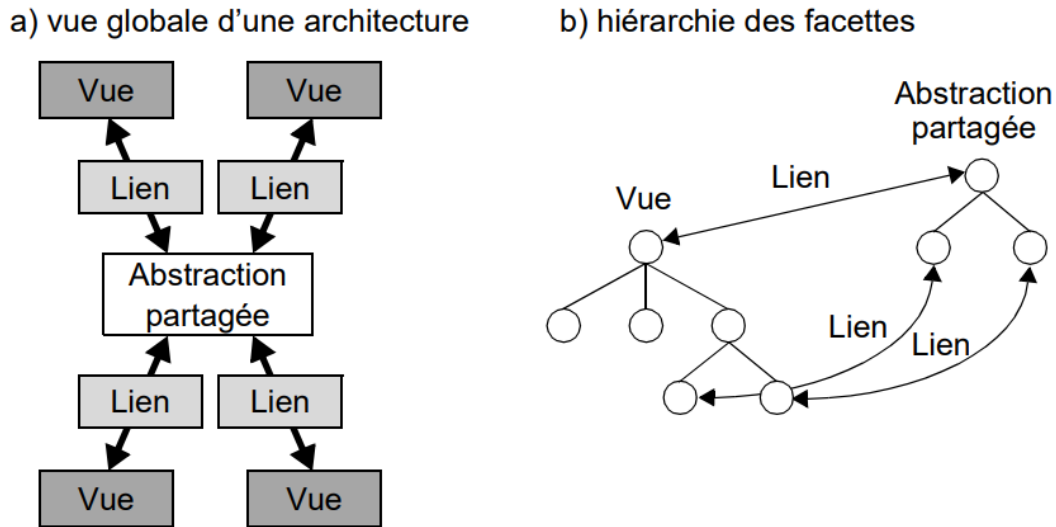


Figure 3.3: *Modèle ALV* [59].

De manière générale, l'architecture d'un collecticiel selon le modèle ALV se compose d'une abstraction partagée qui est connectée à plusieurs vues réparties, une pour chaque utilisateur, à travers des liens.

### 3.4.2 Modèle Zipper

Le modèle Zipper [103] repose aussi sur la notion d'états partagés. Ce modèle décompose le collecticiel en quatre niveaux d'états :

- **Etat de l'écran (Display)** : qui représente l'état des périphériques d'entrée/sortie ;
- **Etat de la vue (View)** : qui correspond à l'état de l'interface utilisateur ;
- **Etat du modèle (Model)** : qui correspond au noyau fonctionnel et aux objets de domaine ;
- **Etat du fichier (File)** : qui représente la représentation persistante du modèle.

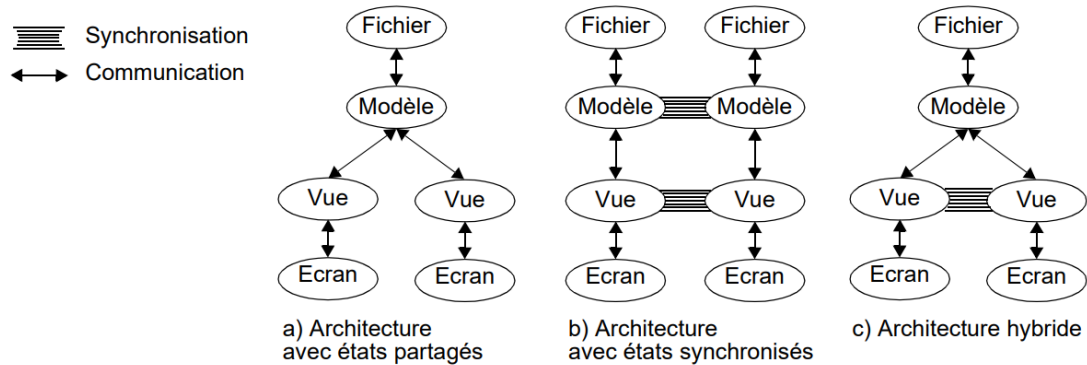


Figure 3.4: *Modèle zipper* [103].

### 3.4.3 Modèle CoPAC

COPAC [116] est une extension du modèle PAC-Amodeus. Ce modèle comprend deux types d'agents : l'agent PAC-Amodeus mono-utilisateur et l'agent collaboratif qui communique avec les autres agents collaboratifs. Le modèle CoPAC, illustré dans la figure, est composé de quatre facettes :

- **La facette Abstraction (A)** : elle interprète les concepts du domaine ;
- **La facette Présentation (P)** : elle représente l'interface utilisateur et interprète les entrées et sorties générées par l'utilisateur lors de l'interaction ;
- **La facette Contrôle (C)** : d'une part, elle fait le lien entre la facette Abstraction (A) et la facette Présentation (P), et d'autre part, elle assure et gère la relation entre les deux facettes (A) et (P) avec la facette Communication (C) et vice versa ;
- **La facette Communication (Com)** : elle assure la communication entre les autres agents collaborateurs.

Contrairement à tous les autres modèles, ce modèle permet toutes les combinaisons

possibles pour rendre les composants publics ou privés, ce qui rend le système flexible et capable de changer dynamiquement l'état de ces composants[82].

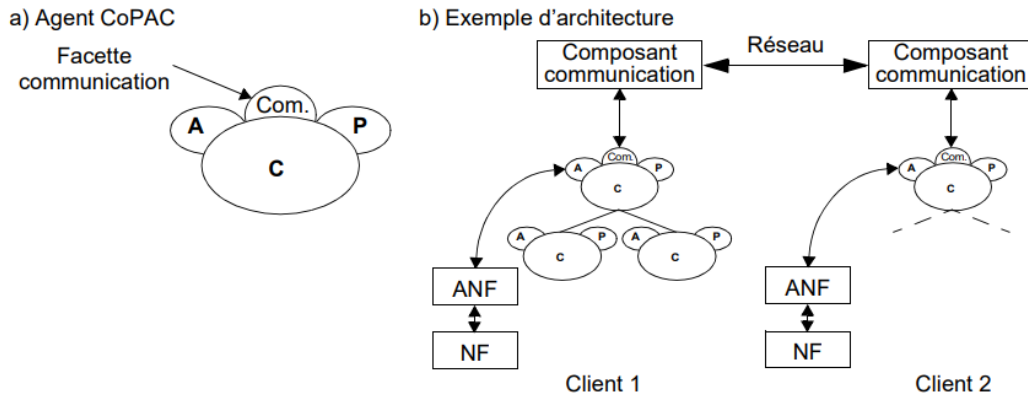


Figure 3.5: *Modèle CoPAC*[116].

### 3.4.4 Modèle PAC\*

Le modèle d'architecture PAC\* est une extension du modèle PAC (Présentation Abstraction Contrôleur), qui est un modèle multi-agent utilisé dans la conception logicielle des systèmes interactifs. Le modèle PAC\* étend également la décomposition du PAC selon le modèle PAC-Amodeus, il respecte aussi une décomposition selon le modèle de trèfle (3C). Le modèle PAC\* propose trois formes d'agents : centralisé, réparti et hybride. Dans la forme centralisée, un agent ciment est responsable de la communication entre trois agents connectés, ainsi que du lien avec le reste du monde. Dans la forme répartie, les trois agents communiquent directement entre eux et assurent la liaison avec les autres agents du système. La forme hybride est une combinaison des deux formes précédentes, où l'agent ciment conserve son rôle de lien avec le monde extérieur.

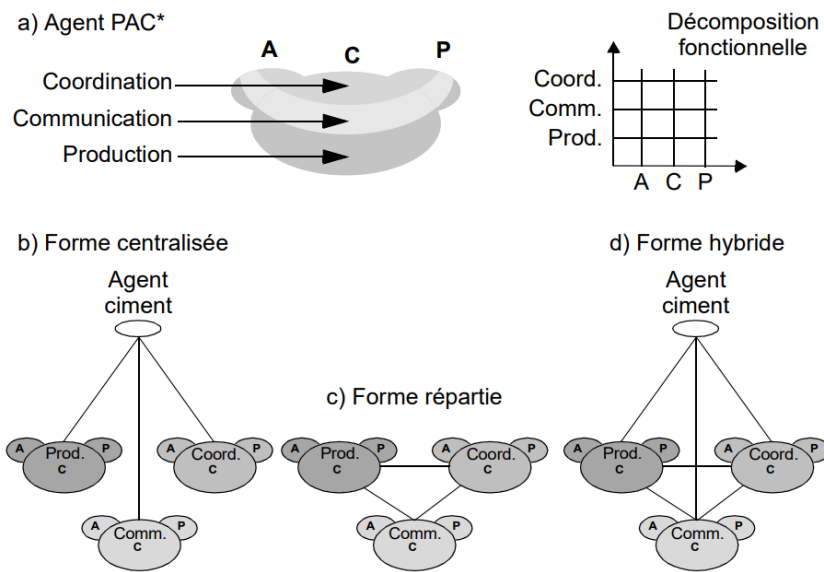


Figure 3.6: *Modèle PAC\** [82].

### 3.4.5 Modèle AMF-C

Le modèle AMF-C, basé sur une approche multi-agent multi-facette [135], va au-delà des facettes traditionnelles de l'Abstraction, de la Présentation et du Contrôle. Il permet l'ajout de nouvelles facettes qui se concentrent sur des aspects spécifiques liés à la facette Contrôle, tels que la gestion des erreurs ou l'assistance. Cela permet d'enrichir l'architecture en incluant des fonctionnalités supplémentaires qui peuvent être nécessaires pour répondre aux besoins spécifiques du système.

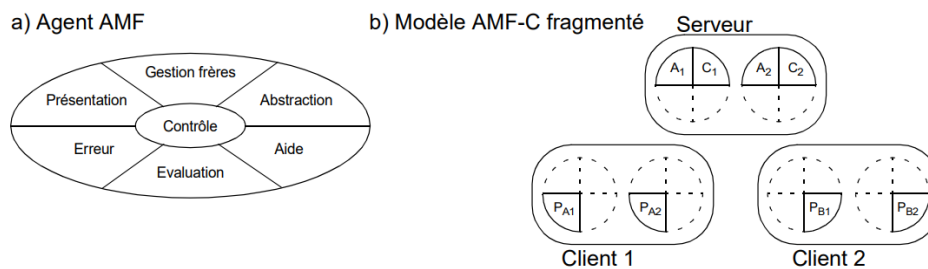


Figure 3.7: *Modèle AMF-C* [135].

Cette approche propose deux stratégies pour la répartition des agents sur différents sites : la fragmentation et la réplication. Dans la stratégie de fragmentation, les différentes

facettes d'un agent sont divisées et réparties sur plusieurs sites. En revanche, dans la stratégie de réplication, un agent complet est représenté à l'aide d'un agent local et d'un agent de référence partagé par plusieurs clients. Les agents locaux assurent la cohérence de la vue de l'agent de référence partagé. Ces deux stratégies permettent une distribution efficace des agents et de leurs fonctionnalités sur plusieurs sites, en fonction des besoins du système distribué.

### 3.4.6 Méta modèle de Dewan

Le méta modèle de Dewan [35] est structuré en couche. Ainsi dans ce modèle, un collecticiel est constitué d'un nombre variable de couches de différent niveau d'abstraction. La couche la plus haute (couche N) représente le noyau fonctionnel de l'application d'où la plus basse (couche 0) correspond au matériel comme le montre la figure ci-dessous.

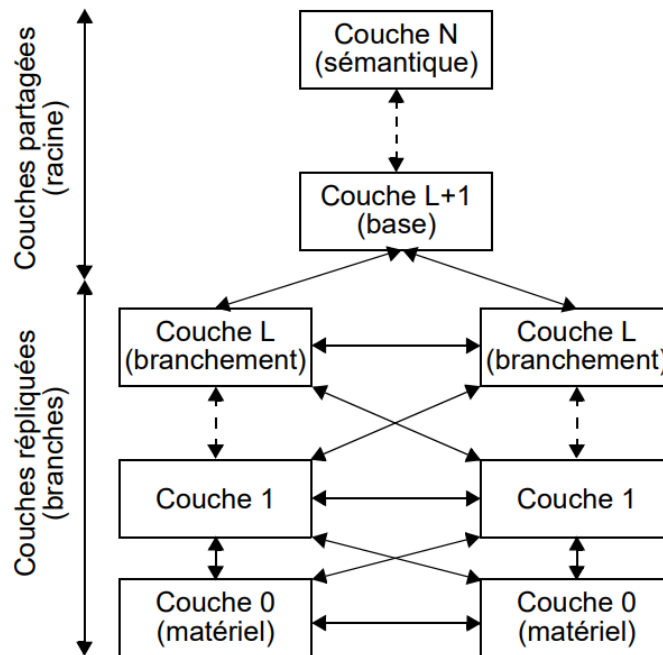


Figure 3.8: Modèle Dewan[35].

L'architecture globale est constituée d'une racine et de plusieurs branches tel que :

- **Les couches partagées** (niveau  $L + 1$  à  $N$ ) constituent la racine et sont respon-

sables de la gestion des objets publics.

- **Les couches répliquées** (niveau 0 à  $L$ ) forment les branches et sont chargées de la gestion des objets privés.

Les couches de ce modèle établissent une communication entre elles en utilisant deux types d'événements : les événements interactifs qui reflètent l'interaction de l'utilisateur avec le système, et les événements collaboratifs qui reflètent l'interaction entre les utilisateurs. De plus, Dewan propose une méthode pour déterminer le niveau auquel la collaboration doit être mise en œuvre dans un collectifiel. Ainsi, si le degré est faible, la collaboration est implémentée au niveau de la Présentation. Si le degré est élevé, la collaboration est mise en œuvre au niveau du Noyau Fonctionnel. Ce degré permet de déterminer le niveau auquel il est nécessaire de mettre en place des mécanismes de contrôle d'accès, de verrouillage des données ou de couplage des données.

### 3.4.7 Modèle Clock et DragonFly

Le modèle d'architecture Clock, présenté par Graham en 1997 [53], est un modèle multi-agent basé sur une approche par composants qui s'appuie sur le modèle MVC. Ce modèle est représenté dans la Figure 3.9.a.

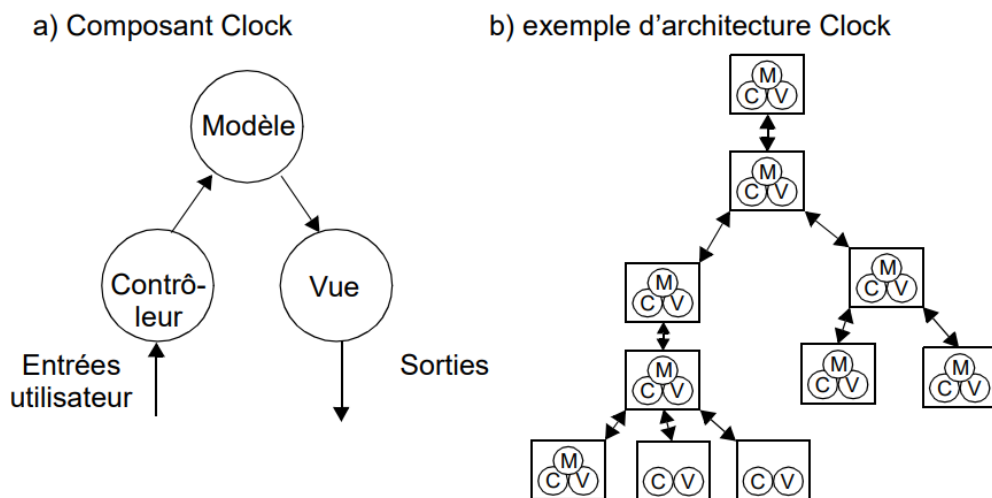


Figure 3.9: *Modèle clock*[53].



Un composant Clock, similaire à un agent MVC, se compose de trois aspects :

1. Le modèle : il est responsable de la gestion des objets du domaine et de leur encodage sous forme de données abstraites.
2. Le contrôleur : il est responsable de l'interprétation des interactions de l'utilisateur avec le système.
3. La vue : elle est chargée de gérer la visualisation des résultats.

### 3.4.8 Modèle Clover

Le modèle Clover est le résultat de la fusion de l'approche en couches répliquées et partagées de Dewan avec la décomposition fonctionnelle du trèfle des collecticiels. Le trèfle des collecticiels identifie trois classes de fonctionnalités essentielles, à savoir la production, la communication et la coordination. Ces trois classes se retrouvent dans chaque couche fonctionnelle du modèle.

Le modèle d'architecture Clover est une instantiation du méta-modèle de Dewan grâce à l'application des cinq couches recommandées par le modèle Arch. Ce modèle se compose de six couches fonctionnelles, où le Noyau Fonctionnel du modèle Arch est divisé en deux couches distinctes : couches répliquées et privées, ainsi que le Trèfle Fonctionnel partagé et public, comme illustré dans la Figure 3.10.

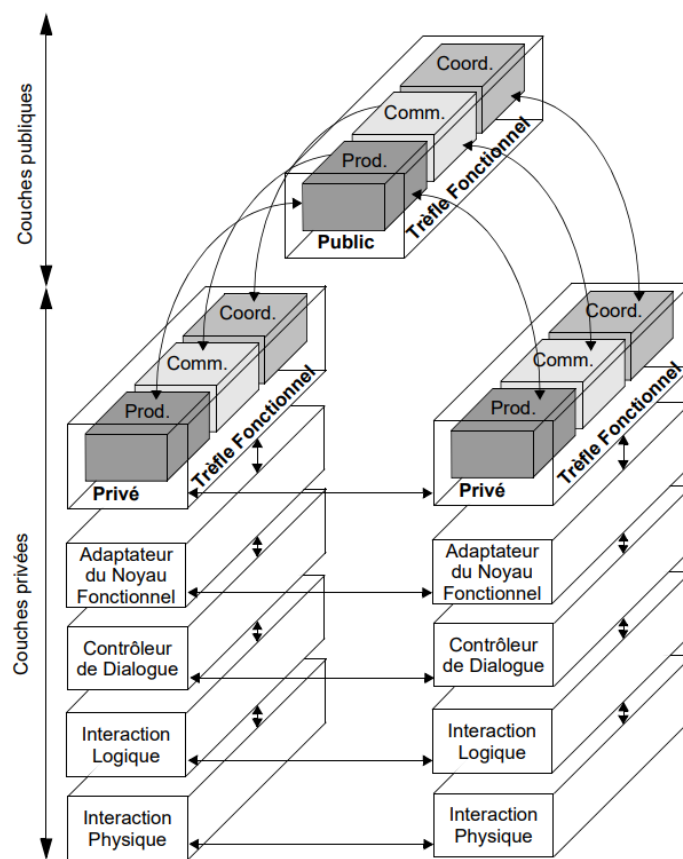


Figure 3.10: *Modèle Clover* [82].

### 3.4.9 Modèle C4

Le modèle C4, proposé par Khezami, est un formalisme d'agent spécifiquement conçu pour la téléopération collaborative via Internet. Ce modèle s'appuie sur le modèle PAC\* et se compose de trois agents, chacun dédié aux trois espaces du modèle 3C, garantissant ainsi la modularité du système. En plus de ces trois agents, le modèle C4 inclut un quatrième agent : l'agent de collaboration. L'ensemble de ces quatre agents forme un agent collaboratif.

La figure suivante illustre l'interaction externe entre deux agents collaboratifs dans le modèle C4 :

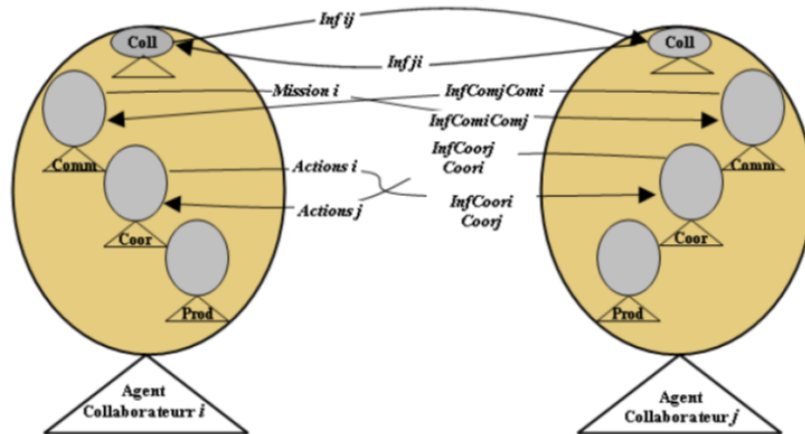


Figure 3.11: *Modèle C<sub>4</sub> [71].*

L'agent de collaboration joue un rôle crucial en assurant la communication directe entre les agents d'un même agent collaborateur et leurs homologues des autres agents collaborateurs. Cette approche évite la centralisation et améliore l'efficacité du système. L'agent de communication est le pivot de l'agent collaborateur, prenant des décisions sur la possibilité de collaboration en se basant sur les perceptions et les informations qu'il reçoit. L'agent de coordination détermine les actions à entreprendre en fonction des choix de mission établis par l'agent de communication, tandis que l'agent de production exécute les actions résultant de la collaboration. Cependant, il est important de noter que l'agent de collaboration est préprogrammé et ne possède pas la capacité de rechercher de nouvelles ressources pour introduire de nouveaux comportements dans le système.

### 3.4.10 Modèle MVC-3DC

Benbelkacem et al. ont proposé un modèle architectural MVC-3DC, qui est une évolution explicite du modèle MVC-3D [16], spécifiquement conçu pour les environnements collaboratifs augmentés et virtuels en 3D. Dans ce modèle, le comportement de chaque objet 3D partagé dans l'environnement est réparti entre quatre composants distincts, à savoir le modèle (M), la vue interactive (iV), la bibliothèque (L) et le contrôleur (C). Les processus de communication ("Comm") et de coordination ("Coord") sont responsables

de la mise en œuvre des politiques d'échange et de distribution des données, tandis que le processus "Prod" est chargé de mettre à jour les vues interactives.

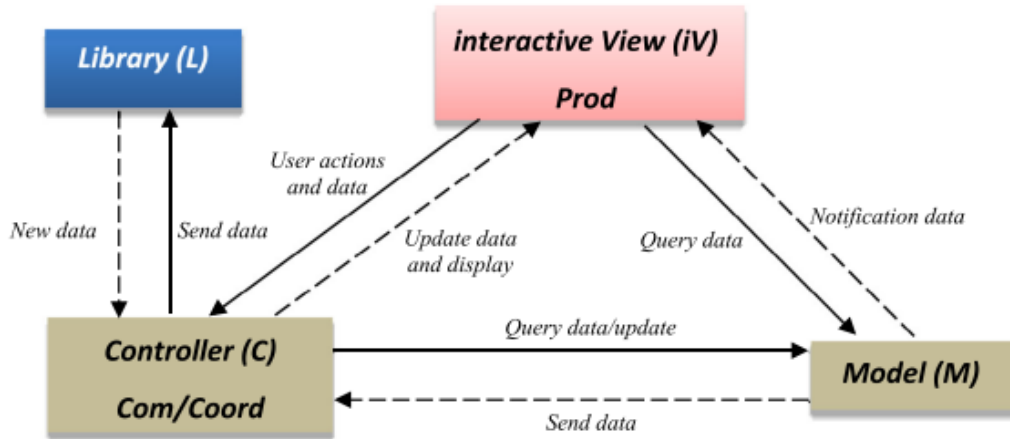


Figure 3.12: Modèle MVC-3DC [16].

- Le modèle (M) stocke les données d'un objet dans des mondes de réalité augmentée/virtuelle, telles que son identifiant, ses types, ses marqueurs d'identifiant, sa position, son orientation et sa matrice de transformation 3D. Il communique avec le contrôleur (C) par le biais des processus "Comm" et "Coord". Lorsqu'un utilisateur effectue une action sur un objet, le processus "Comm" du nœud correspondant demande au processus "Coord" du même nœud de signaler une demande de modification sur l'objet. Une vérification est ensuite effectuée pour autoriser la modification des paramètres de l'objet. Le processus "Coord" envoie une demande de modification au modèle (M) pour obtenir les données de l'objet, puis effectue la modification et envoie des messages de mise à jour au modèle (M). De plus, le processus "Comm" envoie des messages de mise à jour aux autres processus "Comm" des contrôleurs (C) associés à différents nœuds du réseau.
- Le contrôleur (C) gère les demandes de modification et les paramètres des objets des mondes augmentés/virtuels grâce aux processus "Comm" et "Coord". Il communique avec le modèle (M) pour mettre à jour l'objet concerné par la demande de modification, et distribue ces messages de mise à jour à différents nœuds du

réseau.

- La vue interactive (iV) offre une représentation dédiée des mondes augmentés/virtuels. Par exemple, dans un monde 3D composé d'objets 3D, la vue interactive (iV) correspond aux éléments de la scène qui graphiquement représentent ces objets. Afin de maintenir la cohérence du monde 3D, les différentes vues interactives intègrent les matrices de transformation 3D des objets grâce aux processus "Prod", ce qui permet d'afficher plusieurs vues adaptées aux mouvements des utilisateurs.
- La bibliothèque (L) permet à tous les objets de l'environnement virtuel d'être représentés de manière différente dans chaque vue interactive de la session collaborative. Pour ce faire, la bibliothèque intègre, dans chaque nœud, des composants logiciels adaptés aux dispositifs matériels utilisés par chaque utilisateur.

### 3.5 Malléabilité des collecticiels

Les collecticiels sont des systèmes informatiques conçus pour faciliter la collaboration et la coordination entre les utilisateurs dans le cadre de tâches communes. Ils font partie du domaine plus large de l'informatique coopérative, également appelée "Computer-Supported Cooperative Work" (CSCW) en anglais. Le CSCW étudie la collaboration entre les utilisateurs dans un contexte informatique et englobe diverses technologies et approches pour faciliter le travail collaboratif, même à distance.

La malléabilité est une caractéristique essentielle des collecticiels, permettant aux utilisateurs d'ajuster le logiciel selon leurs besoins individuels et les évolutions de leurs objectifs au fil du temps. Cette malléabilité favorise la personnalisation de l'interface utilisateur et des fonctionnalités, contribuant ainsi à une collaboration efficace. Elle offre également une adaptabilité des collecticiels aux nouvelles situations de travail et aux préférences changeantes des utilisateurs. La recherche sur la malléabilité des collecticiels vise à concevoir des systèmes qui permettent aux utilisateurs d'adapter l'application de manière flexible en fonction de leurs besoins. Cela peut être réalisé par le paramétrage, l'intégration et

l'extension de l'application. La malléabilité permet aux utilisateurs de configurer des options préétablies, de relier des composants existants ou de modifier l'implémentation de l'application pour répondre à leurs besoins spécifiques.[28].

Il est important que les collecticiels soient malléables pour répondre aux besoins variables des utilisateurs et aux exigences changeantes du travail collaboratif. La flexibilité et l'adaptabilité des collecticiels sont essentielles pour soutenir une collaboration efficace, où les utilisateurs peuvent ajuster leurs applications sans interruption de leurs activités. La malléabilité "à chaud" est souvent avantageuse dans les environnements coopératifs, permettant aux utilisateurs de personnaliser leurs collecticiels en temps réel.

## 3.6 Approches et méthodologie des collecticiels malléables

Diverses approches ont été développées pour intégrer la malléabilité dans les systèmes de travail coopératif assisté par ordinateur (TCAO). Parmi celles-ci, on trouve les architectures à base de composants [121, 127, 136], ainsi que les architectures réflexives [20], ces dernières ont suscité un grand intérêt. Néanmoins, bien que ces approches soient généralement conçues pour des domaines spécifiques, elles sont également applicables à divers types de domaines [28]. Dans ce qui suit, nous présentons une étude de ces approches.

### 3.6.1 Théorie de l'activité et la coévolution

D'après [20], la malléabilité d'une application repose sur des caractéristiques essentielles de l'activité humaine, telles que la réflexivité et la cristallisation. La réflexivité permet aux concepteurs de l'application d'accéder et de modifier sa structure pendant son exécution, tandis que la cristallisation permet à un artefact transformé par les utilisateurs au cours de leurs activités de conserver et de sauvegarder leurs expériences. Bourguin propose une architecture cadre basée sur ces concepts et mécanismes pour la

construction de collecticiels, qui est implémentée dans la plateforme CoolDA, un environnement réflexif du TCAO. Dans cet environnement, les utilisateurs ont la possibilité de modifier la structure de l'application et de sauvegarder leurs expériences, favorisant ainsi une approche plus globale et coopérative de la malléabilité. La figure 3.13 illustre une capture d'écran de l'environnement DARE, qui constitue la base de la plateforme CoolDA. Selon Bourguin, un collecticiel peut être défini en relation avec la théorie de l'activité. Dans sa proposition d'architecture cadre pour les collecticiels, [20] s'appuie sur

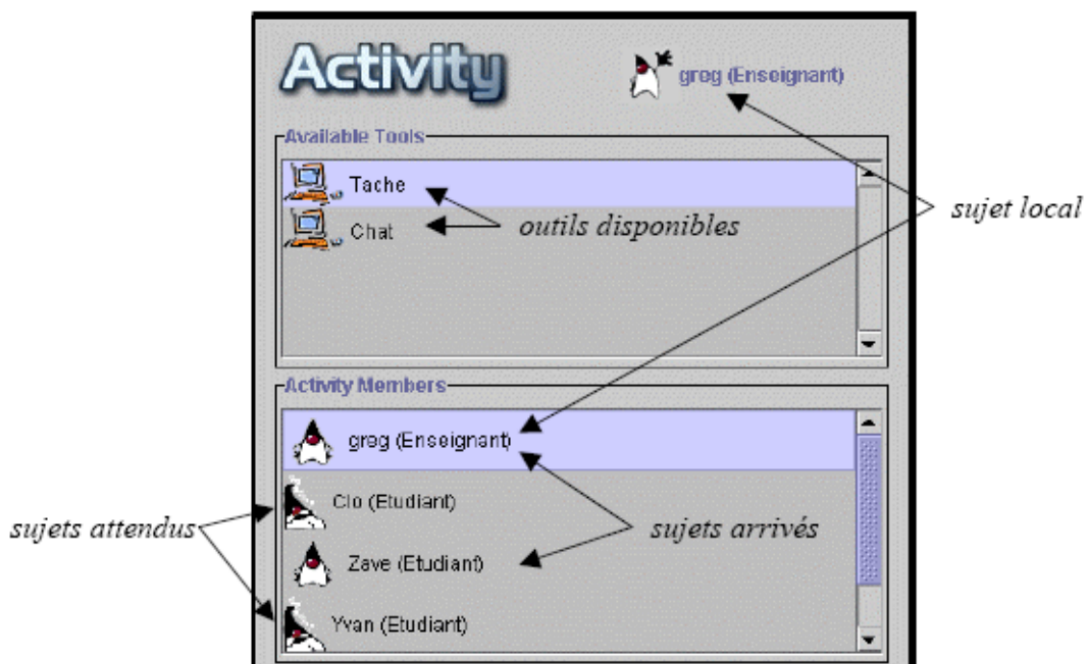


Figure 3.13: Capture d'écran de DARE[21].

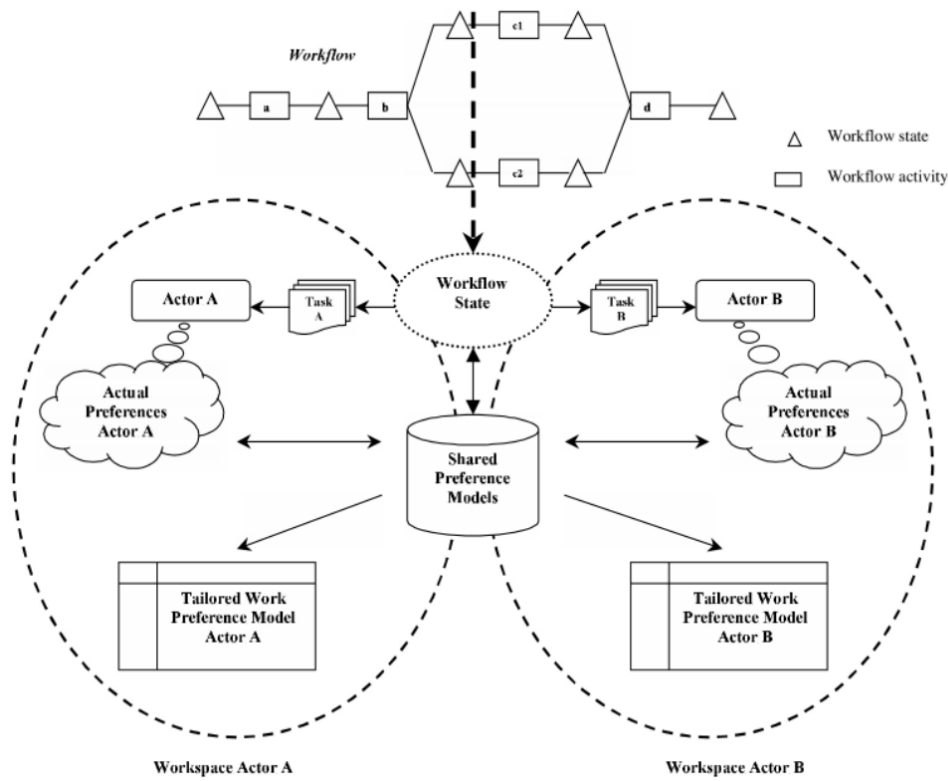
les concepts et mécanismes de la théorie de l'activité pour définir les propriétés fondamentales de l'activité humaine nécessaires à la construction de ces systèmes. La réflexivité est ainsi identifiée comme une propriété fondamentale permettant aux concepteurs de l'application d'accéder et de modifier sa structure pendant son exécution. Une autre propriété est la cristallisation, qui décrit la capacité d'un artefact à transformer et à sauvegarder l'expérience des utilisateurs au fil de leur utilisation. L'auteur a ensuite mis en pratique ces concepts dans la plateforme CoolDA, un environnement réflexif de TCAO (Travail

Coopératif Assisté par Ordinateur) conçu dans le but de mieux appréhender le principe de la coévolution. Cette plateforme est basée sur l'environnement DARE, et une capture d'écran de celui-ci est présentée dans la figure 3.13. L'auteur a également proposé une définition de collecticiel en se basant sur la théorie de l'activité.

### 3.6.2 Approche à base de workflow

Selon Alda et al., la capacité des utilisateurs à personnaliser des flux de travail adaptés sera de plus en plus cruciale dans les applications futures [9]. De plus, Dangelmaier et al. soulignent que la malléabilité vise à améliorer l'utilisation des données provenant des processus de travail collaboratif afin d'optimiser la performance des tâches[32]. Ils présentent une approche (voir Figure 3.14) qui vise à optimiser la malléabilité pour les utilisateurs finaux dans des zones de travail partagées et hétérogènes, en se basant sur des systèmes de gestion de workflow. Cependant, dans les systèmes à grande échelle, la complexité croissante de spécifier de nombreux paramètres liés au système et au domaine représente souvent le problème central de la malléabilité. Par conséquent, la malléabilité se concentre généralement sur des propriétés détaillées d'une tâche étroitement liée à des aspects techniques, ce qui nécessite souvent l'expertise des spécialistes et peut résulter en différents modèles de travail.





**Figure 3.14:** Exemple illustrant l'approche de la malléabilité dans les systèmes de workflow [32].

D'après Chaeib [28], la collaboration joue un rôle crucial dans les systèmes de workflow lorsque les tâches requièrent une connaissance approfondie des personnes impliquées. Afin de soutenir ces workflows, il est essentiel de présenter et d'agrèger les données pertinentes à travers des interfaces qui permettent aux utilisateurs de recueillir des informations et de partager leurs connaissances. Ainsi, la flexibilité de ces interfaces est une partie essentielle de leur fonctionnalité. Dans ce contexte, une tâche flexible peut être simplifiée en choisissant un modèle de préférence approprié pour accomplir un travail basé sur la connaissance collective des parties flexibles du système. Ce modèle de préférence est présenté en trois aspects : la "Situation" qui spécifie l'environnement préféré pour un modèle de préférence donné, la "Décision" qui traite des questions de mise en œuvre, et l'"Evaluation" pour communiquer la position des individus.

### 3.6.3 Approche à base d'objets médiateurs

Le modèle proposé par [132] vise à intégrer des services de base pour soutenir la collaboration dans un système. Ces services, appelés "CSCW enablers", encapsulent des fonctionnalités de collaboration et les fournissent aux objets dans le système pour les rendre "conscients de la collaboration". Les utilisateurs peuvent ajouter des services supplémentaires et ajuster les paramètres de la coopération en fonction de leur choix. Un "CSCW mediator" administre tous les "enablers" appartenant à un objet et encapsule leurs interactions. Les médiateurs sont la caractéristique clé qui permet la malléabilité de la coopération pendant l'exécution. En utilisant ce modèle, les fonctionnalités de collaboration peuvent être spécifiées pour être appliquées lorsque les objets sont modifiés. La figure 3.15 représentée illustre la transmission d'un appel de méthode effectué sur un objet cible à son médiateur. Il sélectionne les "enablers" associés à l'objet cible et invoque les fonctions de collaboration qu'ils fournissent.

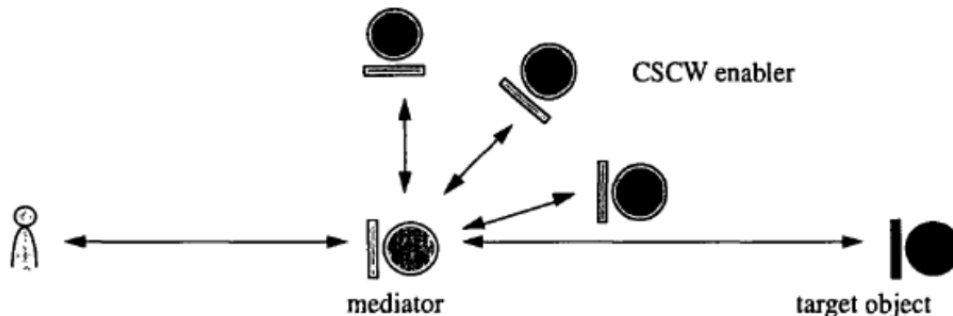


Figure 3.15: Approche basée sur des médiateurs[132].

### 3.6.4 Approche à base d'objets composants

Beaucoup de recherches ont été réalisées pour la conception des architectures à base de composants. Nous présentons brièvement quelques-unes :

- **Approche à base de la réflexivité** : Selon Stiemerling [127], le concept d'une application computationnelle réflexive est proposé. Cette application possède une autoreprésentation qui englobe ses différents aspects. Il est crucial que cette autoreprésentation soit

modifiable et interconnectée avec les aspects qu'elle représente. Ainsi, si la représentation de l'application est modifiée, l'application elle-même subit également des changements en conséquence.

- **Approche basé sur le champs de validité** : Les auteurs [148] ont centré leur attention sur la définition des comportements des systèmes au moyen de champs de validité spécifiques. Ils ont élaboré une approche basée sur la malléabilité déclarative, qui démontre comment les incohérences résultant de déclarations contradictoires peuvent être traitées, soit automatiquement, soit en impliquant les utilisateurs concernés.
- **Approche basé sur les modules fonctionnels** : les auteurs introduisent le concept de "malléabilité à l'extrême", qui se traduit par l'extension des ensembles de fonctions d'un système au moyen de nouveaux modules. Un exemple concret de cette approche est la possibilité pour un utilisateur de télécharger des modules depuis Internet et de les intégrer directement dans son système (tels que des plugins, des widgets, etc.). Par conséquent, des normes d'interopérabilité entre les modules provenant de différents fournisseurs sont désormais essentielles. Ces normes doivent décrire les types de modules existants, les méthodes de communication entre eux, ainsi que les services qu'ils fournissent.
- **Approche basé sur l'enrichissement de messages** : Le travail proposé par Payet [104] met en avant l'exploitation du concept d'enrichissement des messages au sein des composants. Ce concept permet à chaque acteur d'attacher des informations supplémentaires à tous les messages circulant dans le système. Ainsi, il est possible d'améliorer la qualité et la pertinence des échanges en ajoutant des informations contextuelles aux messages, facilitant ainsi la compréhension et la prise de décision dans le système.
- **Approche orientée-aspect (AOP)** : Selon Torres et al. [140], un collecticiel doit prendre en compte diverses préoccupations, définies par les intérêts liés au développement du système et d'autres aspects critiques impliquant une ou plusieurs

parties prenantes. Ils soutiennent que la séparation des préoccupations favorise la malléabilité du collecticiel. En dissociant et en traitant individuellement les différentes préoccupations, il est possible de mieux adapter et faire évoluer le système pour répondre aux besoins changeants des parties prenantes, améliorant ainsi sa flexibilité et sa capacité d'adaptation.

En effet, le concept d'un logiciel à base de composants est indépendant de n'importe quel domaine d'application. Ainsi, ces composants peuvent être appliqués dans presque tous les systèmes. De plus, dans ces types d'architectures, la fonctionnalité du système et la fonctionnalité de la malléabilité sont traitées comme des problèmes séparés[114, 122, 137].

### **3.6.5 Approche à base de système multi-agent**

Chaieb a présenté une proposition d'architecture pour un collecticiel malléable basé sur le modèle Arch de Bass et al. [13]. Cette proposition comprend cinq composants principaux : le noyau fonctionnel, l'adaptateur fonctionnel, l'interaction physique, l'interaction logique et le contrôleur de dialogue. L'accent principal du travail de Chaieb est mis sur le noyau fonctionnel, qui joue un rôle central dans le système.

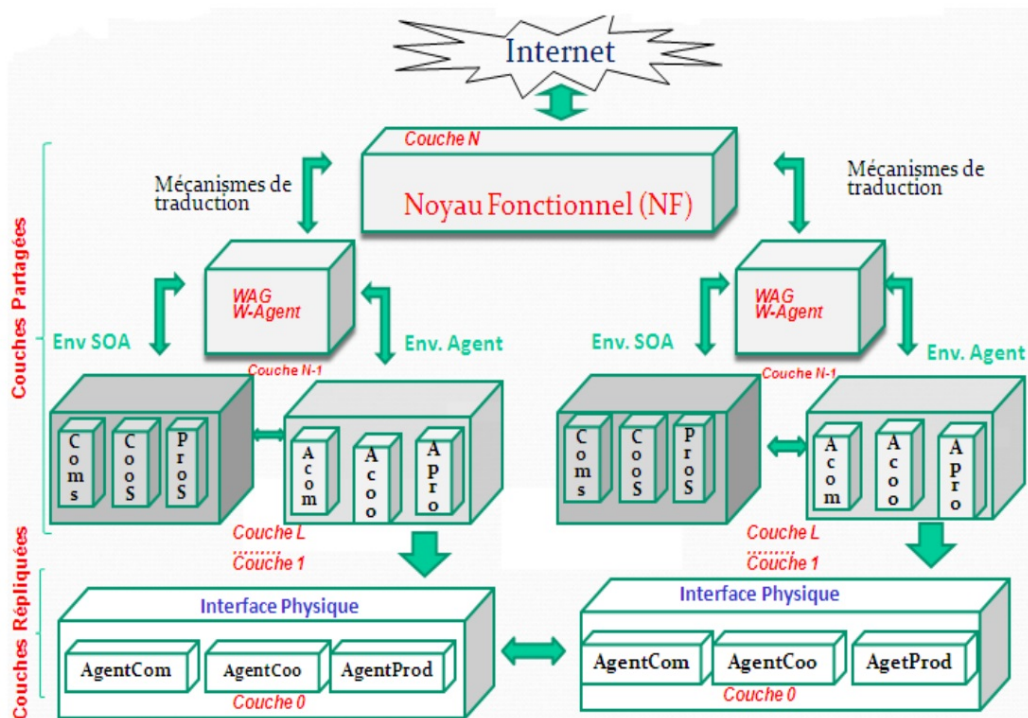


Figure 3.16: Architecture d'un collecticiel malléable à base de services Web d'agents logiciels[28].

De plus, Chaieb s'est inspiré du modèle conceptuel de Dewan [35] pour organiser le collecticiel en différentes couches. La couche supérieure, appelée couche sémantique, correspond au noyau fonctionnel, tandis que la couche inférieure représente l'interface physique du système. Pour soutenir la malléabilité des systèmes collaboratifs basés sur des services web et/ou des agents logiciels, [28] a proposé l'architecture logicielle UDDI4C. Cette malléabilité est obtenue en intégrant des services web externes ou en composant des services internes pour générer de nouveaux services de communication, de coordination ou de production.

## 3.7 Synthèse

### 3.7.1 Analyse des architectures des collecticiels 2D/3D

Le tableau ci-dessous met en évidence le constat selon lequel les architectures existantes ne répondent pas pleinement aux trois dimensions d'un Environnement Virtuel Collaboratif Malléable (EVCM) tel que présenté dans 2.6.2. Bien qu'elles répondent partiellement à la décomposition fonctionnelle en termes de communication, coordination et production pour l'aspect de la collaboration, peu d'efforts sont consacrés à l'architecture d'un collecticiel dans le domaine de la réalité virtuelle. La majorité des travaux sur les environnements virtuels collaboratifs se concentrent davantage sur l'implémentation pratique que sur l'architecture en elle-même.

Nous avons utilisé les symboles et expressions suivants dans le tableau d'analyse :

- "Non Connu (NC)" : utilisé pour les combinaisons qui n'existent pas dans la littérature mais qui pourraient être possibles. Par exemple, si nous prenons le méta-modèle de Dewan, il n'est pas spécifiquement conçu pour la réalité virtuelle, mais il pourrait être pertinent dans ce contexte.
- "-" : utilisé pour les combinaisons pour lesquelles nous sommes certains qu'elles ne sont pas prises en compte.
- Concernant la colonne "malléabilité", chaque exigence est représentée par un "+". Ainsi, une architecture qui répond pleinement à cette exigence doit avoir "++++" dans cette colonne.
- Pour les autres colonnes, nous utilisons un "+" dans la colonne "collaboration" si l'architecture a pris en compte la décomposition fonctionnelle (communication, coordination et production). Dans la colonne "réalité virtuelle", un "+" est utilisé pour indiquer une architecture qui a été initialement conçue pour les systèmes de réalité virtuelle.

Modèle	Réalité virtuelle	Malléabilité	Collaboration	EVC
ALV	NC	+	-	+
Zipper	NC	+	-	+
CoPAC	NC	+	+	++
PAC*	NC	+	+	++
AMF-C	NC	-	-	-
Méta modèle de Dewan	NC	+	-	+
Clock et Dragon Fly	NC	+	-	+
Clover	NC	+	+	++
C4	NC	-	+	+
MVC-3DC	+	-	+	++

**Tableau 3.1:** Analyse des architectures existantes en termes de leur prise en compte des dimensions de l'EVC.

Les architectures des collecticiels existantes répondent partiellement aux exigences de malléabilité que nous avons définies. Dans ces architectures, un espace privé est généralement prévu pour effectuer des tâches individuelles, tandis qu'un espace partagé est réservé aux actions collectives. Nous avons intégré cette notion dans l'une de nos exigences, en stipulant que l'espace privé correspond également au concept des variables d'adaptation qui sont adaptées à chaque utilisateur individuellement. Cependant, il est important de noter que dans le tableau, nous avons indiqué que certaines architectures répondent partiellement à l'aspect de la malléabilité. Cependant, sans l'intégration des autres exigences, on ne peut pas affirmer qu'elles ont véritablement pris en compte la malléabilité.

Le modèle MVC-3DC, tel qu'expliqué par [16], établit une architecture pour les environnements collaboratifs virtuels. Néanmoins, cette proposition ne considère pas la malléabilité et ses exigences associés.

### 3.7.2 Analyse des méthodologie des collecticiels malléables

Après une analyse approfondie, nous avons constaté qu'il n'y a pas de travaux spécifiques sur les collecticiels malléables dans le contexte des environnements virtuels collaboratifs/environnements augmentés collaboratifs (EVC/EAC). Même parmi les approches existantes concernant la malléabilité des collecticiels, aucune ne met véritablement l'ac-

cent sur les besoins individuels de chaque utilisateur. Notre objectif principal est de développer une approche malléable axée sur l'utilisateur, capable de prendre en compte les besoins spécifiques de chaque utilisateur à tout moment. Malheureusement, les approches existantes ne répondent pas à cette exigence.

Pour notre contribution, nous avons choisi de nous appuyer sur les environnements sensibles au contexte. En nous concentrant sur ces environnements, nous pouvons concevoir une architecture qui s'ajuste de manière dynamique aux besoins et aux préférences de chaque utilisateur, en prenant en compte les différents contextes dans lesquels ils évoluent. Cette approche nous permet de mieux répondre aux exigences de malléabilité des collecticiels et d'offrir une expérience plus personnalisée et efficace aux utilisateurs.

### 3.7.3 Bilan

Comme mentionné précédemment, notre approche des environnements virtuels collaboratifs malléables (EVCVM) repose sur trois dimensions principales : la réalité virtuelle (RV), la malléabilité et la collaboration. Pour gérer les contraintes liées à la RV, nous utilisons des outils tels que les casques VR, les gants, etc., ainsi que le développement d'applications permettant une interaction 3D et une immersion partielle ou totale.

En ce qui concerne la malléabilité, nous considérons qu'un système VR collaboratif malléable est un système qui répond aux besoins des utilisateurs. Après nos recherches, nous n'avons pas trouvé de concept répondant à cette définition mieux que les systèmes sensibles au contexte. Par conséquent, nous exécutons les contraintes de malléabilité en utilisant une architecture sensible au contexte qui comprend quatre étapes essentielles : la capture, l'analyse, la prise de décision et l'action.

En ce qui concerne les contraintes liées à la collaboration dans les collecticiels, nous avons constaté que les architectures existantes ne répondent pas à nos sept exigences. Par conséquent, nous devons choisir une architecture existante et ajouter des composants pour satisfaire les exigences manquantes. Nous avons opté pour le modèle Clock[53], qui



se base sur le modèle MVC (Modèle-Vue-Contrôleur), comme expliqué dans la section correspondante. Ce choix nous permet de gérer la répartition du système en espaces privés et partagés, en définissant des ressources spécifiques à chaque utilisateur et des paramètres concernant tous les utilisateurs en collaboration. Il garantit également la cohérence entre les ressources privées des utilisateurs et les ressources publiques et privées, grâce à la possibilité de lier les composants MVC des utilisateurs à une racine commune (voir la figure 3.9, ce qui assure une vue cohérente à tous les utilisateurs. Ainsi, nous pouvons mettre en œuvre des actions malléables individuelles ou collectives, assurant ainsi un système malléable privé pour chaque utilisateur et un système malléable partagé pour le groupe d'utilisateurs.

Notre choix du modèle Clock est également renforcé par l'affirmation de Benbelkacem et al.[16], selon laquelle leur modèle MVC-3DC basé sur le modèle MVC, lorsqu'il est appliqué aux EVC/EAC. Ce modèle présente également l'avantage d'une faible dépendance entre les composants, tels que les fonctions principales, les API de graphiques 3D et les modes de distribution des données. Il s'agit de la seule architecture de collecticiel proposée pour les EVC/EAC.

En résumé, notre approche des EVCM intègre les dimensions de la RV, de la malléabilité et de la collaboration. Nous utilisons des outils de RV pour répondre aux contraintes liées à la RV, une architecture sensible au contexte pour gérer la malléabilité, et le modèle Clock basé sur le MVC pour les contraintes de collaboration (collecticiels).

## 3.8 Conclusion

En résumé, les modèles d'architecture existants des collecticiels présentent des limites dans leur application aux Environnements Virtuels Collaboratifs (EVC) malléables. Ces architectures ne prévoient généralement pas de modules pour suivre en temps réel les contextes des utilisateurs, ce qui limite leur capacité à s'adapter rapidement et efficacement aux besoins changeants. Pour concevoir des EVC malléables efficaces, il est es-

sentiel de développer des architectures de collecticiels flexibles, adaptatives et sensibles au contexte des utilisateurs. Cela implique la création de modules capables de suivre en temps réel les contextes des utilisateurs et d'ajuster dynamiquement les fonctionnalités, les interactions et l'interface utilisateur en fonction de leurs besoins.

Dans la prochaine partie de notre étude, nous présenterons nos contributions à la conception d'une architecture de collecticiel 3D malléable dans les environnements virtuels collaboratifs (EVC). Cette section se concentrera sur la création d'un environnement virtuel (EV) malléable, suivie d'un chapitre détaillant la conception d'un EVC malléable intégrant l'aspect de la collaboration.

Deuxième partie

Contributions

# Chapitre 4

## Conception et réalisation d'un Environnement Virtuel Malléable(EVM)

### 4.1 Introduction

L'objectif principal de cette thèse est de concevoir et de mettre en place une architecture pour les environnements virtuels collaboratifs (EVCM). Comme précédemment mentionné, les EVCM reposent sur trois dimensions essentielles : la réalité virtuelle (RV), la malléabilité et la collaboration. Cependant, il convient de souligner que le concept de malléabilité n'est pas encore clairement défini dans le domaine de la RV, comme nous l'avons conclu dans l'état de l'art. Par conséquent, notre attention sera portée sur l'association des deux premiers éléments, ce qui marque le début de notre investigation vers l'objectif de la thèse.

Dans ce chapitre, nous proposons une architecture pour les environnements virtuels malléable(EVM) que nous appliquerons à une plateforme de réalité virtuelle utilisée dans le domaine de la rééducation, que nous avons nommée "Virtual Reality Personalized Exergame based on Emotion Recognition (VR-PEER)". Cette architecture se compose

de trois phases principales : (i) Capture et interprétation, (ii) Analyse, et (iii) Adaptation.

Ensuite, nous présenterons notre jeu, une application 3D de rééducation qui utilise l'interaction 3D (I3D) et le Leap Motion Controller. L'objectif principal de ce jeu est de tester la pertinence de l'architecture d'un environnement virtuel malléable (EVM) dans le contexte de la rééducation. Nous mettrons l'accent sur la personnalisation des exercices et l'utilisation de la RV pour créer une expérience immersive et motivante pour les patients en rééducation fonctionnelle.

## 4.2 Jeux sérieux et rééducation

La rééducation fonctionnelle vise à prendre soin des patients afin de prévenir l'apparition de déficiences liées à leur pathologie [77, 91, 108]. Elle vise également à améliorer leur autonomie et, par conséquent, leur qualité de vie. La rééducation fonctionnelle implique la participation de différents professionnels de la santé, tels que les kinésithérapeutes, les médecins de médecine physique, les ergothérapeutes, les psychomotriciens, les prothésistes, et bien d'autres encore. Ces professionnels travaillent en équipe pour aider les patients à retrouver leur fonctionnalité physique, à améliorer leur mobilité, leur force musculaire, leur coordination et leur qualité de vie.

Cependant, l'un des principaux problèmes de la rééducation est que les séances de thérapie peuvent devenir ennuyeuses en raison de la répétition des exercices. Cela peut entraîner de l'ennui chez les patients, affecter leur performance dans les exercices et perturber leur programme de traitement, ce qui peut prolonger la durée de la rééducation. Récemment, de nombreux chercheurs se sont intéressés aux jeux sérieux et à la réalité virtuelle comme solution à ce problème. Les jeux sérieux, en particulier les exergames utilisés en rééducation, peuvent motiver, engager et améliorer l'adhésion des patients à leur traitement. De plus, la personnalisation automatique des exercices pour chaque patient peut aider les thérapeutes [10, 31, 100, 106, 142].

Les jeux sérieux sont des jeux vidéo dotés d'interfaces utilisateur qui exigent une

participation active et un effort physique de la part des joueurs. Ces jeux détectent les mouvements du corps et procurent aux participants du plaisir et de l'exercice. Ils peuvent être utilisés pour motiver les personnes âgées à pratiquer une activité physique [142] et comme alternative à l'exercice traditionnel, car ils sont largement disponibles, permettent une pratique autonome et offrent des possibilités de contrôle et d'ajustement personnalisés. Les jeux sérieux ont prouvé leur effi-

acité pour améliorer les capacités motrices et cognitives, réduire le risque de chutes chez les personnes âgées [10], améliorer la qualité de vie des personnes âgées [106] et des personnes atteintes de maladies chroniques [31]. Ils peuvent également contribuer à améliorer la flexibilité, l'équilibre [100] et l'inhibition cognitive. Ces technologies ont apporté une aide précieuse à de nombreux patients dans le cadre de leurs programmes de rééducation motrice. .

### **4.3 L'architecture de la plateforme virtuelle collaborative malléable d'assistance à la rééducation fonctionnelle**

Proposer une architecture pour les jeux sérieux est une tâche très complexe, surtout lorsqu'il s'agit de la personnalisation en raison de la complexité des systèmes de réalité virtuelle. La figure 4.1 montre que VR-PEER a une structure modulaire permettant la génération facile d'un jeu sérieux adaptatif. L'architecture est divisée en trois phases principales : (i) Capture et interprétation, (ii) Analyse, et (iii) Adaptation. Typiquement, l'architecture d'un exercice personnalisé comprend l'observation de l'environnement, l'analyse et l'évaluation des données capturées, et enfin, la sélection des adaptations et leur exécution :

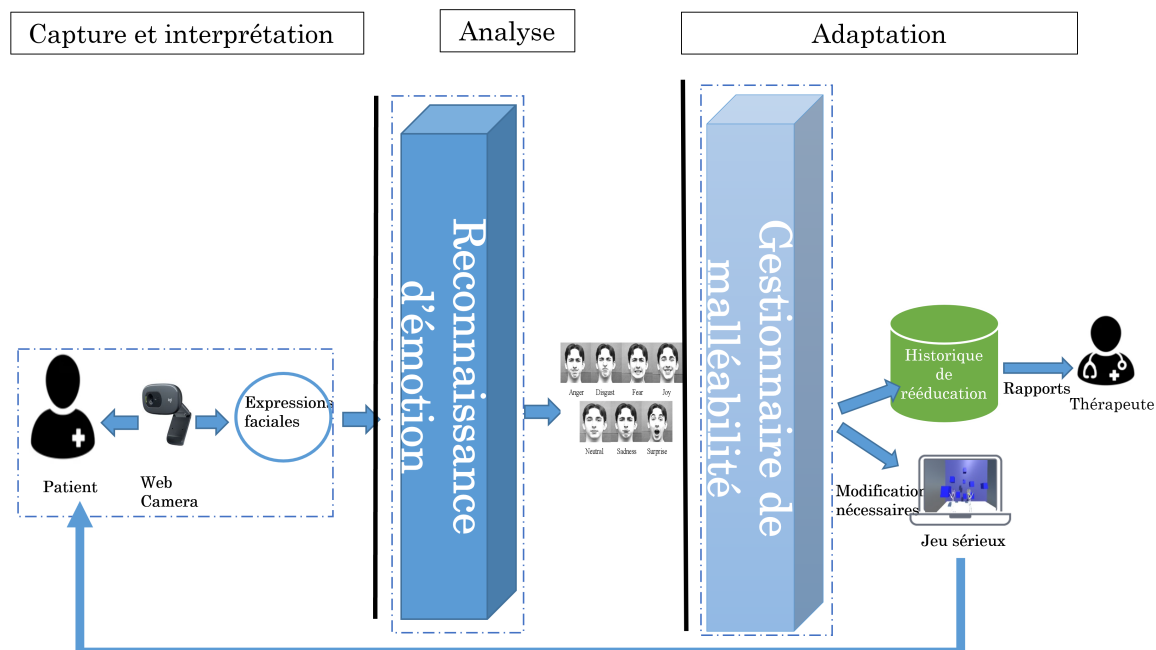


Figure 4.1: VR :PEER :architecture globale d'un jeu sérieux malléable basé sur la reconnaissance des émotions[63] .

### 4.3.1 Capture et interprétation des données

La phase de capture constitue le fondement de notre système. Pour intercepter les données contextuelles, la couche de capture est équipée d'un ensemble de capteurs. Selon une étude [41], ces capteurs sont classés en deux catégories : (i) les capteurs électriques, tels que les signaux EEG, ECG et EMG, et (ii) les capteurs non électriques, tels que Kinect, Web Camera et Oculus, qui permettent la reconnaissance automatique des émotions humaines. Dans notre architecture, chaque type de capteur doit être associé à un composant logiciel permettant d'accéder aux données interceptées. Cette couche se compose de deux parties : un fournisseur de données et un interprète de données. Le fournisseur de données collecte des "méta-informations" contextuelles à partir de l'environnement de l'utilisateur, telles que des photos prises du patient ou des enregistrements audio. L'interprète de données utilise ces informations pour générer une représentation de haut niveau qui décrit les informations contextuelles et les rend plus exploitables et plus faciles à manipuler, telles que les signaux ECG et les pixels de la photo. Pendant le programme de

réhabilitation, une vidéo du visage du patient est capturée sous la forme d'un ensemble d'images. Chaque image est ensuite analysée pour détecter et extraire le visage qui sera utilisé pour la reconnaissance ultérieure.

### 4.3.2 Analyse des données

Nous avons conçu un système pour adapter les jeux d'exercice en fonction de la situation du patient. Pendant les exercices de rééducation, le patient peut éprouver des émotions telles que la douleur, la peur ou la joie, ce qui peut influencer la qualité de la rééducation. Pour s'adapter à l'état émotionnel du patient, nous avons proposé d'utiliser la reconnaissance des émotions à partir de sources telles que les expressions faciales et la parole humaine. Cependant, l'utilisation de plusieurs sources peut conduire à des résultats contradictoires. Pour résoudre ce problème, nous avons proposé d'utiliser des méthodes d'intelligence artificielle telles que les réseaux neuronaux récurrents ou convolutifs. Dans notre étude de cas, nous avons utilisé les réseaux neuronaux convolutifs et le modèle DeepFace proposé par [133] pour reconnaître les émotions faciales des patients. Le modèle DeepFace est l'un des plus populaires dans la reconnaissance faciale et comporte neuf couches. La première couche est constituée d'une image d'entrée de dimensions  $152 \times 152 \times 3$ , qui est convoluée avec 32 filtres de taille  $11 \times 11 \times 3$ . Les deuxième et troisième couches sont respectivement une opération de mise en commun avec des filtres de taille  $3 \times 3$  et un pas de 2 et une opération de convolution avec 16 filtres de taille  $9 \times 9 \times 16$ . Les trois premières couches ont pour but d'extraire la texture et les bords, tandis que les trois couches suivantes sont connectées localement. Le modèle se termine par deux couches convolutionnelles entièrement connectées, suivies d'une couche de sortie softmax avec sept classes possibles. Dans notre cas, nous avons examiné sept émotions, qui ont été classées en trois catégories : l'émotion positive, l'émotion négative et l'émotion neutre. La reconnaissance des émotions faciales a été réalisée à l'aide du modèle DeepFace



et des réseaux neuronaux convolutifs.

$$P(\text{positive}) = P(\text{joie}) + P(\text{surprise}/2) \quad (4.1)$$

Nous avons,

$$\text{négative} = \text{colère} \cup \text{peur} \cup \text{tristesse} \cup \text{surprise}/2 \cup \text{dégoût} \quad (4.2)$$

, donc l'équation sera :

$$P(\text{négative}) = P(\text{tristesse}) + P(\text{colère}) + P(\text{dégoût}) + P(\text{surprise})/2 + P(\text{peur}) \quad (4.3)$$

Durant la séance d'exercice, la webcam enregistre les expressions faciales et calcule les probabilités  $P(\text{négatif})$ ,  $P(\text{positif})$  et  $P(\text{normal})$  pour chaque image. A chaque intervalle de temps  $T$ , il est donc nécessaire de calculer la moyenne émotionnelle (positive, négative et neutre) en utilisant l'équation 4.4.

$$\text{Moyenne} = \sum_{i=1}^{i=n} P_i(\text{émotion})/n \quad (4.4)$$

Tel que :

- ×  $Average = \{APos, ANeg, ANeu\}$  ;
- ×  $Emotion = \{\text{positive}, \text{négative}, \text{normal}\}$  ;
- ×  $N$  : Nombre de frames dans  $T$ .

Ainsi, le résultat du processus sera  $Max(APos, ANeg, ANeu)$ .

### 4.3.3 L'adaptation

La phase la plus importante est l'adaptation du jeu d'exercice en fonction de l'émotion du patient, qui comprend trois parties (i) Le moteur d'adaptation qui définit les mécanismes d'adaptation et les réactions du système en fonction des changements d'émo-

tion du patient. Dans le cadre de la rééducation, cela implique des décisions telles que la modification des métriques de la rééducation motrice (temps de rééducation, nombre de répétitions, difficulté de l'exercice...).

(ii) L'historique de la rééducation motrice qui enregistre tous les changements effectués, permettant ainsi de suivre l'évolution des patients, de préparer des rapports et de les envoyer aux thérapeutes.

(iii) L'application ou le jeu, qui est connecté au moteur d'adaptation pour effectuer les changements appropriés en fonction de l'humeur du patient. Le résultat de l'analyse des expressions faciales du patient peut être soit positif, neutre ou négatif. Le module d'adaptation en temps réel modifie le jeu pour s'adapter à l'émotion du patient, en augmentant la difficulté si le patient est en bonne condition ou en la diminuant si le patient ressent une douleur physique. Si le patient est dans un état normal, le jeu est maintenu tel quel.

Le système est intelligent et ne nécessite pas de modification manuelle des paramètres d'exercice car il détecte automatiquement les changements d'émotion du patient en temps réel et effectue l'adaptation à chaque changement.

## 4.4 Un jeu sérieux malléable à base de la reconnaissance faciale

Nous avons créé un jeu d'exercice en 3D pour la rééducation des membres supérieurs, afin d'évaluer notre système proposé précédemment. Nous commençons par déterminer l'état émotionnel du patient en utilisant la détection des émotions faciales (positives, négatives ou neutres), puis nous ajustons la difficulté du jeu en conséquence, en l'augmentant ou en la diminuant.

#### 4.4.1 Architecture technique

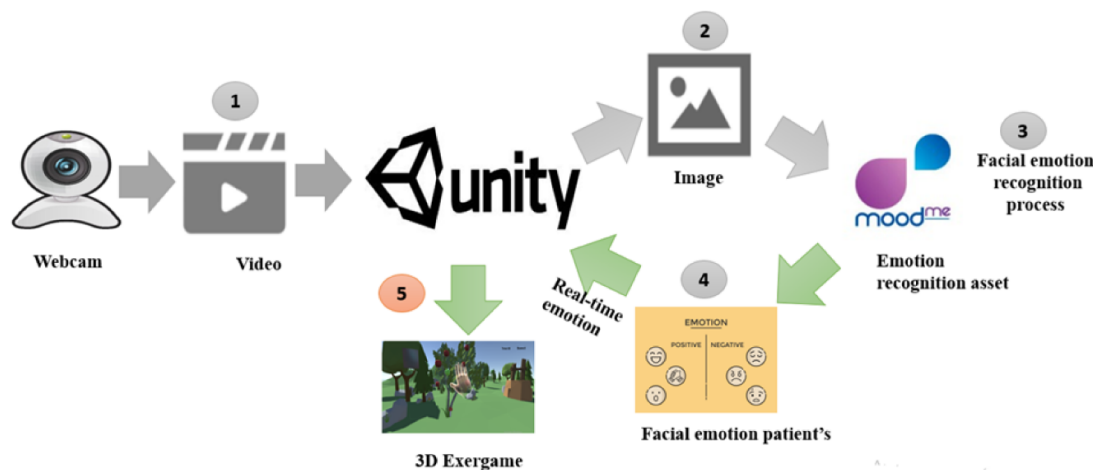


Figure 4.2: L'architecture logique du jeu développé. [63]

Le jeu sérieux a été développé sur la plateforme Unity 3D pour l'exercice de rééducation des membres supérieurs, en utilisant l'architecture préalablement proposée (voir figure 4.2). Nous avons utilisé une caméra web comme capteur pour suivre le visage du patient, afin de détecter en temps réel ses émotions. Pour ce faire, nous avons combiné l'outil Mood Me [3] avec le framework DeepFace [133], qui est un framework utilisable en Python. Nous avons extrait le réseau neuronal (poids de modèle) de DeepFace et l'avons inclus dans la ressource Mood Me. Le package Barracuda [4] a été utilisé pour prétraiter les photos en temps réel du patient et pour détecter son visage pendant l'entraînement. Ensuite, Mood Me a utilisé le modèle DeepFace pour prédire les émotions des utilisateurs en temps réel. Le résultat de l'émotion a été utilisé pour décider quelles actions seront exécutées dans le jeu en utilisant l'équation d'adaptation.

Notre jeu, développé comme une application 3D pour la rééducation, utilise l'interaction 3D (I3D) en utilisant le Leap Motion Controller [2]. Dans ce travail, nous n'avons pas prévu d'utiliser un casque virtuel pour éviter la surcharge cognitive dans les hôpitaux et les centres de réhabilitation. Nous avons utilisé un grand écran et un Leap Motion pour créer un environnement virtuel en semi-immersion, ce qui assure une interaction 3D pour le patient. Nous avons considéré un seul geste dans ce jeu. Le joueur doit étendre

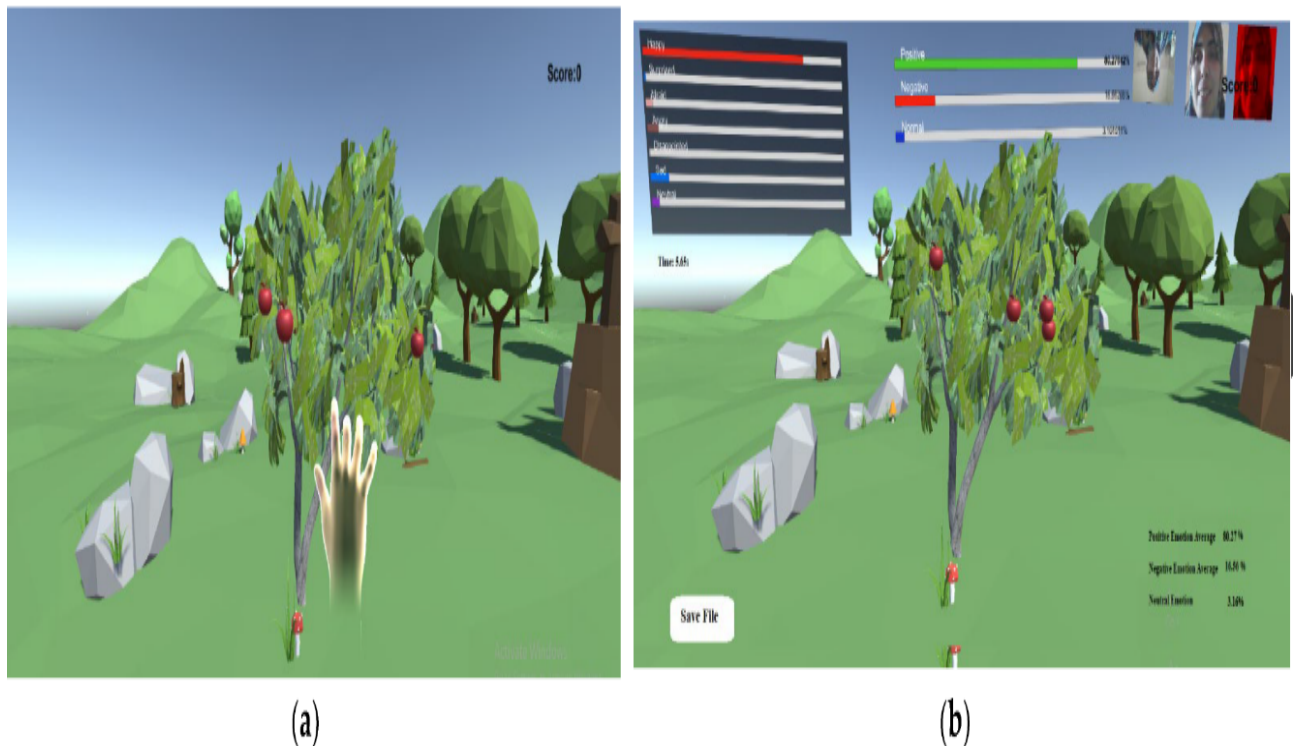
complètement le bras et le déplacer de haut en bas pour faire tomber la pomme sur le sol. Une pomme apparaîtra à chaque intervalle de temps.

#### 4.4.2 Description

Nous avons développé notre jeu comme une application 3D de rééducation, basée sur l'interaction 3D (I3D) et utilisant le Leap Motion Controller. Conçu pour être facile à utiliser dans les hôpitaux et les centres de réhabilitation, nous avons évité d'utiliser un casque virtuel pour limiter la surcharge cognitive. À la place, nous avons utilisé un grand écran et le Leap Motion pour créer un environnement virtuel semi-immersif et permettre l'interaction 3D du patient. Dans ce jeu, nous avons choisi un seul geste : le joueur doit étendre complètement son bras et le déplacer de haut en bas, puis de bas en haut, pour faire tomber une pomme sur le sol à chaque intervalle de temps. Comme précisé précédemment, le processus d'adaptation est basé sur une équation qui utilise le paramètre  $Time\_interval$  une fonction prédéfinie. Nous avons dérivé une autre équation d'adaptation (Equation (4.5)) à partir de celle-ci, dans laquelle  $Time\_interval$  est modifié selon la loi de l'adaptation (Equation (4.6)). Le but pour le patient dans ce jeu est de réussir à faire tomber les pommes au sol en étendant son bras (voir Figure 4.3).

$$Adaptation_F(Time_{interval}) = Time_{interval} \quad Time_{interval} = 5 \quad (4.5)$$

$$Time\_interval = \begin{cases} Time\_interval - 1 & \text{si emotion=positive} \\ Time\_interval + 1 & \text{si emotion=négative} \\ 15 & \text{si } Time\_interval > 15 \\ 3 & \text{si } Time\_interval < 3 \end{cases} \quad (4.6)$$



**Figure 4.3:** Jeu sérieux adaptatif développé sur la base de l'architecture VR-PEER. (a) : interface du participant, (b) : interface utilisée pour tester la reconnaissance des émotions faciales. [63].

Notre objectif est de concevoir un jeu qui s'adapte aux émotions faciales du patient, de sorte qu'il se sente engagé et motivé dans ses exercices de rééducation. Cette adaptation permettra également d'atténuer la douleur causée par la pression sur ses membres et de lui proposer des exercices plus adaptés à ses capacités. En d'autres termes, le jeu doit offrir une expérience personnalisée pour le patient afin de l'encourager à poursuivre sa rééducation.

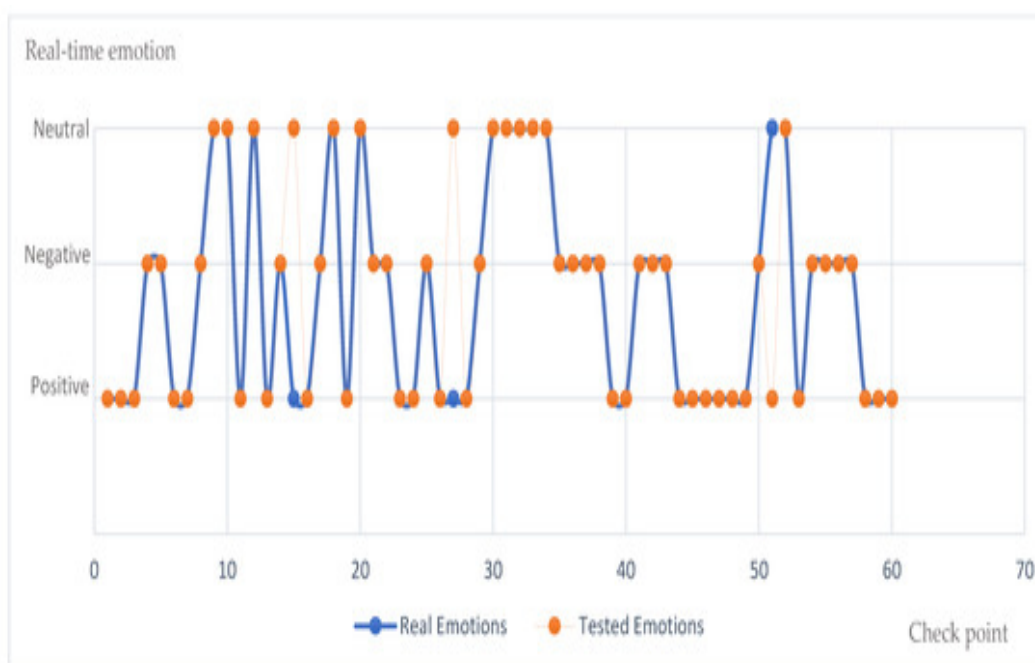
#### 4.4.3 Test et évaluation

Nous avons réalisé deux types de tests afin de démontrer l'efficacité et les performances de notre système. Le premier test visait à prouver l'efficacité de la première étape du processus de reconnaissance des émotions faciales, tandis que le second avait pour but de tester le processus d'adaptation. La première phase d'évaluation de notre système a été

menée pour évaluer ses performances et sa robustesse en prenant en compte les émotions faciales. Ainsi, notre objectif était de répondre à la question suivante : l'adaptation est-elle robuste ? Nous avons donc évalué si l'adaptation pouvait être effectuée de manière indépendante pour tous les patients. Pour ce faire, nous avons utilisé un ordinateur portable équipé d'un processeur Intel Core i7 de 9ème génération, d'une carte graphique GeForce Nvidia, d'une caméra Web intégrée et d'un contrôleur Leap Motion.

**a- Test d'émotion** Nous avons mené un test de 10 minutes, durant lequel nous avons utilisé la méthode DeepFace pour effectuer une reconnaissance des émotions faciales (FER) à chaque trame (soit 24 trames par seconde), ce qui a généré un total de 14 400 reconnaissances émotionnelles pour cette période. L'adaptation du jeu a été effectuée toutes les 10 secondes. Nous avons ensuite calculé la moyenne des émotions positives, négatives et neutres, en utilisant l'équation ci-dessous(4.7), pour obtenir 60 valeurs représentées dans la figure 4.4.

$$\text{Moyenne} = \frac{\sum_{i=1}^{240} \text{PiEmotion}_i}{240} \quad (4.7)$$



**Figure 4.4:** La progression de l'état émotionnel. [63]

Pendant les tests, il a été observé que le système a réussi à reconnaître les émotions prédominantes de l'utilisateur toutes les 10 secondes, et que sur 60 tests effectués, 57 étaient corrects. Les trois tests incorrects ont montré une confusion entre les émotions heureuses et neutres. Par conséquent, il est conclu que le système est efficace dans la plupart des cas, mais qu'il peut y avoir des cas où il est difficile de distinguer entre les émotions heureuses et neutres.

**b- Test d'adaptation** Nous avons divisé notre système en deux parties distinctes : la première partie se concentre sur la reconnaissance de l'émotion de l'utilisateur, et la deuxième partie adapte le jeu en fonction de ses expressions faciales. Pour la première partie, nous avons utilisé une méthode validée dans une étude précédente, et donc il n'était pas nécessaire de la ré-tester. En ce qui concerne la deuxième partie, nous avons testé la réponse du processus d'adaptation en fonction de la reconnaissance de l'émotion.

Le processus d'adaptation a été testé sur un groupe de 15 sujets ayant essayé le jeu (6 hommes et 9 femmes, âgés de 21 à 60 ans). Environ 70% d'entre eux n'avaient jamais utilisé de jeux sérieux en réalité virtuelle, et 15% n'étaient pas habitués à ce type de jeux, mais les ont essayés auparavant ; les autres étaient des habitués. Chaque test a duré 5 minutes, pendant lesquelles les sujets ont effectué des exercices aléatoires tout en variant leurs expressions faciales.

Le processus d'adaptation a été exécuté toutes les 10 secondes, ce qui signifie qu'il y a eu 30 adaptations (points de vérification) pendant chaque test. Toutes les 10 secondes, le jeu a été adapté en fonction de l'émotion de l'utilisateur (positive, négative ou neutre), et nous avons enregistré toutes les 10 secondes l'émotion de chaque sujet, le paramètre d'adaptation testé, ainsi que le paramètre d'adaptation théorique (voir Tableau 4.1). Dans le graphique ci-dessous (Figure 4.5), nous avons modifié la valeur de l'intervalle de temps lors de l'application du système d'adaptation (testé et théorique) et lorsqu'il n'est pas appliqué également. Nous remarquons que l'intervalle de temps varie dynamiquement après chaque exécution du processus d'adaptation, alors qu'il reste constant s'il n'est pas

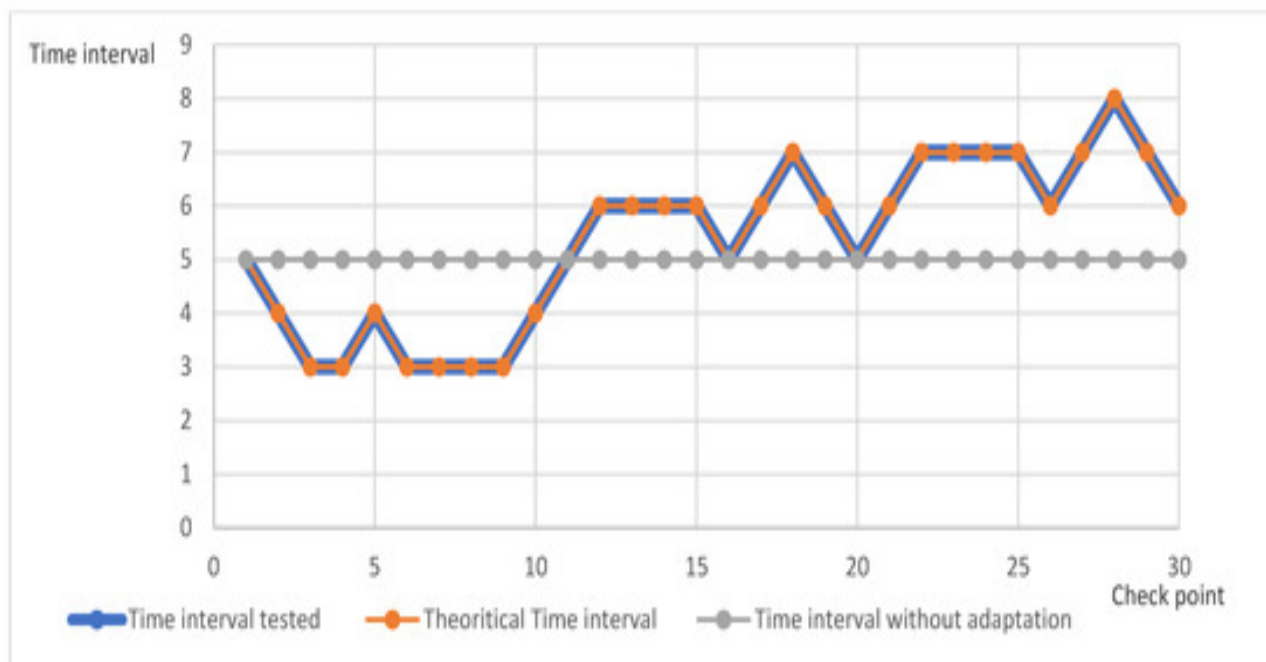
Emotion	Time Interval Tested	Theoretical Time Interval	correct/incorrecte ?
Neutre	5	5	correct
Positive	4	4	correct
Positive	3	3	correct
positive	3	3	correct
Negative	4	4	correct
Positive	3	3	correct
Neutre	3	3	correct
Neutre	3	3	correct
Positive	3	3	correct
Negative	4	4	correct
Negative	5	5	correct
Negative	6	6	correct
Neutre	6	6	correct
Neutre	6	6	correct
Neutre	6	6	correct
Positive	5	5	correct
Negative	6	6	correct
Negative	7	7	correct
Positive	6	6	correct
Positive	5	5	correct
Neutre	7	7	correct
Positive	6	6	correct
Negative	7	7	correct
Negative	8	8	correct
Positive	7	7	correct
Positive	6	6	correct
	correct résultats :26		incorrecte résultats : 0

**Tableau 4.1:** Exemple des données enregistrées pour chaque sujet pendant l'entraînement.

appliqué. Nous avons également constaté que les valeurs obtenues lors des tests sont les mêmes que celles calculées manuellement. Nous pouvons donc conclure que la fonction utilisée pour l'adaptation est correctement appliquée dans le jeu.

Nous avons précédemment annoncé que les informations présentées dans le Tableau 4.1 seront incluses pour chaque expérimentateur dans notre système. Après avoir collecté les données auprès des 15 sujets, nous avons constaté que toutes les adaptations ont été





**Figure 4.5:** Valeur de *TimeIntervalle* à chaque point de contrôle avec l'application et sans l'application de l'adaptation. [63]

effectuées correctement. Nous allons donc calculer la précision de la manière suivante :

$$Prcision = \frac{correctadaptations}{totaladaptations} \times 100 \quad (4.8)$$

Le nombre moyen d'adaptations par sujet est de 30, pour un total de 450 adaptations. Ces résultats indiquent que l'efficacité globale du système est de 100%. Pour évaluer les performances de notre système, nous allons calculer sa précision. Le système est composé de deux parties : (1) le Processus de Reconnaissance d'Emotions (PRE) utilisant le cadre DeepFace, qui est appliqué toutes les 10 secondes et a produit 57 résultats corrects sur 60 tests. (2) le Processus d'Adaptation (PA) avec une précision de 97,35%. La précision globale du système sera la moyenne de la précision du processus d'adaptation et de celle du cadre DeepFace, incluant la Reconnaissance Faciale en Temps Réel d'Emotions (Rt-FER) [129].

$$Prcision = (FER_{prcision} + FerRT_{prcision} + AP_{prcision})/3 \quad (4.9)$$

$$Precision = \frac{97.35 + 100 + 95}{3} = 97.45 \quad (4.10)$$

## 4.5 Conclusion

Dans ce chapitre, nous avons exposé en détail une architecture globale pour le développement d'un jeu sérieux personnalisé axé sur la rééducation fonctionnelle, avec une emphase particulière sur la malléabilité du système. Notre objectif est d'appliquer cette architecture proposée afin de démontrer sa pertinence dans une situation concrète, où elle peut améliorer les mesures de performance de la rééducation en offrant une expérience adaptative et personnalisée.

En utilisant cette architecture, nous visons à créer un environnement virtuel semi-immersif malléable, un jeu sérieux malléable qui peut s'adapter de manière dynamique aux besoins spécifiques de chaque individu en rééducation. En intégrant des mécanismes de malléabilité, nous permettons au système de s'ajuster en temps réel pour fournir des défis et des exercices adaptés aux capacités et aux progrès de chaque utilisateur. L'application de cette architecture dans une situation réelle nous permettra d'évaluer l'efficacité et la pertinence de notre contribution. En offrant une expérience de rééducation adaptative et personnalisée, nous espérons améliorer les résultats et l'engagement des individus tout au long de leur processus de rééducation fonctionnelle.

L'architecture présentée repose sur la reconnaissance des émotions comme mécanisme clé pour la malléabilité. Elle est composée de trois phases principales : capture et interprétation, reconnaissance des émotions et adaptation. La reconnaissance faciale des émotions est utilisée comme étude de cas, mais d'autres techniques de reconnaissance des émotions, telles que le signal ECG, le signal EMG, la parole et le mouvement corporel, peuvent également être explorées.

Dans le prochain chapitre, nous exploiterons les travaux réalisés dans l'état de l'art afin de développer l'architecture mono-utilisateur présentée antérieurement en y intégrant

des aspects collaboratifs. Cette approche nous permettra de prendre en compte et d'interpréter simultanément plusieurs contextes différents. Nous aborderons la problématique suivante : comment adapter un environnement virtuel collaboratif lorsque plusieurs utilisateurs manipulent les mêmes objets en même temps, tout en ayant des contextes individuels distincts ? Afin d'illustrer cette idée, nous proposerons une étude de cas spécifique basée sur le jeu sérieux précédemment exposé.

Il est important de noter que ce chapitre se concentrera davantage sur la mise en œuvre pratique de notre architecture, tandis que le dernier chapitre se penchera davantage sur la conception détaillée et théorique permettant d'intégrer les trois dimensions de l'EVM. Nous aborderons les aspects techniques, les fonctionnalités et les mécanismes pour créer un environnement virtuel collaboratif efficace et malléable.

# Chapitre 5

## Conception et réalisation d'un Environnement Virtuel Collaboratif Malléable(EVCM)

### 5.1 Introduction

Les chapitres précédents ont révélé l'absence d'études sur les Environnements Virtuels Collaboratifs Malléables (EVCM). L'utilisation de tels environnements présente une complexité significative, car elle exige une adaptation en temps réel du contenu 3D et des interactions 3D en fonction des contextes des utilisateurs et de leurs environnements. Il est crucial que cette adaptation ne compromette pas le processus de collaboration. Jusqu'à présent, aucune recherche n'a été entreprise pour concevoir et mettre en place un EVCM, mettant ainsi en évidence le manque de modèles et de méthodes formelles permettant de superviser les ajustements et les modifications des EVC, rendant difficile leur adaptation aux besoins et aux contextes spécifiques des utilisateurs. En conséquence, afin de concevoir un modèle novateur, nous nous sommes engagés pleinement dans l'acronyme EVCM, représentant un environnement qui combine clairement trois systèmes : un système virtuel, un système collaboratif et un système malléable. Ce modèle sera la première approche

approfondie dans la conception d'un EVCM. Nous avons procédé à une définition précise du concept, puis nous avons élaboré une architecture de base pour notre système en identifiant ses principaux composants. Ces derniers ont été conçus en se basant sur les trois dimensions fondamentales (collaboration, malléabilité et réalité virtuelle) décrites dans le chapitre 2. Ils sont répartis dans deux espaces distincts : l'espace des données et l'espace moteur. L'espace des données regroupe les informations essentielles concernant à la fois l'environnement virtuel et le contexte réel de l'utilisateur. Il joue un rôle crucial en rassemblant les données nécessaires pour décrire l'état des deux environnements virtuel(EV) et réel(ER). L'espace moteur, il comprend le gestionnaire de workflow basé sur le modèle proposé par Otmane et al.[99], ainsi que deux nouveaux gestionnaires : le gestionnaire de contexte et le gestionnaire de malléabilité. Le gestionnaire de workflow assure la coordination des différentes tâches et interactions au sein de l'environnement virtuel collaboratif. Le gestionnaire de contexte gère les informations contextuelles, telles que la position et les actions des utilisateurs, afin de les prendre en compte dans la mise à jour de l'environnement. Enfin, le gestionnaire de malléabilité permet d'effectuer les ajustements et les modifications nécessaires pour rendre l'environnement virtuel adaptable aux besoins spécifiques des utilisateurs.

Ensuite, nous présenterons une explication détaillée de chaque composant, suivie d'un modèle d'architecture que nous avons nommé MVC-3DCM (Modèle-Vue-Contrôleur 3D Collaboratif Malléable). Ce modèle mettra en évidence les interconnexions entre les différents composants du système. Par la suite, nous procéderons à une modélisation à l'aide de diagrammes UML, de diagrammes d'activité et d'une architecture réseau afin de représenter cette architecture de manière visuelle et compréhensible. Enfin, nous passerons de l'abstrait à l'application pratique en présentant une simulation : un jeu sérieux collaboratif dédié à la rééducation des membres supérieurs. Cette démonstration concrète permettra d'illustrer comment les concepts décrits précédemment peuvent être appliqués dans un cas réel.

## 5.2 Notre définition d'un EVCM

Un EVCM (environnement virtuel collaboratif malléable) est un système de réalité virtuelle qui offre une expérience immersive personnalisée en adaptant le contenu et les interactions 3D aux contextes des utilisateurs. De plus, un EVCM est conçu pour soutenir le travail collaboratif en fournissant des fonctionnalités de communication, de production et de coordination.

## 5.3 Architecture de base d'un EVCM

L'objectif de cette partie est de proposer une architecture de base des EVCM basée sur les trois dimensions suivantes : collaboration, réalité virtuelle et malléabilité. Comme première étape pour concevoir cette architecture, il est nécessaire de rechercher un modèle permettant l'interaction 3D au sein d'un environnement virtuel. Pour répondre à cette exigence, nous adopterons le modèle de collaboration 3D proposé par Otmane [99] comme référence. Ce modèle, illustré dans la Figure 5.1, décrit les interactions entre les utilisateurs dans le monde réel et le monde virtuel, et facilite la collaboration au sein de l'environnement virtuel collaboratif (EVC), comme expliqué en détail dans la section ??.

Les environnements virtuels collaboratifs (EVCs) ont été conçus pour permettre aux utilisateurs de collaborer et d'interagir de manière efficace à distance. Cependant, pour que ces environnements soient réellement efficaces, il est essentiel de prendre en compte les besoins et les exigences des utilisateurs. La malléabilité joue donc un rôle crucial dans ces systèmes, car les besoins des utilisateurs peuvent varier en fonction de plusieurs facteurs tels que le type de tâches à accomplir, les compétences et connaissances des utilisateurs, ainsi que les contraintes temporelles et spatiales. La malléabilité permet d'adapter le contenu et les interactions en temps réel, ce qui améliore l'expérience des utilisateurs et renforce l'immersion en créant une expérience personnalisée et captivante. En outre, une immersion réaliste favorise la collaboration en permettant aux utilisateurs de se sentir

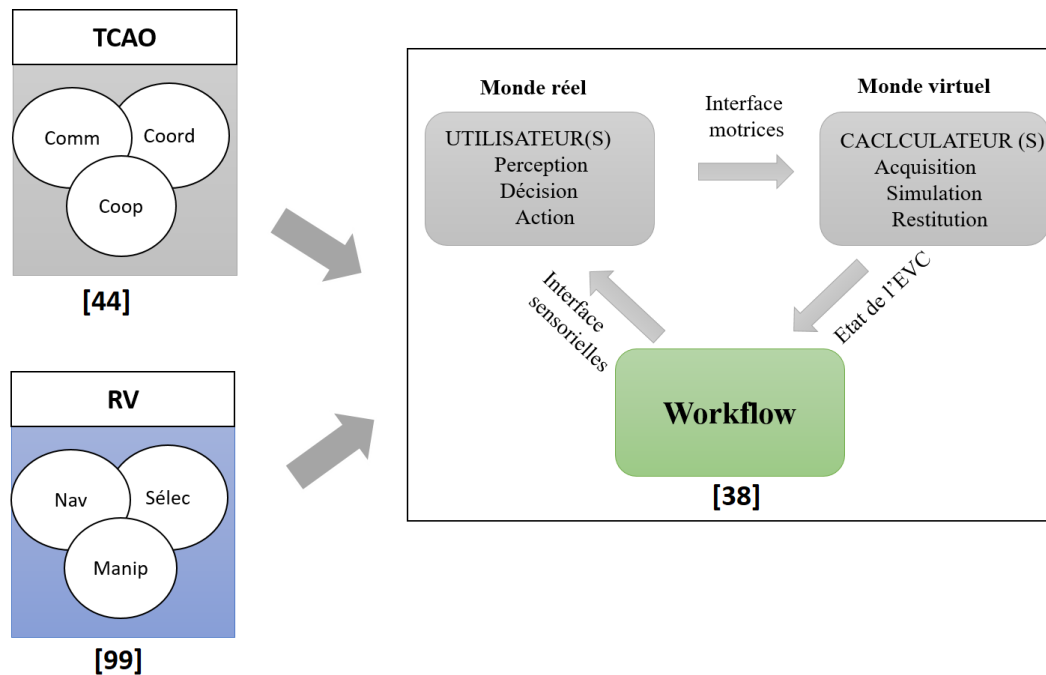


Figure 5.1: Architecture d'un EVC proposé par [99].

complètement immergés, ce qui facilite la communication, approfondit la compréhension de l'espace virtuel et renforce la coordination entre les membres de l'équipe.

Pour répondre à ces besoins, il est essentiel d'intégrer deux composants, à savoir le "Gestionnaire de Contexte (GC)" et le "Gestionnaire de Malléabilité (GM)", dans le modèle illustré dans la figure 5.1. Le GC sera chargé de gérer le contexte, y compris la capture, l'analyse et la classification du contexte, tandis que le GM interprétera, décidera et exécutera les ajustements nécessaires (voir la figure 5.2). Par la suite, nous détaillerons le fonctionnement de ces composants.

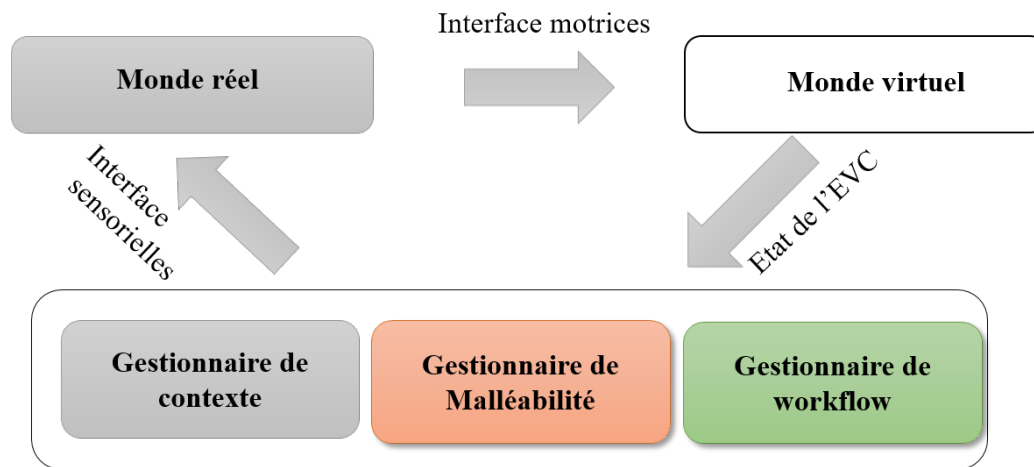


Figure 5.2: Notre architecture de base d'un EVCM.

Ce cadre fonctionnel permet au système d'obtenir des informations sur l'état de tous les utilisateurs, telles que leur activité de navigation, d'interaction, de communication, les difficultés rencontrées ou encore les besoins d'adaptation dans l'EVC. D'autre part, il fournit également les informations sur les contextes des utilisateurs.

## 5.4 La boucle de malléabilité dans un EVCM

La boucle de malléabilité de notre système est essentielle pour assurer une adaptation continue de l'EVC aux différents contextes des utilisateurs. Les utilisateurs commencent par interagir avec l'EVC en effectuant des actions telles que la navigation, la sélection ou la manipulation d'objets (étape 1). Lorsque le système détecte un nouveau contexte ou qu'un processus de malléabilité est déclenché (étape 2), le gestionnaire de contexte collecte des données contextuelles à partir de diverses sources (étape 3), telles que les capteurs environnementaux, faciaux, cinématiques, électriques, etc. Ces données sont ensuite organisées et classées en différents types par le gestionnaire de contexte, ce qui facilite leur interprétation par des algorithmes spécifiques. Ainsi, le gestionnaire de contexte fournit une représentation dynamique et organisée de l'état du monde réel, agissant en quelque sorte comme un auxiliaire pour le gestionnaire de malléabilité afin de réduire sa charge



(étape 4). Les données classifiées sont ensuite transmises au gestionnaire de malléabilité (GM) (étape 5). En utilisant les données fournies par le gestionnaire de contexte, le ge-

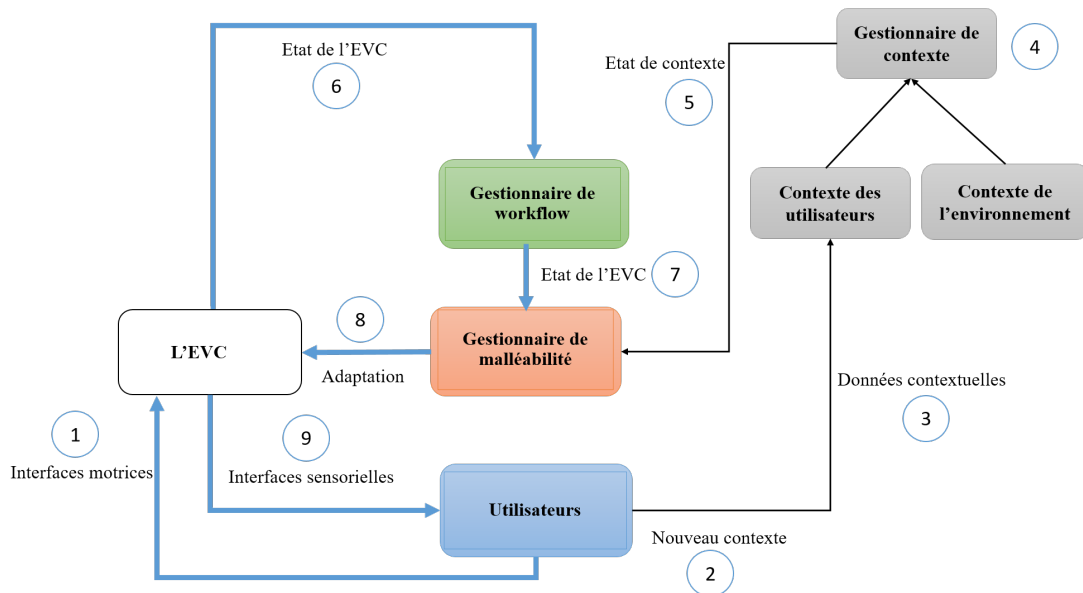


Figure 5.3: La boucle de malléabilité dans un EVCM.

tionnaire de malléabilité (GM) prend ensuite des décisions qui se traduisent par une série d'actions visant à mettre à jour le comportement des objets de l'EVC. Les décisions prises par le GM sont basées sur l'état de l'environnement virtuel, y compris le comportement des objets du système tels que les avatars des utilisateurs, leur position par rapport aux objets de l'EVC, ainsi que les interactions en cours. Les informations nécessaires pour prendre ces décisions sont fournies par le gestionnaire de flux de travail (étapes 6 et 7) et l'état de l'environnement réel (étape 5). Les actions générées par le gestionnaire de malléabilité (GM) peuvent être mises en œuvre par un algorithme artificiel, ou par un expert humain. Les décisions prises sont ensuite exécutées (étape 8) afin de modifier les comportements des objets, créant ainsi un environnement virtuel collaboratif malléable. Les résultats de ce processus d'adaptation sont ensuite renvoyés aux utilisateurs via des interfaces sensorielles (étape 9).

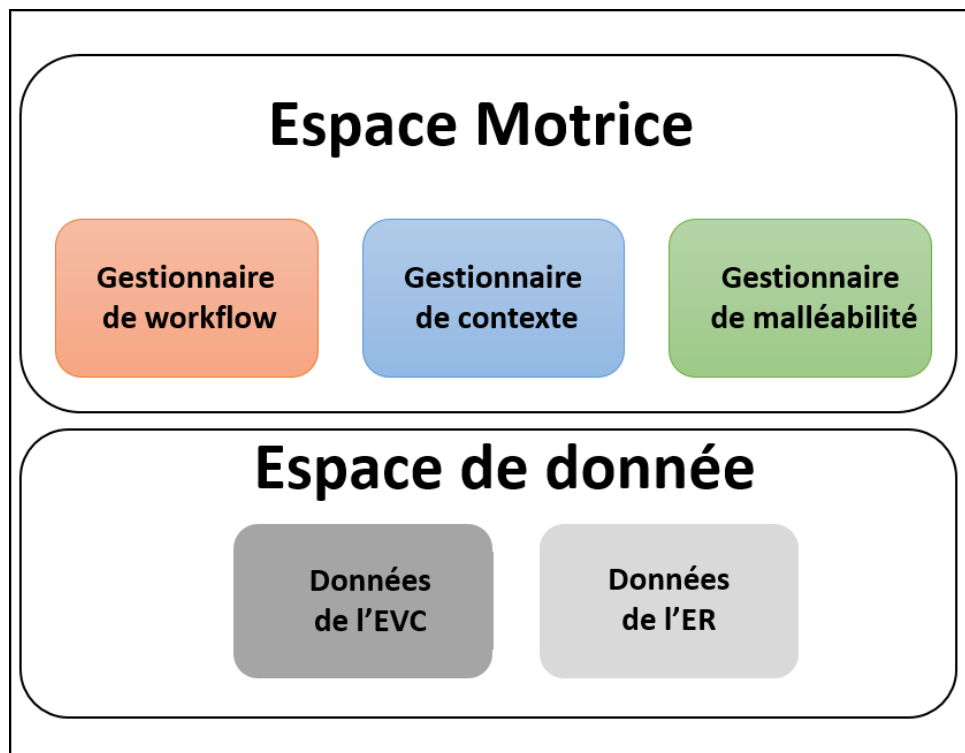


Figure 5.4: Les composants d'un EVCM.

## 5.5 La décomposition fonctionnelle de notre système

Afin d'appliquer efficacement la malléabilité, il est crucial de bien comprendre le fonctionnement global du système. Une approche pour y parvenir consiste à décomposer logiquement le système en deux domaines clés : l'espace des données et l'espace moteur.

La décomposition du système en ces deux domaines met en évidence l'importance de la séparation entre les données et leur traitement. En comprenant cette décomposition logique, le concepteur peut facilement ajuster chaque espace en fonction de ses besoins spécifiques.

### 5.5.1 L'espace motrice

L'espace moteur de notre système se compose de plusieurs composants essentiels. Tout d'abord, nous avons un gestionnaire de contexte(GC) qui est responsable de la capture

et de la classification des données. Ensuite, nous avons un gestionnaire de workflow(GW) qui permet aux utilisateurs de collaborer au sein d'un environnement virtuel collaboratif (EVC) pour accomplir une tâche commune. Enfin, nous avons un gestionnaire de malléabilité(GM) qui interprète les données contextuelles et prend des décisions pour adapter le comportement des objets en fonction du contexte utilisateur et environnemental. En détaillant chaque composant de manière approfondie, nous pourrions mieux comprendre leur fonctionnement et leur utilisation dans le développement d'un environnement virtuel collaboratif malléable.

#### 5.5.1.1 Gestionnaire de workflow(GW)

Le Gestionnaire de workflow joue un rôle essentiel dans la boucle de la RV proposée par Fuchs [47], en assurant la coordination des utilisateurs lors d'activités de groupe telles que la manipulation d'objets, le dialogue et la navigation. Dans notre étude, nous allons intégrer la partie motrice du workflow proposé par [38] (voir la figure 5.5), qui traite les données contextuelles et fournit les outils nécessaires à la collaboration dans l'EVC.

Cette composante comprend des fonctions d'assistance qui utilisent les données de l'espace partagé pour agir sur des objets spécifiques, ce que nous expliquerons en détail dans la section suivante.

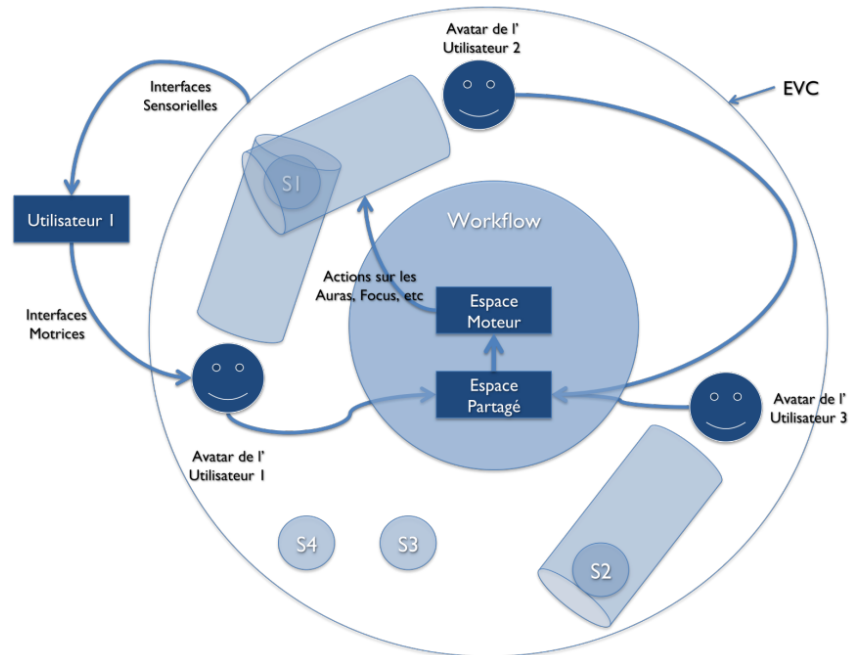


Figure 5.5: Gestionnaire de workflow [38]

## 1. La fonction d'assistance

Les fonctionnalités d'assistance proposées par Domingues [38] sont des outils utilisés par le moteur pour interagir avec des éléments spécifiques tels que le focus, l'aura, l'assistant et l'avatar, en utilisant les informations de l'EVC. L'objectif de ces fonctionnalités est de faciliter diverses tâches d'interaction en 3D, telles que la navigation, la sélection et la manipulation. En agissant sur ces éléments spécifiques, le système peut offrir une assistance aux utilisateurs pour coordonner leurs actions dans l'environnement virtuel collaboratif (EVC).

## 2. La fonction de navigation

La fonction de navigation a pour rôle de modifier la couleur de l'aura en fonction des données de l'EVC afin d'indiquer à l'utilisateur sa position relative par rapport aux objets 3D. Cette fonction permet à l'utilisateur local de visualiser s'il se rapproche ou s'éloigne des objets en observant les changements de couleur de son aura. Pour les utilisateurs distants, cette fonction peut modifier la couleur de l'avatar des autres participants afin

d'indiquer leur position respective par rapport aux sources. Ainsi, chaque utilisateur peut avoir une perception visuelle de la position des sources et des autres participants dans l'environnement virtuel collaboratif (EVC).

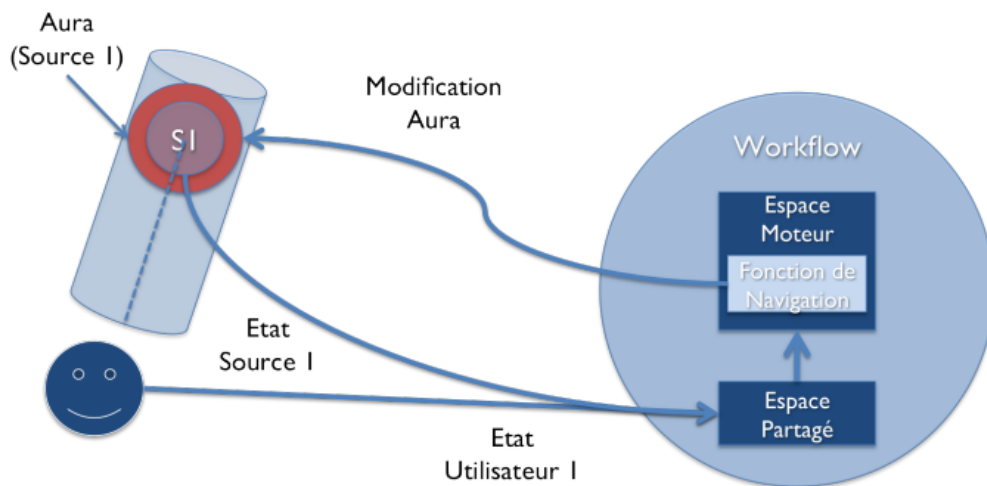


Figure 5.6: La fonction de navigation [38].

### 3. La fonction de sélection

La fonction de sélection est chargée de gérer les sélections effectuées par les utilisateurs. Elle est activée dès que les utilisateurs entrent dans l'aura de l'objet, où un assistant apparaît pour les aider dans la sélection. Une fois la sélection validée, la manipulation de la source devient possible. Cette fonction agit de manière similaire pour les utilisateurs éloignés[38].

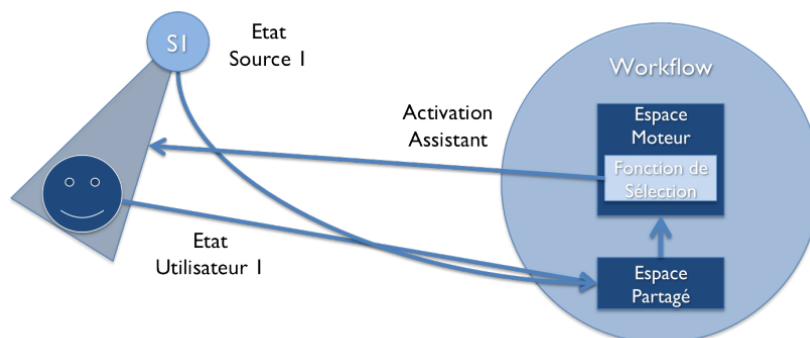


Figure 5.7: La fonction de sélection [38].

## 4. La fonction de manipulation

Domingues a proposé une approche qui intègre des concepts de mécanique classique pour modéliser la manipulation multi-utilisateur à l'aide d'un système mécanique composé de liaisons mécaniques. Les concepts qui s'inscrivent dans l'espace de manipulation de ce modèle sont détaillés dans son travail [38].

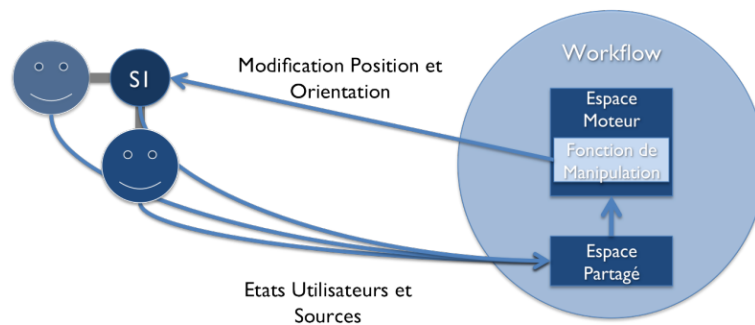


Figure 5.8: La fonction de manipulation [38].

### 5.5.1.2 Gestionnaire de contexte

Le gestionnaire de contexte (GC) est un composant essentiel de notre système, car il est responsable de la collecte, de l'analyse et de la classification des données contextuelles de chaque utilisateur. Son rôle principal est de recueillir des informations pertinentes sur l'utilisateur, telles que sa position, ses actions, ses préférences, son historique, etc. Ces données contextuelles permettent de mieux comprendre les besoins et les préférences de chaque utilisateur.

Le GC joue un rôle crucial pour le gestionnaire de malléabilité, car il fournit les données nécessaires pour déclencher le processus d'adaptation de l'environnement virtuel. Il permet de répondre à des questions telles que "quand s'adapter ?" Le GC (Gestionnaire de contexte) est composé de différentes fonctions clés, parmi lesquelles la fonction de déclenchement, la fonction de capture et la fonction de classification. La fonction de déclenchement joue un rôle crucial dans le processus de manipulation en activant le GC selon un ensemble de conditions préalablement définies dans le contexte de l'utilisateur.

en surveillant en permanence les changements de contexte et en déterminant les moments propices pour apporter des ajustements. Par exemple, le GC peut détecter une nouvelle position de l'utilisateur ou un changement dans son interaction avec les objets virtuels, et déclencher ainsi des adaptations appropriées.

Le GC (Gestionnaire de contexte) est composé de différentes fonctions clés, parmi lesquelles la fonction de déclenchement, la fonction de capture et la fonction de classification. La fonction de déclenchement joue un rôle crucial dans le processus de manipulation en activant le GC selon un ensemble de conditions préalablement définies dans le contexte de l'utilisateur.

## 1. La fonction de déclenchement

Le déclenchement de la malléabilité est une étape fondamentale dans le GM, car elle permet au notre système de détecter les changements de contextes pertinents pour l'adaptation. Le mécanisme de déclenchement utilise différents types de déclencheurs. L'objectif de cette fonction est de répondre à l'exigence proposé, visant à réguler la malléabilité, de sorte qu'elle ne soit pas aléatoire mais conforme à un ensemble de règles, il peut donc répondre à la question Quand lancer le processus de la malléabilité ?

Ces déclencheurs se basent sur des requêtes pour être à l'écoute de tous les changements nécessitant une modification dans le système, afin d'assurer un système malléable et cohérent. En lançant les processus de malléabilité au moment opportun, le système reste toujours réactif aux évolutions du contexte utilisateur et environnemental, et maintient une expérience utilisateur de qualité. Dans un EVCM, les déclencheurs (*trigger* en anglais) sont d'une grande importance pour automatiser les actions en temps réel. Ils permettent d'éviter l'exécution aléatoire des processus d'adaptation. Ainsi, nous proposons trois types de déclencheurs : temporel (DT), conditionnel (DC) et temporel-conditionnel (DCT).

$$F_D = \langle DT, DC, DCT \rangle \quad (5.1)$$

$-F_D$  représente l'ensemble des déclencheurs de l'ECM , DT (déclencheur temporel), DC (déclencheur conditionnel) et DCT (déclencheur temporel-conditionnel).

### 1. Déclencheur temporel :

Le déclencheur temporel est une fonction qui permet de déclencher le processus de malléabilité à un moment précis. Dans le contexte d'un jeu sérieux virtuel collaboratif de rééducation, le système utilise un déclencheur temporel toutes les 15 minutes pour passer automatiquement d'un niveau à un autre plus difficile ou inversement selon le contexte de patient (Algorithme 5.1).

---

**Algorithme 5.1 :** Déclencheur\_Temporel

---

```
while Vrai do  
    | attendre(t);  
    | commencer_malléability();  
end
```

---

Cet algorithme sera exécutée indéfiniment. À chaque itération, l'algorithme attendra un temps "t", puis déclenchera le processus d'adaptation avec la fonction "*commencer\_malléability()*". Le temps d'attente "t" peut être défini en fonction des besoins spécifiques de l'application ou du système sur lequel l'algorithme sera exécuté.

### 2. Déclencheur conditionnel :

Le déclencheur conditionnel permet d'exécuter une ou plusieurs actions si une condition spécifiée est vraie. Cette condition peut être un changement dans une variable ou dans le contenu d'un système, ou une nouvelle donnée capturée. Par exemple, dans un environnement de simulation militaire, un déclencheur peut être programmé pour effectuer des modifications dans l'environnement lorsque le joueur termine un niveau sans complications, en atteignant un certain nombre de répétitions défini par un expert. Un autre déclencheur peut être programmé pour déclencher une action en cas où le joueur présente des signes de stress et d'anxiété,



qui sont détectés par le capteur EEG, afin de réduire la difficulté de l'entraînement(Algorithme 5.2).

---

**Algorithme 5.2** : Déclencheur\_ conditionnel

---

```
while True do  
    | if condition then  
    |     commencer_malléability();  
    | end  
end
```

---

3. **Déclencheurs temporel-conditionnel** :

Dans certaines situations, l'Environnement Virtuel de Connaissance et de Modélisation (EVCM) requiert à la fois un déclencheur temporel et conditionnel. Par exemple, dans le contexte de l'éducation, il peut être bénéfique de programmer un déclencheur qui propose un test à l'élève toutes les 10 minutes afin d'évaluer sa compréhension de la leçon et déterminer si des ajustements sont nécessaires au niveau de l'enseignement. En fonction des résultats, le système peut choisir de maintenir le même niveau d'explication, d'augmenter le niveau d'explication ou, au contraire, de le réduire. Par conséquent, l'utilisation combinée d'un déclencheur temporel et conditionnel permet à l'EVCM d'adapter en temps réel l'expérience d'apprentissage en fonction des besoins et du rythme de chaque apprenant (voir Algorithme 5.3).

---

**Algorithme 5.3** : Déclencheurs\_ temporel\_ conditionnel

---

```
while True do  
    | attendre(t);  
    | if condition then  
    |     commencer_malléability();  
    | end  
end
```

---

## 2. La fonction de capture

La capture des données est un terme utilisé dans les systèmes sensibles au contexte, sous différents noms tels que l'acquisition. Notre système se distingue des modèles existants car il exige l'identification des sources d'informations et du type de données à utiliser en fonction du domaine pour lequel le concepteur souhaite développer un environnement virtuel collaboratif malléable. De plus, notre système impose la définition des appareils à mettre en oeuvre pour obtenir ces informations contextuelles. La conceptualisation de ce terme a été élargie en raison de la complexité des environnements virtuel collaboratif (EVC), qui nécessitent une grande quantité et une grande variété d'informations pour adapter le système aux différents besoins de l'utilisateur.

Pour obtenir les données contextuelles des utilisateurs, notre système requiert un ensemble de sources de données, également connues sous le nom de capteurs. Ces capteurs sont essentiels à notre système, en particulier au gestionnaire de contexte. Ils sont intégrés dans le système et ont pour responsabilité de collecter les données contextuelles. Chaque capteur est associé à un composant logiciel permettant d'accéder aux données capturées et de les transformer en données numériques exploitables par le système. Par exemple, il peut convertir des images en pixels ou des signaux EEG en valeurs numériques, afin de les rendre utilisables par les autres modules du système. Cette étape de normalisation prépare les données pour qu'elles puissent être efficacement exploitées par le système.

## 3. La fonction de classification

Une fois que les données contextuelles ont été capturées à l'aide de divers capteurs, la prochaine étape essentielle pour le gestionnaire de contexte consiste à classer ces données en différents types (voir Figure 5.10). Cela facilite le travail du gestionnaire de malléabilité lors de la phase d'interprétation en regroupant les données dans différentes catégories, telles que les données oculaires, faciales, cinématiques, électriques et environnementales. Pour chaque type de données, nous fournissons une description et des exemples

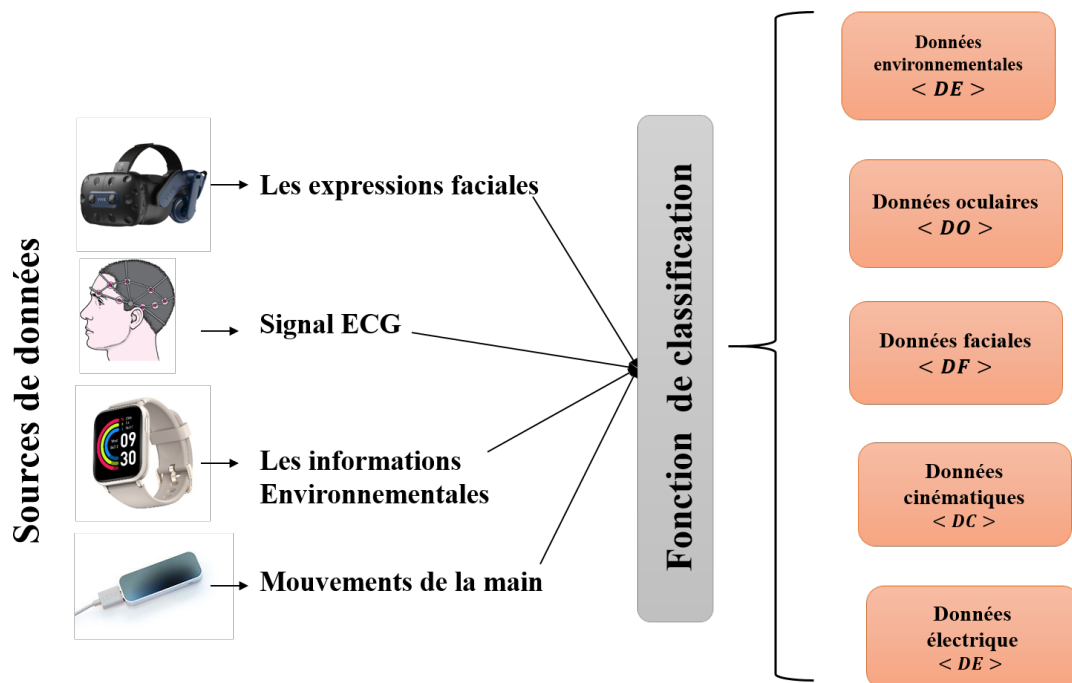


Figure 5.9: Classification des données.

de dispositifs (sources de données) dans le tableau 5.1, en nous référant à la catégorisation proposée par Moon en 2022 [94].

## 1. Les données oculaires (DO) :

Les données oculaires (ou données de suivi oculaire) sont collectées à partir des mouvements des yeux d'un utilisateur à l'aide d'une technologie de suivi oculaire. Ces données peuvent inclure des informations telles que le point de fixation (l'endroit où l'utilisateur regarde), la durée de la fixation, la fréquence des clignements des yeux, les mouvements oculaires rapides (saccades), et bien d'autres encore.

Ces données sont importantes pour adapter l'environnement virtuel car elles peuvent fournir des informations précieuses sur les préférences et les comportements de l'utilisateur, ainsi que sur son niveau d'engagement et d'attention. Ces informations peuvent être utilisées pour personnaliser l'interface utilisateur, améliorer l'efficacité des interactions et optimiser la présentation de l'information. Par exemple, en utilisant les données oculaires, on peut adapter la taille, la position et le contenu des éléments d'interface pour améliorer

l'accessibilité et la convivialité, ou encore pour optimiser la présentation de contenu dans des environnements de formation ou d'apprentissage.

## **2. Les données faciales (DF) :**

Les données faciales sont des informations numériques qui décrivent les caractéristiques du visage d'un individu, telles que la forme du visage, les traits du visage, les expressions faciales et les mouvements des muscles faciaux. Ces données peuvent être collectées à l'aide de caméras et de logiciels de reconnaissance faciale.

Les données faciales revêtent une grande importance dans l'adaptation de l'environnement virtuel, car elles permettent de créer une expérience plus personnalisée pour l'utilisateur. Par exemple, dans le cadre des environnements virtuels de rééducation, l'utilisation des données faciales permet de suivre et d'analyser les mouvements du visage de l'utilisateur, ce qui permet de fournir un feedback en temps réel sur ses performances. Cette approche permet de personnaliser l'environnement virtuel en fonction des besoins spécifiques de l'utilisateur, améliorant ainsi l'efficacité de la rééducation et augmentant sa motivation. Dans notre étude, nous utiliserons des données faciales pour la reconnaissance des émotions et nous profiterons des résultats obtenus pour ajuster le niveau de jeu en conséquence.

## **3. Les données cinématiques (DC) :**

Les données cinématiques sont des données relatives aux mouvements et aux positions des utilisateurs d'un système. Elles sont enregistrées sous forme de données numériques et peuvent être acquises en utilisant différents types de technologies telles que des capteurs de mouvement, des caméras, etc.

Ces données jouent un rôle crucial dans l'adaptation de l'environnement virtuel, en particulier dans des domaines tels que la rééducation et la formation militaire. Dans le domaine de la rééducation, les données cinématiques sont essentielles pour suivre et évaluer les mouvements d'un patient lors d'une séance de rééducation, ce qui permet d'évaluer

ses performances et de lui proposer des exercices adaptés à son amélioration. Dans le domaine militaire, les données cinématiques sont utilisées pour simuler des situations tactiques, offrant ainsi aux militaires la possibilité de s'entraîner dans un environnement virtuel réaliste et de développer en temps réel leurs compétences.

#### **4. Les données électriques (DE) :**

Les données électriques, telles que l'EEG (électroencéphalogramme) et l'ECG (électrocardiogramme), sont des enregistrements des activités électriques du cerveau ou du cœur respectivement. L'EEG mesure l'activité électrique des cellules nerveuses dans le cerveau, tandis que l'ECG mesure l'activité électrique du muscle cardiaque.

Les données électriques sont d'une importance capitale pour adapter l'environnement virtuel dans plusieurs domaines. Dans le domaine de la rééducation, par exemple, l'électroencéphalographie (EEG) peut être utilisée pour mesurer les signaux de contrôle moteur dans le cerveau, ce qui contribue grandement à la réadaptation des personnes souffrant de lésions cérébrales ou de maladies neurodégénératives, en leur offrant un environnement virtuel adapté dynamiquement à leur situation. De même, les données d'électrocardiographie (ECG) peuvent être utilisées pour surveiller la santé cardiovasculaire des patients et concevoir des programmes d'exercices adaptés à leurs besoins spécifiques.

Dans le domaine militaire, les données électriques peuvent être utilisées pour surveiller les niveaux de stress et de fatigue des soldats, permettant ainsi d'optimiser les stratégies de formation et de mission. Cela permet de prendre des décisions éclairées concernant la gestion de la charge de travail et de garantir des conditions optimales pour la performance des soldats sur le terrain. Dans le domaine de l'éducation, l'électroencéphalographie (EEG) peut être utilisée pour mesurer l'engagement des élèves lors des tâches d'apprentissage. Cela permet de comprendre leur niveau d'attention et de concentration, ainsi que leur réactivité émotionnelle, ce qui peut contribuer à adapter les programmes éducatifs en conséquence. L'utilisation de données électriques dans l'éducation peut aider à optimiser l'expérience d'apprentissage des élèves en fournissant des informations précieuses sur leur

état cognitif et émotionnel.

**5. Les données environnementales (DEN) :** Les données environnementales d'un utilisateur dans un Environnement Virtuel de Connaissance (EVC) font référence aux informations collectées sur l'environnement physique dans lequel l'utilisateur interagit avec le système. Ces données englobent des éléments tels que la luminosité, la température, le niveau sonore, l'humidité, la qualité de l'air, et d'autres paramètres similaires.

L'intégration des données environnementales dans un EVC permet de prendre en compte le contexte physique de l'utilisateur. Cela peut être bénéfique pour adapter l'expérience d'apprentissage en fonction des conditions environnementales. Par exemple, si l'environnement est bruyant, le système peut ajuster le volume des instructions pour s'assurer qu'elles sont clairement audibles. De la même manière, si la température est élevée, le système peut suggérer des activités plus calmes et moins physiquement exigeantes afin d'éviter la surchauffe de l'utilisateur.

Il convient de souligner qu'un EVCM peut prendre en compte différents types de données ou seulement quelques-uns, en fonction de ses objectifs. Par exemple, dans un EVCM de shopping, le système doit être capable de suivre la direction du regard de l'utilisateur afin de déterminer les produits qui l'intéressent, et ainsi adapter l'espace commercial en conséquence. En revanche, dans un EVCM de rééducation, le système n'a pas besoin d'informations oculaires, mais il est crucial de suivre les mouvements corporels pour évaluer et corriger les exercices du patient, ainsi que pour adapter l'environnement virtuel à ses capacités physiques. De plus, dans le domaine de la rééducation, le système peut également utiliser des signaux électriques EEG pour surveiller l'activité cérébrale de l'utilisateur et ainsi obtenir des informations sur son état psychologique. En conclusion, chaque EVCM a des besoins et des attentes spécifiques en termes de types de données, qui sont déterminés par ses objectifs et ses utilisateurs.

Le tableau ci-dessous (tableau 5.1) donne quelques exemples de dispositifs qui peuvent

être utilisés pour avoir chaque type de donnée.

Type de donnée	Description	Dispositifs (sources)
<i>Oculaires</i>	l'ouverture des yeux, la taille de la pupille, le clignement, la position des yeux, la cible de regard.	HTC VIVE Pro Eye HMD, Varjo XR/VR-3, Tobii eye-tracking, etc.
<i>Faciales</i>	les mouvements du visage	Webcam, HTC VIVE facial-tracker.
<i>Cinématiques</i>	les mouvements du corps.	Les systèmes de tracking tel que, Azure Kinect, VIVE Tracker (3.0), HTC Vive Cosmos, Meta Quest2, Leap Motion Controller, etc.
<i>Electriques</i>	l'électroencéphalographie (EEG), l'électrocardiogramme (ECG), l'électromyogramme(EMG).	VR(5-channel Muse 2, 5-channel Emotiv Insight, Looxid Link affixed to HTC Vive), Médicaux (EEG(Emotiv, NeuroSky, EmSense, etc), ECG(SmartWatch, PPG Sensor, etc), EMG(EMG Hands, The Myo Armband, etc))
<i>Environnementales</i>	Virtuel les données météorologiques, la localisation. . .	Smartwatch, etc

**Tableau 5.1:** Exemples des capteurs existants pour les différents types de données.

### 5.5.1.3 Gestionnaire de malléabilité

Le gestionnaire de malléabilité regroupe plusieurs fonctionnalités qui permettent d'adapter en temps réel les comportements des objets de l'EVCM en fonction des contextes des utilisateurs avec lesquels ils interagissent lors de la collaboration. Ainsi, il exploite les données contextuelles collectées par le gestionnaire de contexte pour ajuster et modifier les propriétés des objets de l'EVC en utilisant des fonctions d'interprétation, de prise de décision et d'action. Le gestionnaire de malléabilité est constitué de trois espaces, correspondant chacun à l'une des trois tâches d'un système sensible au contexte : l'interprétation, la prise de décision et l'action. Dans l'espace d'interprétation, les données contextuelles organisées sont reçues du gestionnaire de contexte afin de fournir un aperçu de l'état de l'environnement réel. Ces données sont interprétées et donnent un sens à des

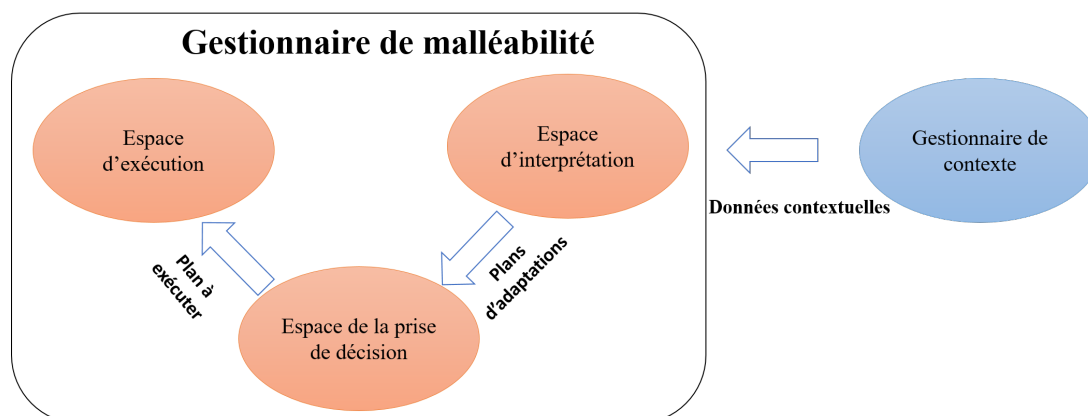


Figure 5.10: Gestionnaire de malléabilité.

informations telles que l'état émotionnel du patient au moment de l'exécution de cette fonction, qu'il soit heureux ou triste, par exemple. Ensuite, les fonctions de l'espace de décision génèrent un ensemble d'actions qui seront exécutées par les fonctions de l'espace d'action (exécution). Chaque espace contient un ensemble de fonctions que nous définissons comme suit :

**a. La fonction d'interprétation** Une fois les données organisées et classifiées reçues du gestionnaire de contexte (GC), les fonctions d'interprétation analysent ces données et les traduisent en informations significatives qui peuvent décrire le comportement physique et psychologique de l'utilisateur. Ces informations seront ensuite utilisées en entrée dans l'espace suivant pour prendre une série de décisions sur les ajustements à apporter dans le système afin de répondre aux comportements des utilisateurs. Il est nécessaire d'assigner une(des) fonction(s) d'interprétation ( $F_I$ ) à chaque type de données, qui prend en compte les données contextuelles classifiées ( $DC$ ) en entrée et renvoie l'état correspondant de l'environnement réel ( $E_{ER}$ ) en sortie, comme l'illustre la fonction 5.2. La structure de ces données est exposée en détail dans la section 5.5.2.2.

$$F_I(DC) = E_{ER} \quad (5.2)$$

Plusieurs techniques d'interprétation de données ont été employées dans les systèmes



adaptatifs. En nous appuyant sur le travail réalisé par Zahabi et al. [151], nous avons établi le tableau 5.2 ci-dessous, qui synthétise les méthodes d'adaptation utilisées dans la littérature.

Approche d'interprétation		Références	Domaines
Optimisation algorithms	-Random search (RS), predictive search(PS).	[97]	Rééducation
	-Monte Carlo optimisation.	[81]	formation de conduite
	-Genetic algorithm.	[88]	Mission-based learning (MBL)
Bayesian network		[58, 115]	Rééducation, formation en électrique
Rule-based system		[14, 15, 57]	Rééducation, interaction sociale pour les enfants autistes, jeux
Machine learning	-Artificiel neural network(ANN)	[12, 61, 95]	Rééducation, formation sociale pour adolescents
	-Support vector machines(SVM).	[145]	Rééducation
	-Lear discriminant analysis(LDA).	[75]	Rééducation
	-Reinforcement learning algorithm.	[141]	Rééducation
	-k-nearest neighbor(KNN).	[67]	Education électronique
	-Random forest algorithm.	[18]	Formation de conduite des autistes
Decision making	-Markov decision process.	[141]	Rééducation
	-Fuzzy inference system.	[111]	formation sur les grues de construction
Conditional statements		[19, 78, 144]	Rééducation, Formation sociale pour les TSA
Self-developed algorithm/language		[89, 143]	Rééducation, formation médicale

**Tableau 5.2:** Quelques méthode d'interprétation utilisées dans la littérature

Après avoir mené nos recherches sur les différentes méthodes d'adaptation existantes, nous avons constaté qu'aucune méthode n'était universellement supérieure aux autres. Chaque auteur utilise une approche spécifique pour atteindre les résultats escomptés. Cependant, au cours des dernières années, la plupart des approches ont été basées sur l'apprentissage automatique, en raison de sa grande diffusion dans de nombreux domaines

tels que la médecine, les jeux vidéo, le domaine militaire, le sport et la réalité virtuelle. Les méthodes d'apprentissage automatique sont connues pour leur efficacité en réalité virtuelle, notamment pour la création d'expériences qui se rapprochent de la réalité grâce à des avatars réalistes dotés de mouvements réalistes.

Il est également important de souligner que la plupart des dispositifs récents intègrent des composants facilitant l'utilisation de l'intelligence artificielle, tels que la reconnaissance faciale, le suivi oculaire, le suivi des mouvements corporels et la détection de signaux EEG pour chaque utilisation (comme présenté dans le tableau 5.1)

En somme, même si la méthode d'adaptation pourrait être intégrée dans le capteur lui-même, nous avons préféré la séparer dans notre étude pour plus de généralité. Par exemple, si une webcam est utilisée, le concepteur devra déterminer la méthode de suivi oculaire à utiliser. Toutefois, l'utilisation d'un casque HTC VIVE Pro Eye HMD ne nécessite pas de choix de méthode de suivi, car cela est assuré par le dispositif lui-même. Néanmoins, il est toujours possible de proposer une autre méthode si celle intégrée dans l'appareil ne convient pas à l'objectif recherché.

**b. La fonction de décision** À cette étape, le système évalue l'état de l'environnement virtuel collaboratif (EVC) et de l'environnement réel (ER) en utilisant des ensembles de règles et de tests pour déterminer s'il nécessite une adaptation. Si une adaptation est jugée nécessaire, trois types de décideurs peuvent intervenir pour prendre des décisions sur les actions à exécuter.

1. Dans le cas où le système est en charge du processus de malléabilité, celui-ci sera automatique.
2. Dans le cas où l'expert est en charge du processus de malléabilité, celui-ci sera manuel.
3. Dans le cas où le système propose des actions, l'expert choisi les actions proposées et le processus de malléabilité sera semi-automatique.

Ainsi, la fonction de décision ( $F_D$ ) prend en entrée l'état de l'EVC ( $E_{EVC}$ ) et de l'environnement réel ( $E_{ER}$ ), puis elle produit en sortie un ensemble d'actions ( $A_i$ ) à exécuter dans l'EVCM, comme démontré dans la fonction ci-dessous.

$$F_D(E_{ER}, E_{Evc}) = A_i \quad \text{avec } i = 1..n \quad (5.3)$$

Pour l'aide à la décision, la fonctionnalité "Nimbus" (expliquée dans la partie 5.5.2.1) est essentielle dans les environnements collaboratifs où un objet peut être partagé par un ou plusieurs utilisateurs. Cette fonctionnalité permet d'adapter le comportement de l'objet en fonction des utilisateurs qui le visualisent. Avant de décider des actions à effectuer, le système doit avoir le "nimbus" de chaque objet, c'est-à-dire une liste des utilisateurs qui le partagent. Par exemple, si deux utilisateurs préfèrent des couleurs différentes, l'un préférant le jaune et l'autre le rouge, les objets partagés entre les deux utilisateurs doivent être de couleur orange pour satisfaire les deux utilisateurs. Le Nimbus est l'ensemble des utilisateurs souhaitant interagir avec une source(objet virtuel). Il représente le groupe qui pourrait sélectionner la source pour la manipuler ultérieurement

**c. La fonction d'action (exécution)** La phase d'action de la malléabilité revêt une grande importance dans le processus global de malléabilité. Elle intervient après la sélection des modifications à apporter au système et peut impliquer diverses tâches telles que la modification du contenu 3D de l'EVC, la mise à jour des protocoles de collaboration et le changement des techniques d'interaction. Cette étape est cruciale pour assurer que le système s'adapte de manière dynamique aux changements du contexte tout en offrant une expérience utilisateur optimale.

La fonction 5.4 ci-dessous prend en entrée un ensemble d'actions  $A_i$  générées par les fonctions de l'espace de décision, ainsi que l'état de l'environnement virtuel. En suivant un ensemble de règles, cette fonction vérifie si l'environnement virtuel collaboratif (EVCM) a besoin d'une adaptation. Si tel est le cas, elle exécute les actions correspondantes, rendant

ainsi l'EVCM malléable.

$$F_A(A_i, E_{Evc}) = EVCM \quad (5.4)$$

Dans le contexte de la plasticité 3D, Jérémy a identifié dans [80] cette étape comme une étape cruciale qui constitue la cible de la plasticité. Il a proposé de la subdiviser en trois catégories distinctes : l'adaptation de contenu et l'adaptation des techniques d'interaction et enfin l'adaptation de système . Toutefois, dans le cadre de notre étude, nous considérons également le protocole de collaboration comme un élément essentiel de notre approche. Nous avons donc éliminé la caractéristique d'interopérabilité vue comme notre approche s'oriente toujours vers le contexte réel de l'utilisateur par le matériel qu'il utilise mais sa reste très important, et c'est possible de l'ajouter sans problème , nous avons donc la remplacer par une autre catégorie qui est l'adaptation de collaboration. Dans la suite de cette section, nous décrirons avec des exemples chacune de ces catégories d'adaptation pour mieux comprendre leur fonctionnement et leur impact sur l'EVCM.

### 1. Adaptation de contenu 3D

L'adaptation de contenu 3D en temps réel dans les environnements virtuels collaboratifs (EVC) est un processus qui consiste à ajuster dynamiquement le contenu 3D. Cette adaptation peut être réalisée de différentes manières en fonction des besoins de l'utilisateur et des objectifs de l'environnement virtuel. Voici quelques exemples d'adaptation de contenu 3D dans différents domaines :

- **Jeux vidéo** : Dans les jeux vidéo en ligne multijoueurs, l'adaptation de contenu 3D peut être utilisée pour ajuster la difficulté du jeu en temps réel en fonction du niveau de compétence des joueurs. Par exemple, si un joueur novice commence à jouer, le contenu 3D peut être simplifié pour permettre une expérience de jeu plus simple, tandis que pour un joueur avancé, le contenu 3D peut être plus complexe pour offrir un défi plus difficile.
- **Formation professionnelle** : Dans les environnements de formation professionnelle, l'adaptation de contenu 3D peut être utilisée pour offrir une expé-

rience d'apprentissage personnalisée en fonction des besoins de l'apprenant. Par exemple, si un apprenant a des difficultés à comprendre un concept particulier, le contenu 3D peut être adapté pour fournir des informations supplémentaires et faciliter la compréhension.

- **Architecture et design** : Dans les environnements de conception architecturale, l'adaptation de contenu 3D peut être utilisée pour permettre aux utilisateurs de visualiser leur conception en temps réel et de l'ajuster en fonction de leurs besoins. Par exemple, les architectes peuvent ajuster les dimensions d'un bâtiment pour voir comment cela affecte l'apparence globale, et les designers peuvent ajuster les couleurs et les textures en temps réel pour créer le meilleur design possible.
- **Simulation industrielle** : Dans les simulations industrielles, l'adaptation de contenu 3D peut être utilisée pour offrir une expérience de simulation plus réaliste. Par exemple, les simulateurs de vol peuvent ajuster les conditions météorologiques en temps réel pour simuler des situations de vol réalistes, tandis que les simulateurs de conduite peuvent ajuster la vitesse et le trafic pour offrir une expérience de conduite plus réaliste.
- **La rééducation** : les EVC peuvent être utilisés pour la rééducation motrice. Par exemple, les patients atteints d'une lésion de la moelle épinière peuvent bénéficier d'un environnement virtuel qui simule des tâches quotidiennes telles que la marche ou la montée d'escaliers. L'adaptation de contenu 3D en temps réel permet de personnaliser l'environnement en fonction des besoins individuels du patient, en ajustant les paramètres tels que la vitesse de marche ou la hauteur des marches en temps réel. De plus, le thérapeute peut suivre en temps réel les mouvements du patient dans l'environnement virtuel et adapter les exercices en fonction de ses progrès.
- **L'éducation** : les EVC peuvent être utilisés pour fournir des expériences d'apprentissage immersives. Par exemple, un environnement virtuel peut simuler

une visite guidée d'un site historique ou d'un musée, offrant ainsi aux étudiants une expérience d'apprentissage interactive et engageante. L'adaptation de contenu 3D en temps réel permet de personnaliser l'environnement en fonction des préférences et des besoins d'apprentissage de chaque étudiant, en ajustant la difficulté ou le niveau de détail des informations présentées en temps réel. De plus, les enseignants peuvent suivre en temps réel les progrès des étudiants dans l'environnement virtuel et adapter leur enseignement en fonction de leurs performances.

## 2. Adaptation des techniques d'interaction

Il existe plusieurs façons d'adapter les techniques d'interaction aux contextes des utilisateurs dans un environnement virtuel collaboratif (EVC), voici quelques exemples :

- **Utilisation de la reconnaissance vocale ou gestuelle** : Si l'utilisateur qui a un trouble neurologique courant qui se caractérise par des tremblements involontaires et rythmiques des mains, l'utilisation des mallettes peut être difficile. Dans ce cas, la reconnaissance vocale ou gestuelle peut être utilisée pour permettre à l'utilisateur de communiquer avec l'environnement virtuel sans avoir à utiliser les mains. Par exemple, l'utilisateur peut donner des commandes verbales pour naviguer dans l'environnement virtuel.
- **Adaptation du niveau de détail** : L'environnement virtuel peut s'avérer trop complexe pour les utilisateurs novices en réalité virtuelle (VR). Dans de tels cas, il est possible d'ajuster le niveau de détail en fonction des besoins de chaque utilisateur. Par exemple, un utilisateur novice peut bénéficier d'une interface plus simple et moins détaillée, tandis qu'un utilisateur expérimenté peut nécessiter une interface plus complexe. En adaptant le niveau de détail, on veille à ce que chaque utilisateur puisse interagir de manière efficace et confortable avec l'environnement virtuel, favorisant ainsi une meilleure expérience utilisateur.

## 3. Adaptation des protocoles de collaboration

En effet, cette adaptation découle des deux premiers types mentionnés précédemment. Toutefois, si deux utilisateurs collaborent pour manipuler un objet unique et utilisent des techniques d'interaction différentes, il est crucial d'ajuster ces techniques pour qu'elles soient compatibles. Dans ce cas, il peut être nécessaire de modifier les propriétés de l'objet, comme sa masse ou sa taille. L'objectif est de faciliter la collaboration entre les utilisateurs et de leur permettre d'effectuer leurs tâches en toute fluidité. Prenons l'exemple de deux patients qui participent à un environnement virtuel collaboratif pour résoudre des puzzles en 3D. Le premier patient utilise des gestes de saisie et de lâcher pour manipuler les objets, tandis que le deuxième patient utilise des gestes de rotation pour les orienter correctement. Pour avancer dans le jeu, ils doivent collaborer et résoudre chaque puzzle ensemble. Pour faciliter leur collaboration, il est nécessaire d'adapter les techniques d'interaction. Par exemple, le comportement des objets peut être ajusté pour faciliter leur manipulation pour les deux patients, et les gestes de rotation peuvent être optimisés pour être plus intuitifs pour le premier patient. Grâce à cette adaptation, les deux patients peuvent travailler ensemble efficacement et atteindre leur objectif commun.

### 5.5.2 L'espace de données

L'espace de données (ED) est une composante fondamentale du processus de malléabilité. Il est divisé en deux sous-espaces qui représentent respectivement l'état de l'environnement réel ( $E_{ER}$ ) et de l'environnement virtuel collaboratif ( $E_{EVC}$ ). La combinaison de ces deux espaces permet de caractériser l'état global du système et offre une vision complète de l'EVC et de l'ER à tout moment.

C'est grâce à cette connaissance approfondie que le gestionnaire de malléabilité peut recevoir en continu des données contextuelles. Ces données lui permettent de réagir de manière appropriée et de maintenir la cohérence entre le monde réel et le monde virtuel.

Ainsi, la matrice d'état est définie par l'équation 5.5, garantissant ainsi une représentation précise et complète de l'état du système.

$$ED = \langle E_{EVC}, E_{ER} \rangle \quad (5.5)$$

Dans la figure 5.11 ci-dessous, nous donnons un exemple concret de ces deux espaces dans le contexte d'un jeu sérieux de rééducation basé sur les expressions faciales. Les données relatives aux objets dans l'EVC (leur comportement, leur interaction avec les utilisateurs, etc.) ainsi que les données de collaboration (le nimbus, le focus, etc.) sont stockées dans l'espace de données de l'EVC et sont constamment mises à jour pour assurer une flexibilité et une cohérence optimales entre le monde réel et virtuel. Les informations concernant les expressions faciales sont stockées dans l'espace de données d'ER. Par exemple, pour l'utilisateur 1 à l'instant t, un vecteur de sept émotions est stocké [joie= $x_1$ , tristesse= $x_2$ , peur= $x_3$ , colère= $x_4$ , dégoût= $x_5$ , surprise= $x_6$ , neutralité= $x_7$ ]. Chaque valeur représente le pourcentage de de chaque émotion de cet utilisateur à l'instant t.

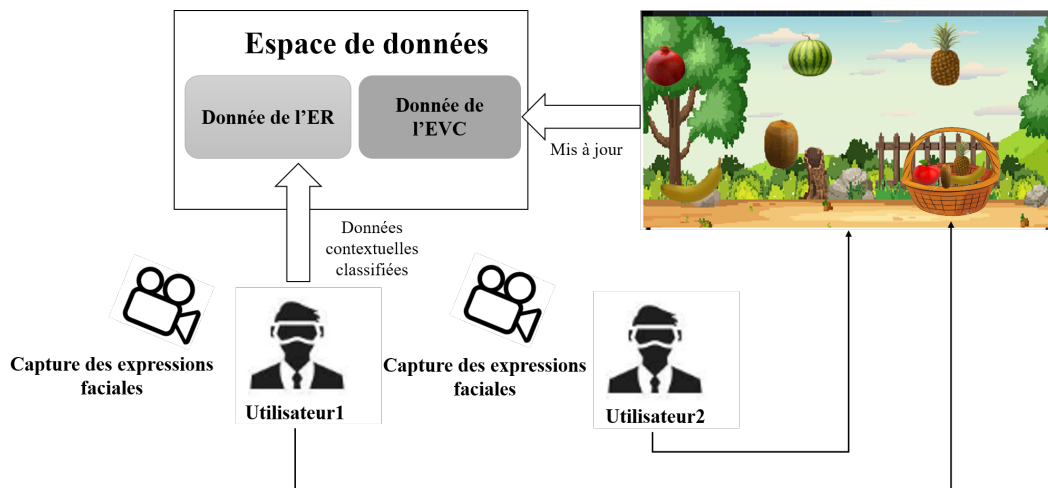


Figure 5.11: Illustration de l'espace de données

### 5.5.2.1 L'espace des données de l'EVC

Pour décrire l'état global de l'EVC, on utilise trois matrices. La première correspond à l'état des sources particulières ( $ES$ ), la deuxième à celui de tous les avatars ( $EA$ )



présents dans l'EVC, et la dernière représente l'état de tous les objets (*EO*) dans cet environnement. En résumé, l'état d'un EVC peut se représenter par cet ensemble :

$$E_{EVC} = \langle EA_i, EO_j, EA_k \rangle \quad \text{avec } i = 1..n \quad \text{et } j = 1..m \quad \text{et } k = 1..p \quad (5.6)$$

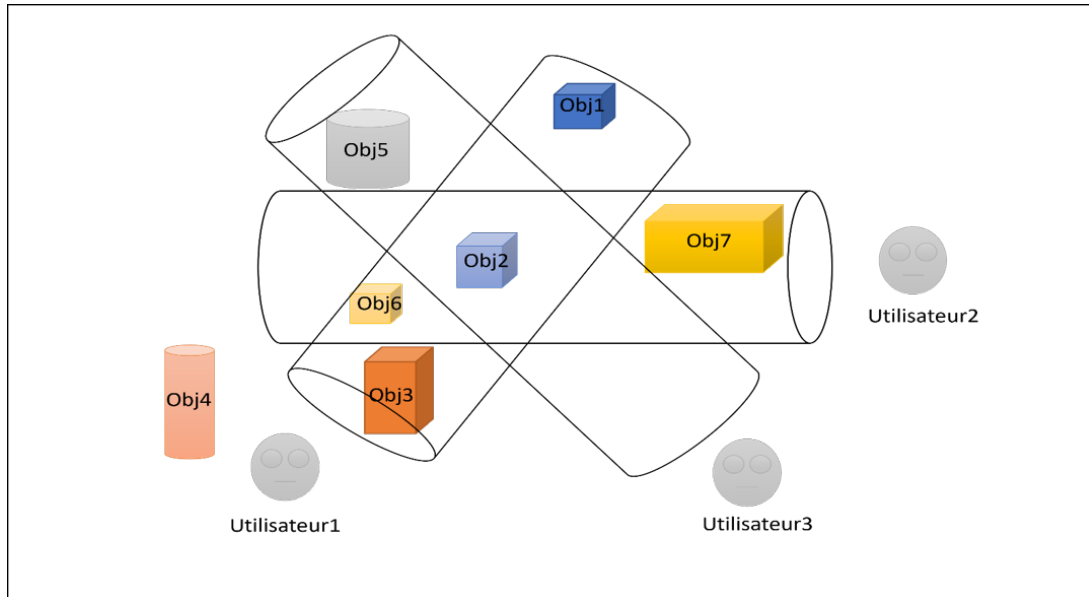
### a- Les sources particulières

Selon [38], les sources particulière sont définies pour négocier l'interaction entre les utilisateurs, en fonction de leur position et orientation. Elles comprennent l'aura, le focus, le nimbus, l'assistant et l'avatar. Les utilisateurs sont caractérisés par des vecteurs d'état, qui alimentent le moteur de workflow pour agir sur ces sources et modifier la perception dans l'Environnement Virtuel Collaboratif (EVC). L'objectif est de détecter les actions des participants et d'améliorer la collaboration. Les sources particulière dans l'EVC sont constituées de plusieurs éléments :

1. Le focus représente les objets que l'utilisateur peut interagir et qui sont dans son champ de vision. Si deux utilisateurs ont un focus commun, ils peuvent interagir sur la même source.
2. Le nimbus est l'ensemble des utilisateurs qui souhaitent interagir sur un objet. Il permet de visualiser le groupe qui peut potentiellement sélectionner l'objet.
3. L'aura représente la zone autour d'une source qui permet à un ou plusieurs utilisateurs de la sélectionner. Si l'avatar de l'utilisateur se trouve dans l'aura de l'objet, il peut la sélectionner.
4. L'assistant est un objet virtuel qui aide l'utilisateur à réaliser des actions spécifiques, comme la sélection d'une source. Il peut prendre la forme d'un guide visuel, moteur ou combiné pour aider l'utilisateur à atteindre sa cible.
5. L'avatar est la représentation de l'utilisateur dans l'environnement virtuel collaboratif. Il permet de visualiser sa position et son orientation par rapport aux autres

utilisateurs et dans l'environnement.

Dans le cadre de notre recherche, nous allons utiliser le Nimbus comme étude de cas. Nous allons vous présenter un exemple concret afin de mieux le décrire (figure 5.12).



**Figure 5.12:** Exemple illustratif d'utilisation de la fonction Nimbus dans un EVC.

Dans la figure ci-dessus, considérons trois utilisateurs partageant un EVC. Les avatars de chaque utilisateur sont représentés en gris. Chaque utilisateur possède un champ de vision qui lui permet de voir et de manipuler des objets spécifiques. Ces objets peuvent différer d'un utilisateur à un autre, mais il peut y avoir des moments où deux utilisateurs visualisent les mêmes objets.

L'objet	Nimbus
Objet1	
Objet2	U1,U2,U3
Objet3	U1
Objet4	/
Objet5	U3
Objet6	U2,U1
Objet7	U2

**Tableau 5.3:** Nimbus de chaque objet dans l'EVC

C'est là qu'intervient l'importance de Nimbus. Prenons l'exemple de l'objet 2, qui est visualisé par U1, U2 et U3. Son comportement doit donc être adapté simultanément pour les trois utilisateurs.

### **b- L'état des avatars**

L'état de l'avatar  $EA_i$  est représenté par un couple  $(idA_k, UA_k)$  dans lequel  $idA_k$  correspond à l'identifiant de l'avatar de l'utilisateur dans l'EVC et  $UA_k$  représente le vecteur d'état de l'utilisateur. Ce vecteur est construit à partir des éléments fournis par le travail de Domingues [38]. Plus précisément, les éléments du vecteur  $UA_k$  comprennent :

- $PA_k$  : la position de l'avatar ;
- $OA_k$  : l'orientation de l'avatar ;
- $FA_k$  : la force exercée par l'utilisateur via l'avatar ;
- $VA_k$  : la vitesse de l'avatar ;
- $mA_k$  : la masse virtuelle de l'avatar.

**c- L'état des objets** Domingues utilise les mêmes éléments pour décrire les avatars  $A_i$  et les objets  $O_i$ . Notre travail se distingue de celui de Domingues en ce sens que nous cherchons à maximiser l'information disponible pour adapter l'EVC selon le contexte utilisateur et garantir une grande flexibilité de l'environnement. Pour cela, chaque objet de l'EVC est caractérisé par :

- $PG$  : les propriétés géométriques ;
- $PPh$  : les propriétés physiques ;
- $PV$  : les propriétés visuelles ;
- $PI$  : les propriétés interactives.

1. **Les propriétés géométriques** : elles comprennent la forme, la taille, l'orientation et la position de l'objet dans l'espace virtuel.
2. **Les propriétés physiques** : incluent la masse, la densité, la friction, l'élasticité, la viscosité, etc.

3. **Les propriétés visuelles** : comprennent la couleur, la texture, le niveau de transparence, etc
4. **les propriétés interactives** : elles définissent comment un objet virtuel réagit aux interactions de l'utilisateur, telles que l'utilisation d'un leap motion.

Ainsi, chaque objet est représenté par un couple  $(idO_s, PO_s)$  où  $idO_s$  est l'identifiant de l'objet et  $PO_s$  est un vecteur de propriétés spécifiques à l'objet, qui est représenté dans l'équation 5.7.

Chaque objet dans l'EVC peut être caractérisé par un ensemble de propriétés sélectionnées en fonction du contexte de l'application. Par exemple, le concepteur peut avoir besoin d'adapter la vitesse de l'objet mais pas nécessairement de sa couleur. En d'autres termes, les objets ne possèdent pas nécessairement toutes les propriétés citées précédemment.

$$PO_s = \langle PG_{O_s}, PPh_{O_s}, PV_{O_s}, PI_{O_s} \rangle \quad (5.7)$$

### 5.5.2.2 L'espace des données de l'ER

Il est primordial d'obtenir autant de données que possible sur l'utilisateur afin que le système puisse s'adapter au mieux à ses besoins et à son contexte. Les données contextuelles  $(U_i)$  telles que les données faciales  $(DF_i)$ , électriques  $(DE_i)$ , oculaires  $(DO_i)$  et cinématiques  $(DC_i)$ , ainsi que les données environnementales de l'utilisateur  $(E_i)$ , sont toutes essentielles pour comprendre et analyser le comportement de l'utilisateur dans l'environnement virtuel collaboratif. Toutefois, chaque système doit sélectionner les données sur lesquelles il doit se baser pour adapter l'EVC en fonction de ses objectifs et de son contexte d'utilisation. Dans la partie 5.5.1.2, nous avons fourni une explication détaillée de chaque type de données contextuelles et environnementales, et nous les avons également illustrées avec des exemples pour démontrer leur importance dans le développement d'un

EVC malléable. L'équation de cet espace est présentée par l'équation :

$$E_{ER} = \langle U_i, E_i \rangle \quad (5.8)$$

Ainsi, l'état de l'environnement réel  $E_{ER}$  est la combinaison de données contextuelles de l'utilisateur ( $U_i$ ) et de données environnementales ( $E_i$ ), comme montré dans l'équation 5.9 :

$$U_i = \langle DO_i, DF_i, DC_i, DE_i \rangle \quad (5.9)$$

Ainsi, une décomposition logique du système est indispensable pour séparer les différentes fonctions en plusieurs parties clés, simplifier leur gestion et leur développement, et permettre au système de traiter efficacement ces différentes données. La contribution proposée répond pleinement aux trois dimensions décrites dans le chapitre 2. La dimension "collection" est prise en compte par le gestionnaire de workflow, qui utilise les données stockées dans l'espace des données. La dimension "malléabilité" est assurée par le gestionnaire de malléabilité et le gestionnaire de contexte. Les exigences liées à la dimension de la "réalité virtuelle" sont satisfaites d'une part par le développement d'un système de réalité virtuelle (RV) et l'utilisation de dispositifs VR. D'autre part, elles sont également comblées en permettant une interaction 3D collaborative efficace. L'immersion et la présence sont également assurées en maintenant un haut niveau de malléabilité. En effet, plus il y a de cohérence entre le monde virtuel et le monde réel, plus l'utilisateur ressent la présence et l'immersion. Cette dimension est donc étroitement liée aux deux premières dimensions.

## 5.6 MVC-3DCM : Modèle d'architecture de notre système

Dans les sections précédentes, nous avons détaillé chaque composant du système. En les regroupant au sein d'une architecture unique, nous pouvons bénéficier de leurs avantages

respectifs, avoir une vision globale du système et mieux appréhender les interactions entre ses différentes parties. Cette intégration sera basée sur le modèle Clock présenté dans la partie 3.9. À partir de ce modèle, nous proposons une extension appelée MVC-3DCM (Modèle-vue-contrôleur 3D collaboratif malléable), comme illustré dans la figure 5.13..Par rapport à d'autres architectures, le modèle MVC présente l'avantage de pouvoir séparer efficacement les différentes parties du système et faciliter leur intégration.

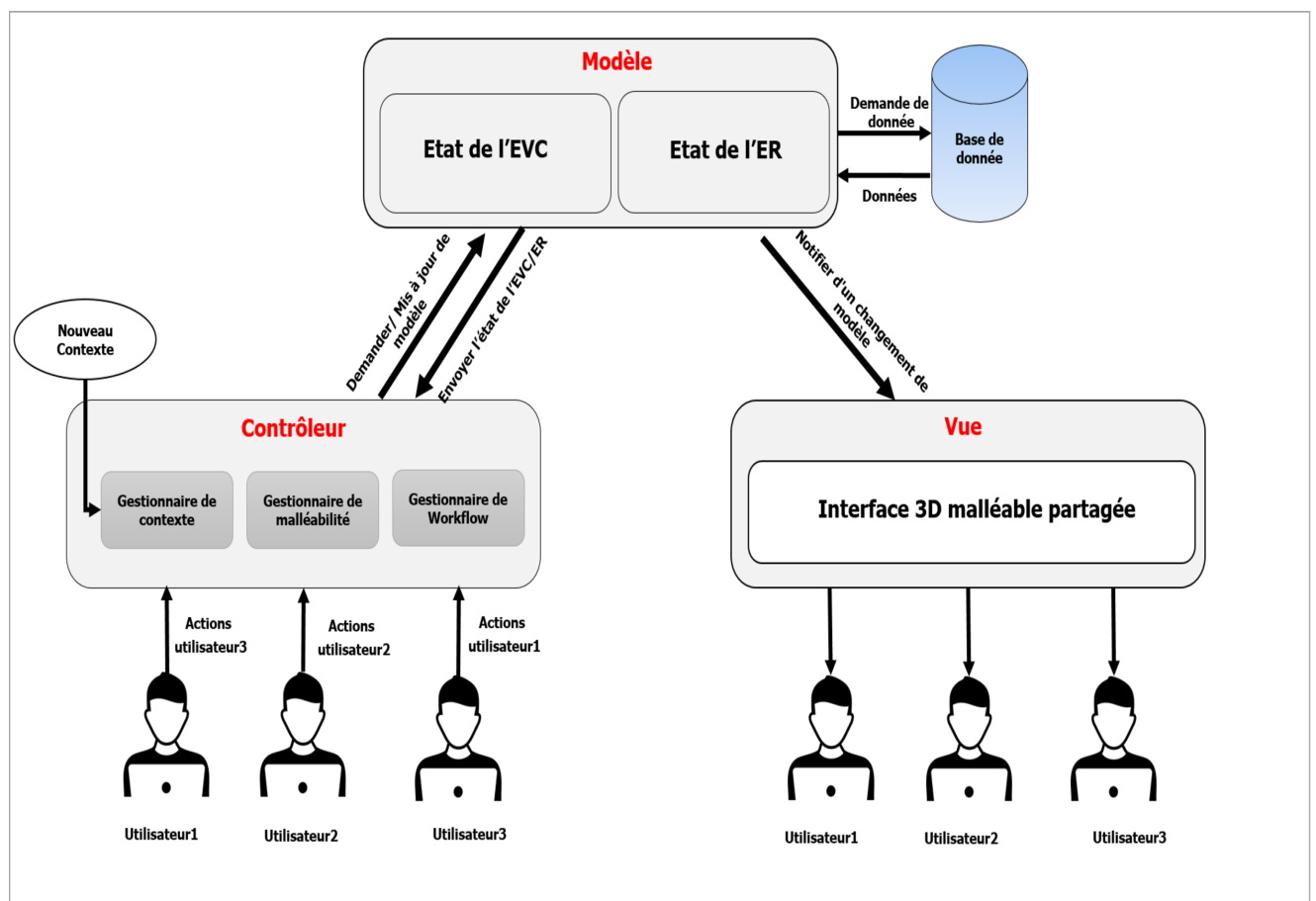


Figure 5.13: MVC-3DCM : Modèle d'architecture de notre système

Notre architecture MVC-3DCM se compose de trois parties distinctes : le contrôleur, le modèle et la vue.

1. **Le contrôleur** : il englobe le gestionnaire de malléabilité, le gestionnaire de contexte et le gestionnaire de workflow. Ce modèle est responsable de la gestion des entrées des utilisateurs, de l'analyse, de l'interprétation et de la génération des

actions à appliquer dans le système, ainsi que de leur exécution. De plus, il joue un rôle essentiel dans la gestion de la collaboration au sein de l'EVC.

2. **Le modèle** : contient l'état de l'EVC et de l'ER, et est lié à une base de données pour stocker les historiques de malléabilité. C'est l'espace où le système stocke les données nécessaires à la gestion de la malléabilité et à la gestion du workflow.
3. **La vue** : elle comprend l'interface 3D malléable partagée, adaptée à chaque utilisateur

Le gestionnaire de contexte capte et classe les données, et met ensuite à jour l'état de l'ER. Le gestionnaire de malléabilité demande l'état de l'ER et de l'EVC afin de tester la cohérence entre les deux mondes. Si la cohérence est maintenue, l'état de l'EVC reste inchangé. Dans le cas contraire, le gestionnaire de malléabilité effectue les modifications nécessaires sur l'état de l'EVC. Ensuite, le modèle notifie la vue pour que les interfaces utilisateur soient mises à jour. Le gestionnaire de workflow travaille en parallèle en fournissant une assistance aux utilisateurs pour permettre l'interaction 3D à plusieurs, et il met à jour les sources particulières (focus), qui est un élément de base dans le processus de malléabilité pour adapter un contenu partagé à tous les utilisateurs qui le visualisent.

Cette architecture présente plusieurs avantages, notamment une séparation claire des responsabilités entre les différents composants. Le contrôleur gère les entrées de l'utilisateur et les actions à effectuer, le modèle contient l'état du système et les opérations possibles sur cet état, tandis que la vue fournit une représentation visuelle malléable aux utilisateurs.

## 5.7 Modélisation

Lorsque l'on conçoit un système informatique, il est crucial de disposer d'une représentation claire et précise de ses fonctionnalités, de sa structure et de ses interactions. Cela permet de mieux comprendre les besoins du système et de faciliter sa mise en œuvre. C'est pourquoi la modélisation UML (Unified Modeling Language) est souvent utilisée

pour décrire les différents aspects d'un système. Ce langage de modélisation graphique standard fournit une représentation visuelle et facilement compréhensible des composants et des interactions du système, ce qui facilite la communication entre les différentes parties prenantes et permet de détecter plus rapidement les erreurs de conception pendant le processus de développement.

### 5.7.1 Diagramme de classe de notre système

Nous avons organisé notre diagramme de classe en utilisant la même logique que celle présentée dans la section précédente, à savoir le modèle MVC. Le diagramme est divisé en trois parties : les classes responsables de la gestion (le contrôleur) en gris, les classes responsables des données (le modèle) en bleu et celles de l'affichage (la vue) en orange comme illustré dans la figure 5.14.

La classe *GestionnaireWorkflow* s'occupe de modifier les *ObjetParticulier* (voir la section 5.5.2.1) qui seront ensuite utilisés par le gestionnaire de malléabilité. Elle comporte également un ensemble de fonctions telles que *CollNav*, *CollNavig*, et *CollManip* qui assurent l'assistance nécessaire dans l'EVC pour garantir une collaboration efficace. La classe abstraite *GestionnaireContexte* orchestre la collecte de données à partir de différentes sources de données. Elle est composée d'un ou plusieurs Déclencheurs qui disposent de méthodes pour déclencher des événements de manière conditionnelle, temporelle ou conditionnelle/temporelle, ainsi que pour ajouter ou supprimer un déclencheur. La classe *GestionnaireContexte* a la capacité d'arrêter la collecte de données de tous les types (*DéclencherCollection(Déclencheur)*) ou de types spécifiques (*DéclencherCollectionParType(Déclencheur, Type)*), de classer les données (*ClassifierDonnée*) et d'écrire les données dans un fichier *CSV*. Le développeur peut choisir quels types de données *TypeContexte* utiliser, et il peut ajouter un autre type de données en créant un gestionnaire héritant de la classe *GestionnaireContexte*. Chaque type possède un gestionnaire, dans notre cas, *GestionnaireOcContexte*, *GestionnaireFacContexte*, *GestionnaireCinCon-*





*texte*, *GestionnaireElContexte*, *GestionnaireEnvContexte*. Chaque type doit se connecter à un ou plusieurs capteurs pour acquérir des données, il doit donc être associé à une liste d'objets dans la classe *ContexteData*. Chaque gestionnaire possède des fonctions pour l'analyse et l'interprétation des données. Dans notre modélisation, nous voyons que chaque gestionnaire doit être associé à sa fonction *InterpréterDonnée*.

La classe *GestionnaireMalleabilite* utilise les données stockées dans chaque *buffer* des gestionnaires de contexte ainsi que l'état de l'EVC pour analyser les propriétés des objets de l'EVC, à savoir les *ObjetVR* et les objets particuliers *ObjetParticulier* décrits dans la section précédente. Cette classe teste la cohérence entre le contexte réel de l'utilisateur et son monde virtuel. La fonction *PrendreDecision* retourne un ensemble d'actions à exécuter, ces décisions ayant été prises en analysant le *focus* de chaque objet. Le comportement de chaque objet doit être adapté à l'ensemble des utilisateurs qui les visualisent. Ensuite, ces actions seront exécutées afin d'adapter la vue Interface3D aux contextes des utilisateurs.

### 5.7.2 Architecture réseau de notre système

Dans une architecture client-serveur d'un EVCM, le serveur occupe une place centrale car il gère à la fois le modèle, la malléabilité et le workflow. En effet, afin d'avoir une vue générale sur les données des utilisateurs, il est nécessaire de les centraliser au niveau du serveur. De même, pour que le système puisse s'adapter à tous les utilisateurs de manière efficace, le gestionnaire de malléabilité doit également être centralisé.

Cependant, afin de réduire la charge sur le serveur central et d'améliorer le temps de réponse lors de la collecte des données, chaque client dispose également d'un gestionnaire de contexte qui s'occupe des données contextuelles de l'utilisateur en question. Cette approche distribuée permet de déléguer certaines tâches aux clients et d'alléger la charge sur le serveur central.

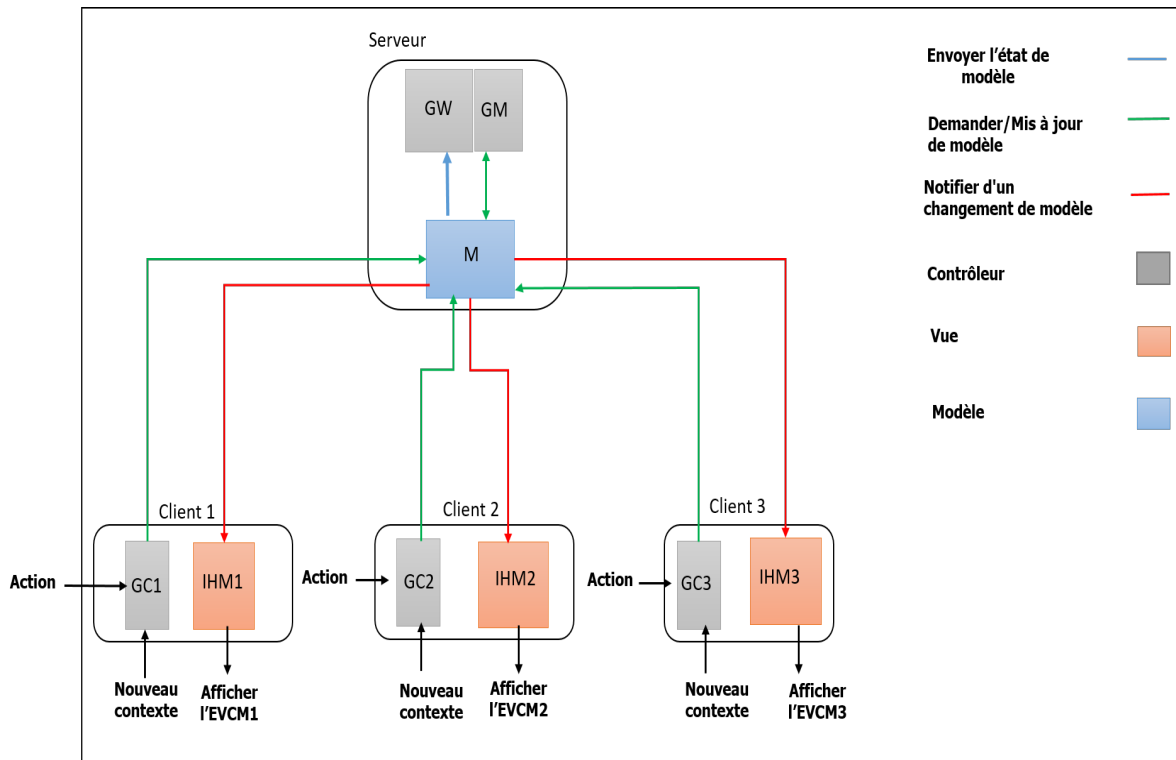


Figure 5.15: Architecture réseau de notre système

En conclusion, l'utilisation d'une architecture client-serveur pour un EVCM est de centraliser les données importantes tout en répartissant les tâches et la charge de travail entre les clients et le serveur. Cela permet de garantir une meilleure efficacité et réactivité du système dans son ensemble.

### 5.7.3 Diagramme d'activité d'un scénario de collaboration

Dans le cadre de notre exemple de collaboration entre deux utilisateurs, le système doit faire face à un changement de contexte de la part de l'utilisateur 1. Pour mieux comprendre cette situation, nous pouvons nous référer au diagramme d'activité présenté dans la figure 5.16. Le système est composé de trois acteurs : un serveur contenant le modèle, un gestionnaire de workflow et un gestionnaire de malléabilité, ainsi que deux clients (client 1 et client 2) équipés d'une interface 3D et d'un gestionnaire de contexte.

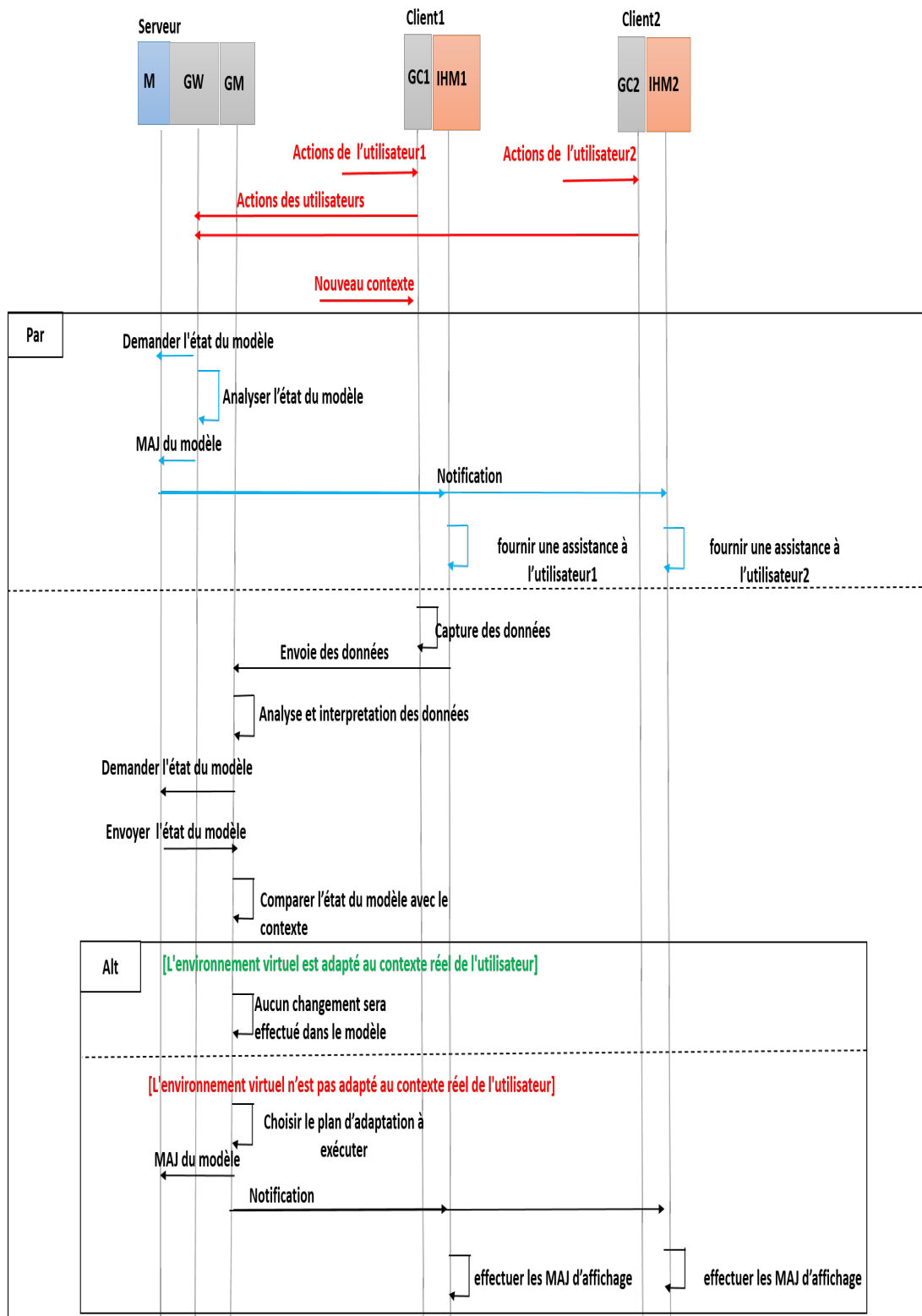


Figure 5.16: Diagramme d'activité

Le gestionnaire de contexte (GC) capte les actions des utilisateurs et les transmet au

gestionnaire de workflow qui, à son tour, demande l'état du modèle (EVC) pour mettre à jour les objets particuliers. À travers les fonctions du workflow, il fournit une assistance aux deux utilisateurs pour faciliter leur collaboration.

Pendant cette collaboration, un nouveau contexte survient chez l'utilisateur 1. Le processus de malléabilité est alors déclenché. Le gestionnaire de contexte du client 1 capture, analyse et classe les données contextuelles collectées. Ces informations sont ensuite transmises au gestionnaire de malléabilité (GM), qui est un composant centralisé du système situé sur le serveur. Le GM interprète ces données, demande l'état de l'EVC et de l'ER à partir du modèle. Après avoir vérifié si l'EVC est adapté au nouveau contexte, deux situations peuvent se présenter :

- Si l'EVC est compatible avec le nouveau contexte, le GM utilise les fonctions de malléabilité pour maintenir le modèle inchangé.
- Si l'EVC n'est pas adapté au contexte réel de l'utilisateur, le GM met à jour le modèle. Une notification est alors envoyée aux interfaces des deux clients pour effectuer les mises à jour nécessaires.

## 5.8 Étude de cas : jeu sérieux collaboratif malléable pour la rééducation

La rééducation est un processus complexe et fastidieux pour les patients, c'est pourquoi les jeux sérieux collaboratifs(JSCs) ont émergé comme une alternative intéressante pour rendre la rééducation plus ludique et motivante. En effet, des études ont démontré leur efficacité pour améliorer les capacités cognitives et motrices des patients en rééducation.

Cependant, pour accroître encore davantage les avantages des JSCs en les rendant adaptés et personnalisés, il est essentiel d'explorer de nouvelles possibilités en matière d'environnement virtuel collaboratif malléable. C'est précisément l'objet de notre étude de cas, où nous allons simuler différents scénarios pour illustrer le processus de malléabilité dans

un environnement virtuel collaboratif.

Nous allons nous appuyer sur nos travaux de référence [64], [63], qui ont validé la malléabilité mono-utilisateur dans un environnement virtuel en utilisant un jeu sérieux personnalisé basé sur la reconnaissance des émotions. Dans cette partie, le jeu est amélioré en ajoutant quatre gestes différents, afin de montrer clairement le basculement entre des gestes difficiles et faciles, et inversement.

Notre objectif principal est de valider notre contribution sur la conception d'EVCM, en nous basant sur un exemple concret de jeu sérieux. Nous allons projeter les concepts et notions de la partie précédente sur ce jeu, afin de montrer leur pertinence dans un contexte réel.

### 5.8.1 Description du jeu

Le Leap Motion [2] est un dispositif de détection de mouvements qui permet à l'utilisateur de contrôler un ordinateur à l'aide de gestes des mains et des doigts. Il peut être utilisé pour des applications de rééducation, notamment des jeux sérieux visant à améliorer la dextérité des mains et la coordination œil-main.

Voici comment le patient utilise le Leap Motion dans le jeu sérieux :

1. Lorsque le patient est prêt à jouer, il doit placer ses mains devant le Leap Motion, à une distance d'environ 20 cm du dispositif(voir Figure 5.17).
2. Le patient peut alors effectuer les gestes de la main nécessaires pour jouer. Par exemple, s'il doit attraper un objet, il peut faire un geste de pincement avec ses doigts. S'il doit déplacer un objet, il peut faire un geste de glissement avec sa main.
3. Le Leap Motion détectera les gestes du patient et les transmettra à l'ordinateur, qui les interprétera comme des actions dans le jeu.
4. Le patient peut continuer à jouer en effectuant les gestes de la main nécessaires pour avancer dans le jeu.



**Figure 5.17:** *Patient jouant à un jeu sérieux de rééducation avec le Leap Motion*

5. Lorsque le patient a terminé de jouer, il peut simplement retirer ses mains du champ de détection du Leap Motion.

Ce jeu de rééducation a été conçu pour aider les patients à améliorer leur dextérité et leur coordination grâce à l'utilisation du Leap Motion. Nous avons créé quatre gestes de la main différents pour permettre à l'utilisateur de progresser à travers des niveaux de difficulté croissants. Le but de cette progression graduelle est de montrer la malléabilité du jeu, en offrant un niveau facile au début et en augmentant la difficulté pour les patients en bonne condition, tandis que pour ceux qui ne sont pas à l'aise, la difficulté diminue. Les gestes incluent un geste de glissement pour déplacer les objets, un geste de pincement avec les cinq doigts pour attraper les objets, et un autre geste avec les deux doigts, l'index et la pouce. Des instructions détaillées pour chaque geste sont fournies dans le jeu pour permettre à l'utilisateur de comprendre comment jouer efficacement. En utilisant ces gestes, l'utilisateur peut non seulement améliorer sa dextérité et sa coordination, mais aussi profiter d'un jeu stimulant et adaptatif pour sa rééducation.

**Geste 1 : toucher sans force** Pour cet exercice, le patient doit effectuer des mouvements de balancier avec son bras de gauche à droite et de haut en bas, puis toucher la pomme avec la paume de sa main pour la faire tomber en utilisant la fonction "Contact

Begin" hérité de l'objet « Interaction Behaviour» du Leap Motion. Chaque fois que la pomme tombe, le score augmente d'un point. Cette tâche est considérée comme facile car elle ne nécessite pas beaucoup d'efforts physiques pour faire tomber la pomme. L'objectif de cet exercice est de mobiliser l'épaule en préparation à des mouvements plus complexes impliquant les doigts des mains, et ainsi améliorer la coordination entre l'épaule et la main.

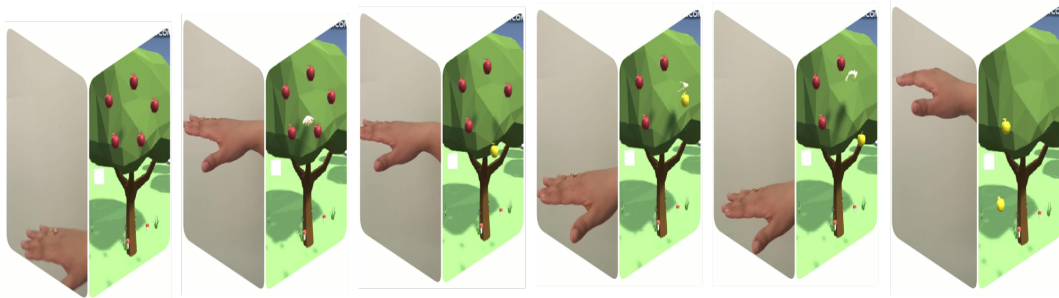


Figure 5.18: Illustration du Geste 1 :toucher sans force

**Geste 2 : toucher avec force** Le geste 2 consiste à effectuer un mouvement similaire au geste 1, mais avec une difficulté accrue. En effet, le patient doit exercer une force supplémentaire pour faire tomber la pomme, contrairement au geste précédent qui ne demandait qu'un simple toucher. Cette exigence est rendue possible grâce à la fonction "Contact End" héritée de l'objet "Interaction Behaviour" du Leap Motion. Le geste 2 peut être considéré comme une suite logique du premier exercice, visant à renforcer la coordination des mouvements entre la main et l'épaule.

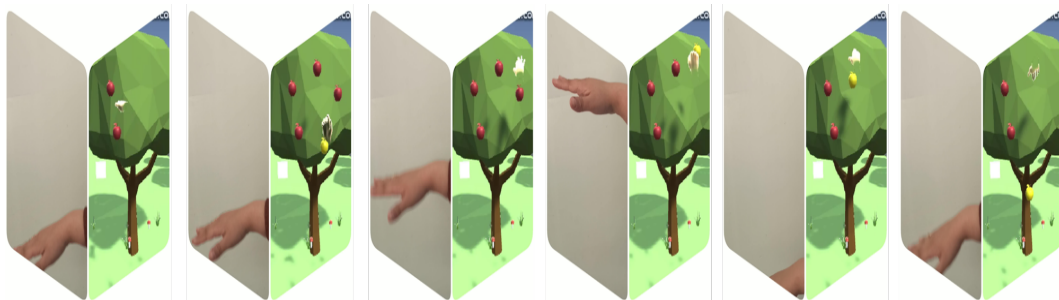
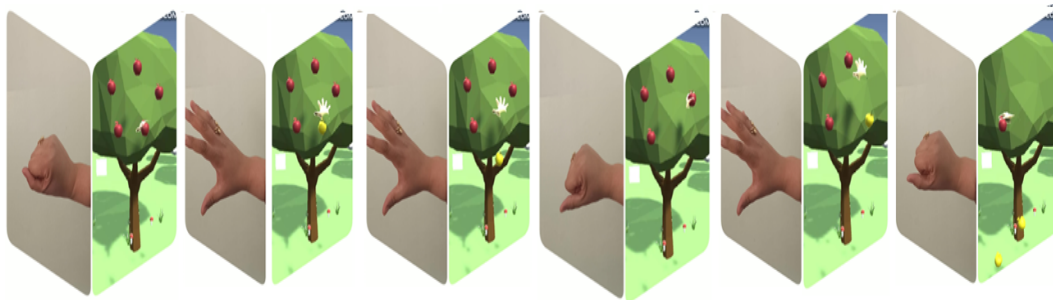


Figure 5.19: Illustration du Geste 2 :toucher avec force



**Geste 3 : saisie et relâcher** L'exercice consiste à demander au patient de saisir fermement une pomme dans sa main, puis de la relâcher. Pour évaluer la performance de ce geste, on mesure la distance entre les extrémités des doigts et la paume de la main. Si la distance entre les doigts et la paume est égale à 0, cela signifie que l'utilisateur a réussi l'exercice. La saisie et la libération de la pomme permettent également de renforcer les muscles de la main. L'objectif de cet exercice est de permettre au patient de s'entraîner à saisir et à lâcher des objets, ce qui est une compétence importante dans la vie quotidienne. En effet, cette compétence est utilisée lorsqu'il est nécessaire de transporter des objets d'un endroit à un autre, ou de les déplacer pour les ranger.



**Figure 5.20:** *Illustration du Geste 3 :saisie et relâcher*

**Geste 4 : zoom-in /zoom-out** Cet exercice vise à améliorer la coordination et la dextérité des doigts en utilisant une pince formée par l'index et le pouce pour agrandir ou réduire les pommes, avec une précision de 0,01. Lorsque la pomme atteint une taille moyenne et change de couleur en jaune, elle tombe de l'arbre et le patient gagne des points. Si la pomme ne parvient pas à atteindre la taille moyenne, elle reste sur l'arbre. Pour mesurer la performance de ce geste, on évalue la distance entre l'extrémité de l'index et de la pouce, et si cette distance est nulle, cela indique que l'exercice a été réalisé avec succès.

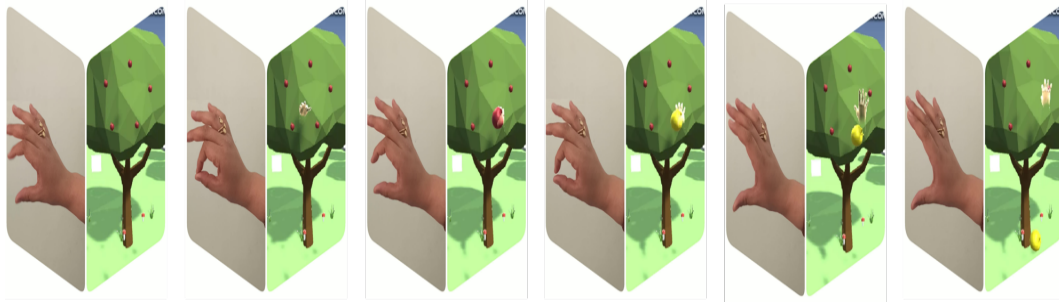


Figure 5.21: Illustration du Geste 4-Zoom-in

L'objectif de cet exercice est de mobiliser les muscles et les articulations des doigts pour améliorer la précision et la coordination, ce qui est particulièrement important pour les personnes ayant subi une blessure ou une maladie qui affecte les mouvements des doigts. Cet exercice peut contribuer à restaurer la fonction manuelle et la capacité à effectuer des tâches quotidiennes telles que la manipulation d'objets ou la prise d'un stylo.



Figure 5.22: Illustration du Geste 4-zoom-out

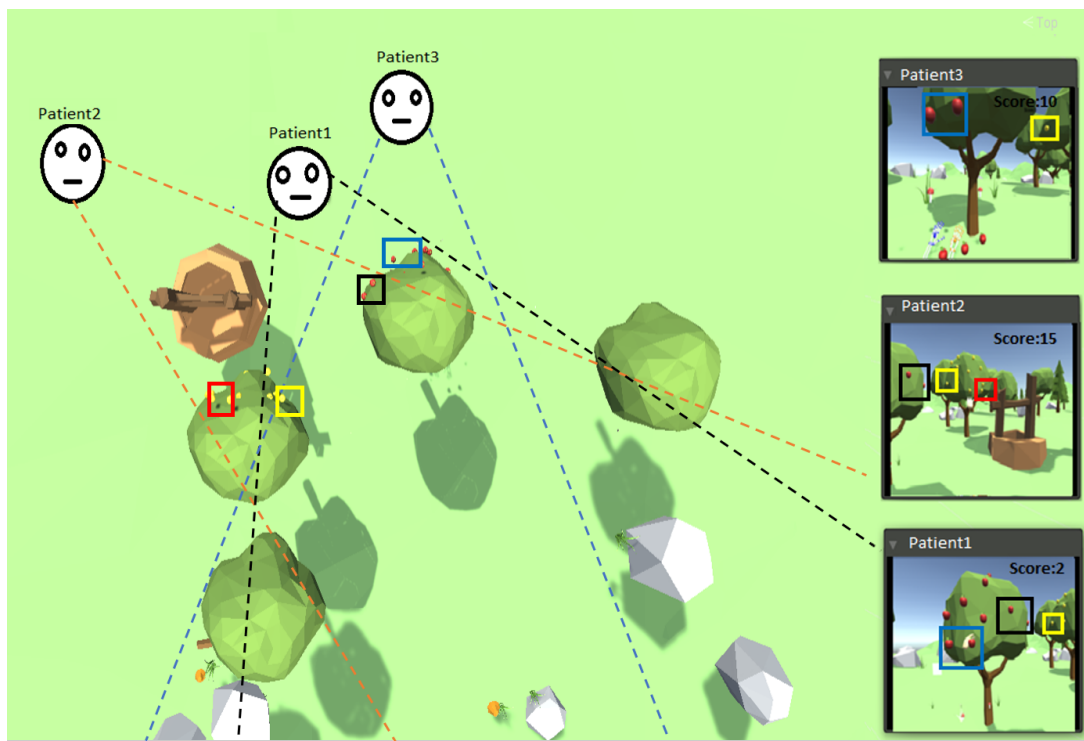
### 5.8.2 Scénario d'exécution

Dans le scénario de notre jeu sérieux virtuel collaboratif et malléable, trois patients comme montre dans la figure 5.23 participent à une compétition pour voir qui obtiendra le plus de points. Pour faciliter la collaboration entre les utilisateurs, un gestionnaire de workflow est mis en place, ainsi qu'un gestionnaire de malléabilité pour adapter le jeu au contexte de chaque utilisateur.

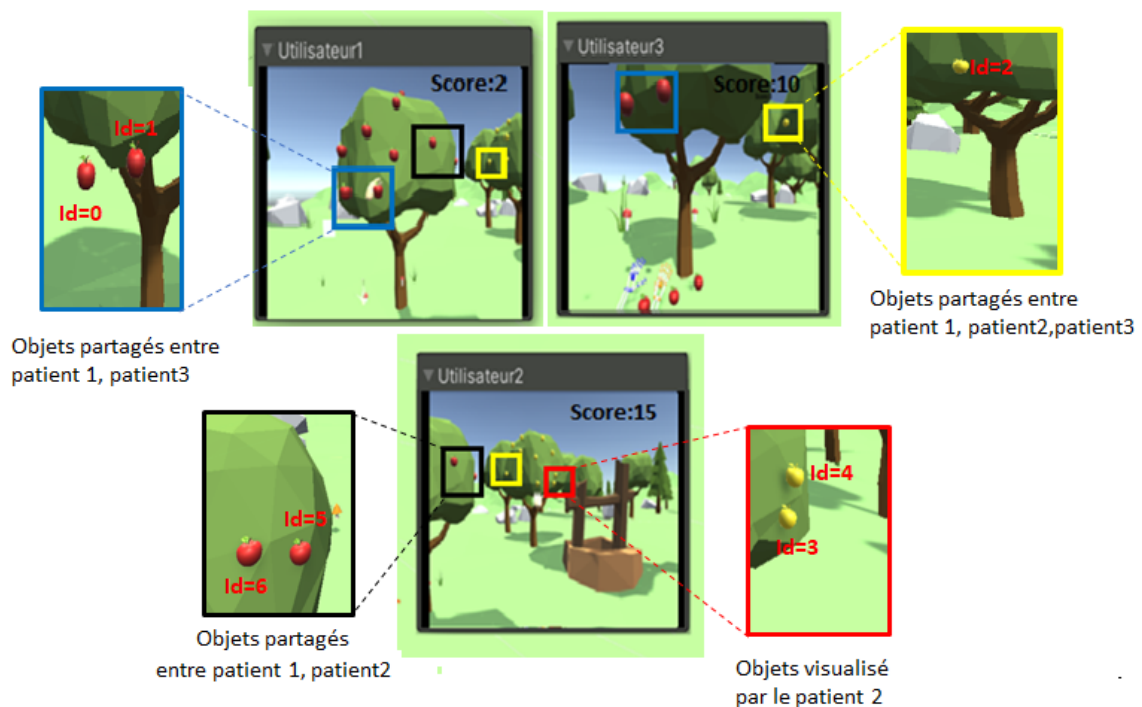
À tout moment, chaque patient dispose d'une interface adaptée à son contexte émotionnel, qui est classé en trois catégories : positif, négatif et neutre. Par exemple, si le

patient 1 est dans un état émotionnel positif, la difficulté du jeu sera augmentée, et inversement. Dans son espace personnel, le patient peut manipuler librement les pommes, qui sont également adaptées à son contexte.

Cependant, lorsqu'ils se trouvent dans l'espace partagé, le patient 1 peut interagir avec d'autres patients, comme le patient 2 dont le contexte émotionnel est négatif. Dans une telle situation, l'interaction avec les pommes doit être adaptée pour répondre aux contextes émotionnels des deux patients afin de créer un environnement virtuel collaboratif malléable. Ainsi, l'adaptation simultanée aux contextes émotionnels positifs et négatifs est essentielle. Dans ce qui suit, nous allons expliquer en détail comment s'adapter à ces deux contextes émotionnels différents en utilisant des concepts de malléabilité et de collaboration présentés précédemment. Le jeu sérieux peut ainsi offrir une expérience plus personnalisée et adaptée à chaque utilisateur, tout en favorisant la collaboration et la compétition entre les participants.



**Figure 5.23:** *Vue en plan de jeu avec trois patients jouant ensemble - Champs de vision de chaque patient(focus) à droite*



**Figure 5.24:** Zoom sur les pommes sélectionnées pour les tests à l'instant  $t$

La figure illustre les patients 1, 2 et 3 jouant ensemble, se déplaçant constamment dans l'espace virtuel. Cela peut conduire à des situations où ils partagent leur champ de vision tandis qu'un autre patient peut se retrouver seul dans une zone donnée. Nous avons sélectionné dix pommes différentes pour effectuer les tests, qui sont présentées dans la figure 5.24. Comme indiqué précédemment, la fonction nimbus est essentielle dans notre contribution pour adapter le comportement des objets dans un espace collaboratif. À chaque point de contrôle, où le processus de malléabilité doit être déclenché, soit temporairement soit conditionnellement. Nous devons donc savoir chaque pomme est visualisé par quel(s) patient(s) pour adapter son comportement en conséquence. Le tableau ci-dessous présente un exemple de pommes sélectionnées pour le test avec leur nimbus à l'instant  $t$ .

La pomme	Nimbus
0	P1,P3
1	P1,P3
2	P1,P2,P3
3	P2
4	P2
5	P1,P2
6	P1,P2

**Tableau 5.4:** Nimbus des pommes 0-6

### 5.8.3 Le processus de malléabilité

Pour un entraînement collaboratif de 20 minutes, le processus de malléabilité est exécuté toutes les deux minutes. Chaque seconde est composée de 24 images, ce qui permet d'obtenir 24 reconnaissances d'émotions par seconde, soit 120 x 24 reconnaissances lors de chaque exécution du processus de malléabilité. Au cours de ces 20 minutes, le système vérifie 10 points de contrôle pour déterminer si une adaptation de l'environnement virtuel collaboratif est nécessaire ou non. À chaque point de contrôle, nous calculons la moyenne des émotions positives, négatives et neutres pour chaque patient au cours des deux minutes précédentes. Le processus de malléabilité basé sur la reconnaissance d'émotions suit trois étapes avant de décider quelles actions doivent être exécutées pour rendre l'EVC malléable.

**Première étape : calculer les pourcentages des sept émotions** La première étape du processus de malléabilité consiste à calculer les pourcentages des sept émotions pour chaque image, en utilisant l'algorithme ci-dessous :

---

**Algorithme 5.4 :** CalculerEmotions

---

**Input :**  $EF[NbF]$  ; // tableau qui contient l'ensemble des photos des expressions faciales.  $NbF$  est le nombre de image.

**Output :**  $Emotions[NbF][7]$  ; // matrice qui contient pour chaque image les pourcentage des sept emotions.

```
1  $i \leftarrow 1$ ; while  $i \leq NbF$  do
2    $Emotions[i] \leftarrow DeepFace(EF[i])$ ; // Framework DeepFace retourne un
   tableau de taille 7.  $Emotions[NbF][1]=joie$ ,
    $Emotions[NbF][2]=colère$ ,  $Emotions[NbF][3]=peur$ ,
    $Emotions[NbF][4]=dégoût$ ,  $Emotions[NbF][5]=tristesse$ ,
    $Emotions[NbF][6]=surprise$ ,  $Emotions[NbF][7]=neutralité$ 
3    $i \leftarrow i + 1$ ;
4 end
5 Retourner  $Emotions$ ;
```

---

L'algorithme "CalculerEmotions" est utilisé pour calculer les pourcentages des sept émotions (joie, colère, peur, dégoût, tristesse, surprise et neutralité) pour chaque image d'une séquence de photos d'expressions faciales. Il prend en entrée un tableau  $EF$  qui contient l'ensemble des photos des expressions faciales, avec  $NbF$  étant le nombre de images dans la séquence. L'algorithme utilise la bibliothèque DeepFace pour calculer les pourcentages des sept émotions pour chaque image. Pour chaque itération de la boucle, l'algorithme prend une photo d'expression faciale du tableau  $EF$  et l'envoie à la fonction DeepFace.

---

**Algorithme 5.5 :** CalculateEmotions

---

```
Input : EF[NbF] ;           // Array containing all the facial expression
                             photos. NbF is the number of frames.

Output : Emotions[NbF][7] ; // Matrix containing the percentages of
                             the seven emotions for each frame.

1  $i \leftarrow 1$ ; while  $i \leq NbF$  do
2    $Emotions[i] \leftarrow DeepFace(EF[i]);$  // DeepFace framework returns an
     array of size 7. Emotions[NbF][1]=joy, Emotions[NbF][2]=anger,
     Emotions[NbF][3]=fear, Emotions[NbF][4]=disgust,
     Emotions[NbF][5]=sadness, Emotions[NbF][6]=surprise,
     Emotions[NbF][7]=neutral
3    $i \leftarrow i + 1$ ;
4 end
5 Return  $Emotions$ ;
```

---

La fonction retourne un tableau de taille 7 qui représente les pourcentages des sept émotions pour cette photo. L'algorithme stocke ensuite ces pourcentages dans la matrice *Emotions*, qui a la même taille que le tableau EF1. En fin de compte, l'algorithme retourne la matrice *Emotions* qui contient les pourcentages des sept émotions pour chaque image. Dans notre cas, cela signifie que pour chaque point de contrôle, qui se produit toutes les 2 minutes pendant l'entraînement collaboratif de 20 minutes, le système calcule la moyenne des émotions positives, négatives et neutres pour chaque patient au cours des 2 dernières minutes, en utilisant une matrice *Emotions* de taille 120 x 24.

**Deuxième étape : Calculer le pourcentage des classes d'émotions** La deuxième étape consiste à calculer le pourcentage des émotions positives, négatives et neutres pour chaque image, comme présenté dans l'algorithme 14 ci-dessous : Cet algorithme, inti-

tulé "CalculerClassesEmotions", permet de calculer les pourcentages des émotions positives, négatives et neutres pour chaque image. Il prend en entrée une matrice *Emotions* de taille  $NbF \times 7$ , où chaque ligne correspond à une image et chaque colonne correspond à une émotion spécifique (colonne 1 : joie, colonne 2 : tristesse, colonne 3 : colère, colonne 4 : peur, colonne 5 : dégoût, colonne 6 : surprise et colonne 7 : neutre).

---

**Algorithme 5.6 :** CalculerClassesEmotions

---

**Input :** *Emotions*[ $NbF$ ][7]  
**Output :** *ClasseEmo*[3]

- 1  $i \leftarrow 0, MPos \leftarrow 0, MNeg \leftarrow 0, MNeu \leftarrow 0, S1 \leftarrow 0, S2 \leftarrow 0;$
- 2 **for**  $i \leftarrow 1$  à  $NbF$  **do**
- 3      $EPos[i] \leftarrow Emotions[i][1] + Emotions[i][6]/2;$
- 4      $ENeg[i] \leftarrow Emotions[i][2] + Emotions[i][3] + Emotions[i][4] +$   
         $Emotions[i][5] + Emotions[i][6]/2;$
- 5      $ENeu[i] \leftarrow Emotions[i][7];$
- 6 **end**
- 7 **for**  $i \leftarrow 1$  à  $NbF$  **do**
- 8      $S1 \leftarrow S1 + EPos[i];$
- 9      $S2 \leftarrow S2 + ENeg[i];$
- 10     $S3 \leftarrow S3 + ENeu[i];$
- 11 **end**
- 12  $MPos \leftarrow S1/NbF; MNeg \leftarrow S2/NbF; MNeu \leftarrow S3/NbF;$
- 13  $ClasseEmo[1] \leftarrow MPos; ClasseEmo[2] \leftarrow MNeg; ClasseEmo[3] \leftarrow MNeu;$
- 14 **return** *ClasseEmo*;

---

Le résultat est une matrice *ClasseEmo* de taille  $1 \times 3$ , où chaque élément représente le pourcentage d'une émotion spécifique (élément 1 : pourcentage d'émotion positive, élément 2 : pourcentage d'émotion négative, élément 3 : pourcentage d'émotion neutre). L'algorithme fonctionne en utilisant deux boucles "For". La première boucle itère sur chaque image et calcule le pourcentage d'émotion positive, négative et neutre pour chaque



image en utilisant les formules appropriées. La deuxième boucle calcule la somme des pourcentages d'émotion positive, négative et neutre pour toutes les images. Enfin, les pourcentages moyens sont calculés en divisant chaque somme par le nombre d'images et stockés dans les éléments correspondants de la matrice *ClasseEmo*.

**Troisième étape : Calculer l'émotion dominante** Lors d'un point de contrôle visant à adapter le comportement d'un objet en fonction de l'ensemble des utilisateurs qui le visualisent, il est nécessaire de calculer son "nimbus". Par exemple, si on souhaite adapter la pomme d'ID=1 aux patients P1 et P3, il est important que le geste utilisé pour faire tomber la pomme soit adapté à ces patients. Tout d'abord, il est nécessaire de déterminer l'émotion dominante des patients P1 et P3. Pour cela, on exécute l'algorithme 5.7, qui prend en entrée les expressions faciales des deux patients ( $EF1[NbF]$  et  $EF3[NbF]$ ) et calcule la classe d'émotion dominante moyenne. Cet algorithme utilise deux autres algorithmes : "CalculerEmotions" et "CalculerClassesEmotion". Après avoir obtenu les classes d'émotions pour P1 et P3, l'algorithme calcule la moyenne de ces classes pour chaque émotion et stocke ces moyennes dans la variable *ClassesEmo*. Enfin, il détermine quelle émotion est dominante en utilisant la fonction "Max" et stocke cette émotion dans la variable *EmotionDominante*. En résumé, cet algorithme calcule la classe d'émotion dominante moyenne pour les deux patients en combinant leurs expressions

---

**Algorithme 5.7 :** CalculerEmotionDominanteP1P3

---

**Input :**  $EF1[NbF]$ ,  $EF3[NbF]$

**Output :**  $EmotionDominante$

```
1  $Emotion1[NbF][7]$ ,  $Emotion3[NbF][7]$ ,  $ClassesEmo1[3]$ ,  $ClassesEmo[3]$  ;
2  $Emotion1 \leftarrow CalculerEmotions(EF1[NbF])$  ;
3  $Emotion3 \leftarrow CalculerEmotions(EF3[NbF])$  ;
4  $ClassesEmo1 \leftarrow CalculerClassesEmotion(Emotion1)$  ;
5  $ClassesEmo3 \leftarrow CalculerClassesEmotion(Emotion3)$  ;
6  $ClassesEmo[1] \leftarrow (ClassesEmo1[1] + ClassesEmo3[1])/2$  ;
7  $ClassesEmo[2] \leftarrow (ClassesEmo1[2] + ClassesEmo3[2])/2$  ;
8  $ClassesEmo[3] \leftarrow (ClassesEmo1[3] + ClassesEmo3[3])/2$  ; // Calculer la
   moyenne des trois classes d'émotions pour les deux patients P1 et
   P3
9  $EmotionDominante \leftarrow Max(ClassesEmo[1], ClassesEmo[2], ClassesEmo[3])$  ;
10 return  $EmotionDominante$  ;
```

---

Lorsque l'on passe d'un geste à un autre, il peut survenir un problème si la pomme est tombée juste avant la collaboration avec le patient1 en utilisant le geste 1, qui consiste simplement à toucher la pomme, alors que pour le patient 2, la pomme tombe avec le geste 3, qui nécessite de saisir puis de relâcher la pomme pour la faire tomber. Afin de résoudre ce problème, nous avons proposé un algorithme (algorithme 19) pour sélectionner le geste approprié. Toutefois, lors des essais avec de vrais patients, d'autres propositions peuvent

être envisagées.

---

**Algorithme 5.8 : ChoixGeste**

---

```
Input : Gestes[N] ;      // Geste[N] représente les gestes des patients
        avant la malléabilité

Output : GesteMalleable;

1 GesMin ← calculerMin(Gestes[N]);
2 if (EmoDominante == positif || EmoDominante == neutre) then
3   | if GesMin == 4 then
4   |   | retourner GesMin
5   | end
6   | sinon
7   |   | retourner GesMin +1
8   | fin
9   |
10 end
11 if (EmoDominante == negatif) then
12 | if GesMin == 1 then
13 |   | retourner GesMin
14 | end
15 | sinon
16 |   | retourner GesMin -1
17 | fin
18 end
19 Fin
```

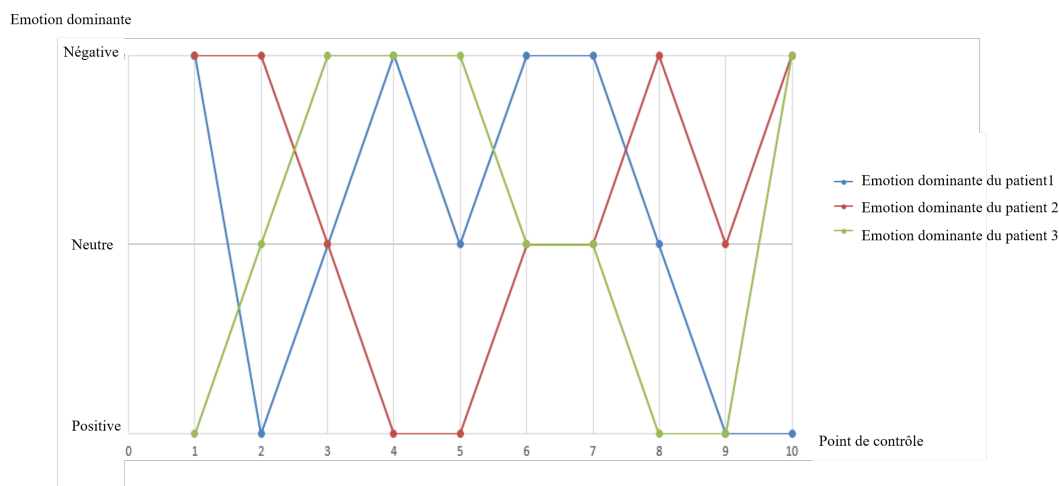
---

Ainsi, dans le cas précédent, nous avons déterminé que  $GestMin = 1$ . Si l'émotion dominante entre les patients 1 et 2 est positive, alors le geste choisi pour attraper la pomme est le geste 2. En revanche, si l'émotion dominante est négative, le geste choisi est le geste 1, dans le but de mettre le patient le plus à l'aise possible. Toutefois, comme nous l'avons mentionné précédemment, les experts peuvent avoir des opinions divergentes concernant l'algorithme de choix. Dans cette étude, nous cherchons simplement à simuler

la réalité.

### 5.8.4 Simulation

À l'aide de Python, nous générons pour chaque patient 10 séries de 120 x 24 x 7 valeurs. Chaque valeur représente le pourcentage d'une des sept émotions (joie, colère, peur, dégoût, tristesse, surprise et neutralité), variant entre 0 et 100 avec des valeurs aléatoires. La somme des pourcentages des émotions est toujours égale à 100. Le graphique ci-dessous représente l'évolution des émotions des patients (positives, négatives et neutres) pendant une session de rééducation collaborative de 20 minutes. Des points de contrôle sont effectués toutes les 2 minutes, ce qui donne 10 points de contrôle au total.



**Figure 5.25:** L'évolution des émotions pendant la séance d'entraînement pour les patients 1, 2 et 3.

Le tableau ci-dessous illustre le processus de malléabilité appliqué aux trois pommes ayant pour identifiants 0, 2 et 4 (voir figure 5.24), à trois points de contrôle différents, soit 3, 6 et 9. Si l'on souhaite retranscrire ce tableau en utilisant les concepts de la conception, le point de contrôle devient le déclencheur, le "nimbus" et le "Geste initial" représentent l'état de l'EVC, l'émotion moyenne représente l'état de contexte de l'utilisateur (soit l'état de l'ER) et l'action correspond à l'ensemble des actions à exécuter pour rendre le système malléable. L'algorithme de choix de plan d'adaptation est l'algorithme 19, qui permet

Pomme	Processus de malléabilité								
0	Point de contrôle	3			6		9		
	Nimbus	P1,P3			P1,P2		P1		
	Emotion moyenne	négative			positive		positive		
	Geste initial	P1	P3		P1	P2		geste4	
		geste1	geste2		geste1	geste1			
Actions malléabilité	Faire geste1			Faire geste 2		Faire geste 4			
2	Point de contrôle	3			6		9		
	Nimbus	P1,P2,P3			P2		P2,P3		
	Emotion moyenne	négative			positive		neutre		
	Geste initial	P1	P2	P3	geste1		P2	P3	
		geste2	geste1	geste3			geste2	geste3	
Actions malléabilité	Faire geste1			Faire geste 2		Faire geste3			
4	Point de contrôle	3			6		9		
	Nimbus	P1,P2			P3		P1,P2,P3		
	Emotion moyenne	positive			neutre		positive		
	Geste initial	P1	P2		geste3		P1	P2	P3
		geste1	geste4				geste1	geste2	geste4
Actions malléabilité	Faire geste 2			Faire geste4		faire geste3			

**Tableau 5.5:** *Processus de malléabilité des pommes 0,2,4 dans les points de contrôle 3,6,9*

de sélectionner le geste à manipuler par les patients en collaboration selon leur contexte. Toutefois, cet algorithme de choix peut être plus complexe ou basé sur l'IA en fonction du contexte d'utilisation.

En examinant le tableau, on constate que la pomme d'ID=0 au point de contrôle 9 est visualisée par le patient 1. À ce moment-là, le comportement de cette pomme doit être adapté au patient 1 dont l'émotion dominante est positive. Nous allons donc augmenter la difficulté du jeu. Cependant, nous remarquons qu'il est en train d'effectuer le dernier geste. Nous lui demandons alors de faire le même geste 4.

Pour la pomme d'ID=2 au point de contrôle 3, elle est visualisée par trois patients : p1, p2 et p3. Le comportement de cette pomme doit être adapté au contexte des trois patients et donc à l'émotion dominante entre les trois patients, qui est calculée comme indiqué dans l'algorithme 5.7. Nous remarquons que p1 a manipulé la pomme avec le geste 2 dans les deux minutes précédentes dans son environnement, p2 avec le geste 1 et p3 avec le geste 3. Nous prenons le minimum, qui est le geste 1, et comme l'émotion dominante

est négative, nous devons donc diminuer la difficulté. Cependant, le geste 1 est le geste le plus facile, nous gardons donc toujours le même geste 1. Le choix est effectué en utilisant l'algorithme 19.

## 5.9 Conclusion

En conclusion, notre étude approfondie de la conception d'un environnement virtuel collaboratif malléable a révélé un manque de modèles et de formalismes permettant de superviser les ajustements et les modifications des EVC. Pour remédier à cette lacune, nous avons proposé une définition précise de ce concept. Ensuite, en nous basant sur le modèle d'un EVC présenté par [99], nous avons intégré d'autres composants tels que le gestionnaire de la malléabilité et le gestionnaire de contexte afin de répondre aux contraintes liées à la malléabilité et à la sensibilité au contexte. Pour établir la relation entre ces composants, nous avons conçu un modèle conceptuel MVC-3DCM qui sépare le traitement des données, facilitant ainsi l'intégration de nouveaux modules en fonction des besoins spécifiques du modèle utilisé. Par la suite, nous avons présenté une simulation d'un jeu sérieux collaboratif malléable pour la rééducation des membres supérieurs basé sur la reconnaissance d'émotions comme données contextuelles, démontrant ainsi l'application des concepts présentés tout au long de ce chapitre. Grâce à notre étude, nous espérons fournir un point de départ solide pour la conception de futurs EVCM répondant aux besoins des utilisateurs et offrant une expérience immersive personnalisée.

# Conclusion Générale

Dans le cadre de cette thèse, notre objectif est de concevoir et mettre en œuvre une architecture pour les collecticiels dans les environnements virtuels/augmentés collaboratifs (EVC/EAC). Nous introduisons un nouveau concept, celui d'"Environnement Virtuel Collaboratif Malléable (EVCM/EACM)", que nous définissons en détail au cours de notre thèse. Notre choix de nous appuyer sur les EVC est motivé par le fait que les EAC combinent à la fois des aspects virtuels et réels, ce qui rend notre contribution pertinente pour la partie virtuelle des EAC. Notre démarche s'appuie sur trois concepts clés : l'environnement virtuel (EV), la malléabilité et le collecticiel, qui servent de fondements à notre étude.

Nous avons entamé notre travail en fournissant une vue d'ensemble des EVC, en présentant la réalité virtuelle et ses concepts fondamentaux. Ensuite, nous avons examiné les architectures réseau spécifiques à ces environnements, ainsi que les modes de distribution des données utilisés dans les EVC. Nous avons également exploré les différents types d'interactions 3D collaboratives. Enfin, pour illustrer notre propos, nous avons présenté trois exemples récents d'EVC.

Ensuite, nous nous sommes intéressés au deuxième concept, celui de la "malléabilité". Cependant, nous n'avons pas trouvé de travaux clairs portant sur l'utilisation ou la définition de ce terme dans le contexte des EVC. Nous avons donc entrepris une recherche sur la définition de la malléabilité dans les systèmes classiques, et nous avons constaté que seuls les collecticiels ont exploré ce concept. Nous avons alors étudié des concepts proches tels que la plasticité 3D, l'adaptation et la sensibilité au contexte. Cette recherche nous a per-

mis de définir des exigences pour un EVCM, en ouvrant les aspects de la réalité virtuelle, en définissant des déclencheurs, des sources d'adaptation, des approches d'adaptation, des variables d'adaptation et un contrôleur, tout en mettant en évidence les aspects de la collaboration.

Par la suite, nous avons souligné qu'il est nécessaire de décomposer ces exigences en différentes dimensions clés : la réalité virtuelle, la malléabilité et la collaboration. L'analyse des concepts existants nous a conduit à conclure que la sensibilité au contexte est le concept qui répond le mieux à nos exigences. Nous avons approfondi notre recherche dans ce concept en présentant son architecture générale, qui servira de base à notre gestionnaire de malléabilité. De plus, nous avons réalisé une étude détaillée sur le concept de contexte dans les environnements virtuels.

Après avoir réalisé une recherche approfondie sur les collecticiels et leurs architectures, nous avons constaté que les modèles d'architecture existants pour les collecticiels sont limités dans leur application aux Environnements Virtuels Collaboratifs (EVC) malléables. Cette limitation restreint leur capacité à s'adapter de manière rapide et efficace aux besoins changeants. Par conséquent, nous avons entrepris de concevoir et de réaliser des architectures de collecticiels flexibles, adaptatives et sensibles au contexte des utilisateurs, dans le but de créer des EVC malléables efficaces.

Les résultats de notre thèse ont conduit à deux contributions significatives. La première contribution consiste en une architecture logicielle pour une plateforme virtuelle adaptable utilisée dans le domaine de la rééducation fonctionnelle. Cette architecture comprend trois modules : un module de capture et d'interprétation des données, un module d'analyse des données et un module d'adaptation des exercices en fonction de l'état émotionnel du patient. Des tests utilisateurs ont confirmé que cette plateforme virtuelle peut s'ajuster en fonction de l'état émotionnel du patient. La deuxième contribution se manifeste par la proposition d'un nouveau modèle architectural appelé MVC-3DCM, spécifiquement conçu pour répondre à la flexibilité dans les environnements collaboratifs virtuels impliquant des interactions 3D entre plusieurs utilisateurs simultanément. Ce modèle étend



le modèle Clock, qui a été initialement conçu pour les systèmes collectifs, en intégrant trois composants : un gestionnaire de flux de travail, un gestionnaire de malléabilité et un gestionnaire de contexte. Une étude de cas sur un jeu sérieux collaboratif a été présentée pour illustrer l'application pratique du modèle proposé, passant d'une représentation de bas niveau (abstraite) à une représentation de haut niveau (application).

## PERSPECTIVES

Plusieurs perspectives de recherche peuvent être envisagées à la suite de cette thèse. Dans un premier temps, il serait intéressant d'appliquer l'architecture développée dans des cliniques de rééducation afin d'évaluer et de démontrer son efficacité thérapeutique. En parallèle, une approche prometteuse consisterait à exploiter les outils d'intelligence artificielle, tels que ChatGPT, pour intégrer des capacités de prise de décision dans le système. Cela permettrait d'améliorer la malléabilité en se basant sur des informations contextuelles. Il serait également possible de proposer d'autres systèmes de réalité virtuelle malléables en se basant sur des informations contextuelles différentes de celles utilisées actuellement, comme les mouvements corporels. Cette diversification des sources d'informations permettrait d'enrichir la malléabilité des environnements virtuels collaboratifs. En outre, il serait envisageable de tester la contribution de cette thèse dans d'autres domaines tels que l'éducation, la médecine, le sport, etc. Cette exploration permettrait d'évaluer l'efficacité de l'approche de malléabilité développée dans des contextes variés et de déterminer son potentiel d'application. Enfin, une perspective intéressante serait d'approfondir le concept même de la malléabilité en le définissant comme une grandeur mesurable. Il serait possible d'élaborer des méthodes spécifiques pour évaluer et quantifier dans quelle mesure l'EVCN répond aux besoins des utilisateurs. Cela contribuerait à une meilleure compréhension de la malléabilité et à l'établissement de critères objectifs pour son évaluation.

En conclusion, ces différentes perspectives de recherche ouvrent de nouvelles voies pas-

sionnantes pour approfondir les concepts de malléabilité dans les environnements virtuels collaboratifs et pour explorer leur application dans des domaines variés tels que la santé, l'éducation et la rééducation.

# Bibliographie

- [1] Eye tracking. <https://www.tobiipro.com>. Consulté : 15/08/2022.
- [2] Leap motion controller. <https://www.ultraLeap.com/product/leap-motion-controller/>. Consulté : 12/2021.
- [3] Mood me. <https://www.mood-me.com/products/unity-face-ai-ar-sdk/>. Consulté : 2021-11-10.
- [4] Unity-technologies- barracuda. <https://github.com/Unity-Technologies/barracuda-release>. Consulté : 2021-12-19.
- [5] Weather map. <https://openweathermap.org/current>. Consulté : 14/08/2022.
- [6] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles. Towards a better understanding of context and context-awareness. In *International symposium on handheld and ubiquitous computing*, pages 304–307. Springer, 1999.
- [7] L. Aguerreche, T. Duval, and B. Arnaldi. Analyse de techniques de cooperation en environnements virtuels 3d. In *Proceedings of the 6th International Conference on Virtual Reality (ICVR)*, 2009.
- [8] L. Aguerreche, T. Duval, and A. Lécuyer. Short paper : 3-hand manipulation of virtual objects. In *Joint Virtual Reality Conference of EGVE - ICAT - EuroVR*, pages 153–156, 2009.

- [9] S. Alda, J. Kuck, and A. Cremers. Tailorability of personalized bpm-based workflow compositions. In *The 2007 IEEE Congress on Services*, pages 245–252, 2007.
- [10] A. Alhagbani and A. Williams. Home-Based Exergames for Older Adults Balance and Falls Risk : A Systematic Review. *Phys. Occup. Ther. Geriatr*, pages 1–17.
- [11] K. Arthur, T. Preston, R. Taylor, F. Brooks, M. Whitton, and W. Wright. Designing and building the pit : a head-tracked stereo workspace for two users. In *Proc. of 2nd International Immersive Projection Technology Workshop*, 1998.
- [12] O. Barzilay and A. Wolf. Adaptive rehabilitation games. *Journal of Electromyography and Kinesiology*, 23(1) :182–189, 2013.
- [13] L. Bass, R. Faneuf, R. Little, N. Mayer, B. Pellegrino, S. Reed, R. Seacord, S. Shepard, and M. Szczur. A metamodel for the runtime architecture of an interactive system. *SIGCHI Bulletin*, 24(1) :32–37, 1992.
- [14] E. Bekele, M. Young, Z. Zheng, L. Zhang, A. Swanson, R. Johnston, J. Davidson, Z. Warren, and N. Sarkar. A step towards adaptive multimodal virtual social interaction platform for children with autism. In *international conference on universal access in human-computer interaction*, pages 464–473. Springer, 2013.
- [15] H. Ben Abdesslem and C. Frasson. Real-time brain assessment for adaptive virtual reality game : A neurofeedback approach. In *international conference on brain function assessment in learning*, pages 133–143. Springer, 2017.
- [16] S. Benbelkacem, D. Aouam, N. Zenati-Henda, A. Bellarbi, A. Bouhena, and S. Otmane. Mvc-3d : Adaptive design pattern for virtual and augmented reality systems. *arXiv preprint arXiv :1903.00185*, 2019.
- [17] S. Benbelkacem, N. Zenati-Henda, H. Belghit, S. Otmane, and A. Bellarbi. Human-human collaboration formalism for groupware tailorability in collaborative augmented environments. In *GRAPP*, pages 498–503, 2015.

- [18] D. Bian, J. Wade, Z. Warren, and N. Sarkar. Online engagement detection and task adaptation in a virtual reality based driving simulator for autism intervention. In *international conference on universal access in human-computer interaction*, pages 538–547. Springer, 2016.
- [19] R. A. Blankendaal and T. Bosse. Using run-time biofeedback during virtual agent-based aggression de-escalation training. In *international conference on practical applications of agents and multi-agent systems*, pages 97–109. Springer, 2018.
- [20] G. Bourguin. *Un support informatique à l'activité coopérative fondé sur la Théorie de l'Activité : le projet DARE*. PhD thesis, Université des Sciences et Technologies de Lille, 2000.
- [21] G. Bourguin. Proposition pour une gestion dynamique de l'inter-activités dans le tcao. In *Proceedings of the 16th Conference on l'Interaction Homme-Machine*, pages 191–194, 2004.
- [22] D. A. Bowman, D. B. Johnson, and L. F. Hodges. Testbed evaluation of virtual environment interaction techniques. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 26–33, 1999.
- [23] J. Burkhardt. Réalité virtuelle et ergonomie : quelques apports réciproques. *Le travail humain - P.U.F*, 96(1), 2003.
- [24] J.-M. Burkhardt, B. Bardy, and D. Lourdeaux. Immersion, realism and presence in the design and evaluation of virtual environments. *Psychologie Française*, 48(2) :35–42, 2003.
- [25] D. M. Calandra, F. G. Praticcò, M. Migliorini, V. Verda, and F. Lamberti. A multi-role, multi-user, multi-technology virtual reality-based road tunnel fire simulator for training purposes. In *VISIGRAPP*, 2021.

- [26] C. Carlsson and O. Hagsand. Dive—a platform for multi-user virtual environments. *Computers & graphics*, 17(6) :663–669, 1993.
- [27] T. Chaari. Adaptation d’applications pervasives dans des environnements multi-contextes. *PhDthesis a l’institut national des sciences appliquees de lyon, laboratoire LIRIS*, 2007.
- [28] N. Cheaib. *Contribution à la malléabilité des collecticiels : une approche basée sur les services web et les agents logiciels*. PhD thesis, Université d’Evry-Val d’Essonne, 2010.
- [29] G. Chen and D. Kotz. A survey of context-aware mobile computing research. 2000.
- [30] E. F. Churchill and D. Snowdon. Collaborative virtual environments : an introductory review of issues and systems. *virtual reality*, 3(1) :3–15, 1998.
- [31] L. Cugusi, L. Prosperini, and G. Mura. Exergaming for Quality of Life in Persons Living with Chronic Diseases : A Systematic Review and Meta-analysis. *PM&R*, 13(7) :756–780, 2021.
- [32] W. Dangelmaier, H. Hamoudia, and R. Klahold. Domain preferences for end-user tailoring in shared workflow interfaces. In *CRIWG Workshop*, Darmstadt, Germany, 2002.
- [33] E. de Matos, L. A. Amaral, R. Tiburski, W. Lunardi, F. Hessel, and S. Marczak. Context-aware system for information services provision in the internet of things. In *2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*, pages 1–4. IEEE, 2015.
- [34] D. Delaney, T. Ward, and S. McLoone. On consistency and network latency in distributed interactive applications : A survey – part i. *Presence : Teleoperators and Virtual Environments*, 15(2) :218–234, 2006.

- [35] P. Dewan. Architectures for collaborative applications. *Trends in Software*, special issue on Collaborative Systems :169–193, 1999.
- [36] A. K. Dey. Understanding and using context. *Personal and ubiquitous computing*, 5(1) :4–7, 2001.
- [37] O. O. Doma. *EEG as an Input for Virtual Reality*, pages 1–4. Springer International Publishing, Cham, 2019.
- [38] C. Domingues. *Interaction 3D collaborative en réalité virtuelle*. PhD thesis, Université d'Evry-Val d'Essonne, 2010.
- [39] T. Duval and C. Fleury. An asymmetric 2d pointer/3d ray for 3d interaction within collaborative virtual environments. In *3D technologies for the World Wide Web. Proceedings of the 14th International Conference on 3D Web Technology*, pages 33–41, 2009.
- [40] T. Duval, A. Lecuyer, and S. Thomas. Skewer : a 3d interaction technique for 2-user collaborative manipulation of objects in virtual environments. In *Proceedings of the 3D User Interfaces (3DUI'06)*, pages 69–72, 2006.
- [41] A. Dzedzickis, A. Kaklauskas, and V. Bucinskas. Human emotion recognition : Review of sensors and methods. *Sensors*, 20(3) :592, 2020.
- [42] P. Ekman and W. V. Friesen. *Facial action coding system*. 1978.
- [43] A. Ellis. Conceptual model of groupware. In *Proc CSCW*, pages 79–88. ACM Press NY, 1994.
- [44] C. Ellis. Groupware : Overview and perspectives. In *Verteilte Künstliche Intelligenz und kooperatives Arbeiten : 4. Internationaler GI-Kongreß Wissensbasierte Systeme München, 23.–24. Oktober 1991 Proceedings*, pages 18–29. Springer, 1991.

- [45] P. M. Emmelkamp and K. Meyerbröker. Virtual reality therapy in mental health. *Annual Review of Clinical Psychology*, 17 :495–519, 2021.
- [46] C. Fleury. *Modèles de conception pour la collaboration Distante en environnements virtuels distribués : de l'architecture aux métaphores*. PhD thesis, INSA Rennes, 2012.
- [47] P. Fuchs. *Le traité de la réalité virtuelle*, volume 2. Presses des MINES, 2006.
- [48] S. Gerbaud, V. Gouranton, and B. Arnaldi. Adaptation in collaborative virtual environments for training. In *International conference on technologies for E-learning and digital entertainment*, pages 316–327. Springer, 2009.
- [49] S. Germain. Use of collaborative software to improve nuclear power plant outage management. In *Proceedings of the 9th International Conference on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies*, Charlotte, NC, USA, February 2015.
- [50] Y. Ghayam. *La Sensibilité au contexte dans un environnement mobile*. PhD thesis, PhD thesis, Université Mohammed V Souissi-RABAT, Ecole Nationale Supérieure . . . , 2011.
- [51] C. S. González-González, P. A. Toledo-Delgado, V. Muñoz-Cruz, and P. V. Torres-Carrion. Serious games for rehabilitation : Gestural interaction in personalized gamified exercises through a recommender system. *Journal of Biomedical Informatics*, 97 :103266, 2019.
- [52] S. Gradl, M. Wirth, T. Zillig, and B. M. Eskofier. Visualization of heart activity in virtual reality : A biofeedback application using wearable sensors. In *2018 IEEE 15th international conference on wearable and implantable body sensor networks (BSN)*, pages 152–155. IEEE, 2018.



- [53] N. Graham and T. Urnes. Integrating support for temporal media into an architecture for graphical user interfaces. In *Proceedings of the International Conference on Software Engineering (ICSE'97)*, pages 172–182. ACM Press, 1997.
- [54] J. Gugenheimer, E. Stemasov, H. Sareen, and E. Rukzio. Facedisplay : Towards asymmetric multi-user interaction for nomadic virtual reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–13, 2018.
- [55] Z. He, R. Du, and K. Perlin. Collabovr : A reconfigurable framework for creative collaboration in virtual reality. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 542–554, 2020.
- [56] M. L. Heilig. Sensorama simulator. *US PAT. 3,050,870*, 1962.
- [57] A. Heloir, F. Nunnari, S. Haudegond, C. Havrez, Y. Lebrun, and C. Kolski. Design and evaluation of a self adaptive architecture for upper-limb rehabilitation. In *ICTs for Improving Patients Rehabilitation Research Techniques*, pages 196–209. Springer, 2014.
- [58] Y. Hernández and M. P. Ramírez. Adaptive and blended learning for electrical operators training. In *proceedings of the 8th international conference on computer supported education*, 2016.
- [59] R. Hill, T. Brinck, S. Rohall, J. Patterson, and W. Wilner. The rendezvous architecture and language for constructing multi-user applications. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 1(2) :81–125, 1994.
- [60] A. Hirlehei, H. Mahmoodi, S. Werner, R. Suadamara, Y. Luo, F. Ekadiyanto, M. Lebedev, and A. Hunger. Towards tailorable groupware : A systematic approach to user’s requirement assessment in supporting a user-defined environment of synchronous groupware. In *2010 IEEE International Professional Communication Conference*, pages 50–57. IEEE, 2010.

- [61] X. Huang, F. Naghdy, G. Naghdy, H. Du, and C. Todd. The combined effects of adaptive control and virtual reality on robot-assisted fine hand motion rehabilitation in chronic stroke patients : a case study. *Journal of Stroke and Cerebrovascular Diseases*, 27(1) :221–228, 2018.
- [62] J. Indulska and P. Sutton. Location management in pervasive systems. In *Conferences in Research and Practice in Information Technology Series*, volume 34, pages 143–151. Citeseer, 2003.
- [63] Y. Izountar, S. Benbelkacem, S. Otmane, A. Khababa, M. Masmoudi, and N. Zenati. Vr-peer : A personalized exer-game platform based on emotion recognition. *Electronics*, 11(3) :455, 2022.
- [64] Y. Izountar, S. Benbelkacem, S. Otmane, A. Khababa, N. Zenati, and M. Masmoudi. Towards an adaptive virtual reality serious game system for motor rehabilitation based on facial emotion recognition. In *2021 International Conference on Artificial Intelligence for Cyber Security Systems and Privacy (AI-CSP)*, pages 1–5, 2021.
- [65] A. Jameson. Adaptive interfaces and agents. In *The human-computer interaction handbook*, pages 459–484. CRC press, 2007.
- [66] R. Johansen. *Groupware : Computer Support for Business Teams*. The Free Press, New York, 1988.
- [67] A. Johnson, Y. Tang, and C. Franzwa. knn-based adaptive virtual reality game system. In *proceedings of the 11th IEEE international conference on networking, sensing and control*, pages 457–462. IEEE, 2014.
- [68] C. Joslin, I. S. Pandzic, and N. M. Thalmann. Trends in networked collaborative virtual environments. *Computer Communications*, 26(5) :430–437, 2003.

- [69] H. Kahler, A. Mørch, O. Stiemerling, and V. Wulf. Tailorable systems and cooperative work. *Special Issue of Computer Supported Cooperative Work : The Journal of Collaborative Computing (JCSCW)*, 9(1), 2000.
- [70] C. R. Kelley. What is adaptive training? *Human Factors*, 11(6) :547–556, 1969.
- [71] N. Khezami. *Vers un collecticiel basé sur un formalisme multi-agent destiné à la téléopération collaborative via Internet*. PhD thesis, IBISC (Informatique, Biologie Intégrative et Système Complexes) Laboratory- University of Evry Val d’Essonne - Evry, France, 2005.
- [72] N. Khezami, S. Otmane, and M. Mallem. A new formal model of collaboration by multi-agent systems. In *IEEE KIMAS*, pages 32–37. IEEE, 2005.
- [73] S. Kim, D. Ko, and W. Lee. Utilizing smartphones as a multi-device single display groupware to design collaborative games. In *Proceedings of the 2017 Conference on Designing Interactive Systems, DIS '17*, page 1341–1352, New York, NY, USA, 2017. Association for Computing Machinery.
- [74] T. J. Kline and J.-L. McGrath. A review of the groupware literature : Theories, methodologies, and a research agenda. *Canadian Psychology/Psychologie canadienne*, 40(3) :265, 1999.
- [75] A. Koenig, D. Novak, X. Omlin, M. Pulfer, E. Perreault, L. Zimmerli, M. Mihelj, and R. Riener. Real-time closed-loop control of cognitive load in neurological patients during robot-assisted gait training. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(4) :453–464, 2011.
- [76] G. Krasner and S. Pope. A cookbook for using the model-view controller user interface paradigm in smalltalk-80. *Journal of Object-Oriented Programming*, 1(3) :26–49, 1988.

- [77] J. Kritikos, P. Caravas, G. Tzannetos, M. Douloudi, and D. Koutsouris. Emotional stimulation during motor exercise : An integration to the holistic rehabilitation framework. In *in Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference · 23-27 July 2019*, number October, pages 4604–4610. IEEE, 2019.
- [78] D. Kumar, A. González, A. Das, A. Dutta, P. Fraise, M. Hayashibe, and U. Lahiri. Virtual reality-based center of mass-assisted personalized balance training system. *Frontiers in bioengineering and biotechnology*, 5 :85, 2018.
- [79] J. Lacoche, T. Duval, B. Arnaldi, E. Maisel, and J. Royan. A survey of plasticity in 3D user interfaces. *2014 IEEE 7th Workshop on Software Engineering and Architectures for Realtime Interactive Systems, SEARIS 2014*, pages 19–26, 2014.
- [80] J. Lacoche, T. Duval, B. Arnaldi, E. Maisel, and J. Royan. A survey of plasticity in 3d user interfaces. In *2014 IEEE 7th workshop on software engineering and architectures for realtime interactive systems (SEARIS)*, pages 19–26. IEEE, 2014.
- [81] Y. Lang, L. Wei, F. Xu, Y. Zhao, and L.-F. Yu. Synthesizing personalized training programs for improving driving habits via virtual reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 297–304. IEEE, 2018.
- [82] Y. Laurillau. *Conception et réalisation logicielles pour les collecticiels centrées sur l'activité de groupe : le modèle et la plate-forme Clover*. PhD thesis, Université Joseph-Fourier - Grenoble I, 2002. NNT : tel-00007472.
- [83] D. Lee, M.-W. Lim, S.-J. Han, and K. Lee. Atlas : A scalable network framework for distributed virtual environments. *Presence : Teleoperators and Virtual Environments*, 16(2) :125–156, 2007.

- [84] D.-Y. Lee, Y.-H. Cho, D.-H. Min, and I.-K. Lee. Optimal planning for redirected walking based on reinforcement learning in multi-user environment with irregularly shaped physical space. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 155–163, 2020.
- [85] S. Lee, Y. Lee, S. Jang, and W. Woo. vr-ucam : Unified context-aware application module for virtual reality \*. 2004.
- [86] X. Li, M. Eckert, J.-F. Martinez, and G. Rubio. Context aware middleware architectures : Survey and challenges. *Sensors*, 15(8) :20570–20607, 2015.
- [87] I. Lindt. *Adaptive 3D-user-interfaces*. Verlag Dr. Hut, 2009.
- [88] L. Luo, H. Yin, W. Cai, M. Lees, and S. Zhou. Interactive scenario generation for mission-based virtual training. *Computer Animation and Virtual Worlds*, 24(3-4) :345–354, 2013.
- [89] A. Mariani, E. Pellegrini, N. Enayati, P. Kazanzides, M. Vidotto, and E. De Momi. Design and evaluation of a performance-based adaptive curriculum for robotic surgical training : A pilot study. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2162–2165. IEEE, 2018.
- [90] J. Marín-Morales, J. L. Higuera-Trujillo, A. Greco, J. Guixeres, C. Llinares, E. P. Scilingo, M. Alcañiz, and G. Valenza. Affective computing in virtual reality : emotion recognition from brain and heartbeat dynamics using wearable sensors. *Scientific reports*, 8(1) :1–15, 2018.
- [91] E. Mergenthaler. Shifts from negative to positive emotional tone : Facilitators of therapeutic change. In *Presentation at the 34th Annual Meeting of the Society for Psychotherapy Research, Weimar, Germany, 25-29 June*, 2003.

- [92] M. Miraoui. *Architecture logicielle pour l'informatique diffuse : modélisation du contexte et adaptation dynamique des services*. PhD thesis, École de technologie supérieure, 2009.
- [93] M. Miraoui, C. Tadj, and C. Ben Amar. Context modeling and context-aware service adaptation for pervasive computing systems. *International Journal of Computer and Information Engineering*, 2(3) :782–791, 2008.
- [94] J. Moon, M. Jeong, S. Oh, and T. H. Laine. Data Collection Framework for Context-Aware Virtual Reality Application Development in Unity : Case of Avatar Embodiment. (June), 2022.
- [95] R. Mourning and Y. Tang. Virtual reality social training for adolescents with high-functioning autism. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 004848–004853. IEEE, 2016.
- [96] L. Nigay. *Conception et modélisation logicielles des systèmes interactifs : application aux interfaces multimodales*. PhD thesis, Université Joseph Fourier - Grenoble 1, 1994.
- [97] J. Nirme, A. Duff, and P. F. Verschure. Adaptive rehabilitation gaming system : online individualization of stroke rehabilitation. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6749–6752. IEEE, 2011.
- [98] M. Noma and T. Miyasato. Cooperative object manipulation in virtual space using virtual physics. *Proceeding of Dynamic System and Control ASME*, 61 :101–106, 1997.
- [99] S. Otmane, C. Domingues, F. Davesne, M. Mallem, and J.-J. Kim. Collaborative 3d interaction in virtual environments : A workflow-based approach. In *Virtual Reality*, DOI : 10.5772/13013. InTechOpen, 2011.

- [100] T. B. F. Pacheco, C. S. P. Medeiros, V. H. B. Oliveira, E. R. Vieira, and F. A. C. Cavalcanti. Effectiveness of exergames for improving mobility and balance in older adults : A systematic review and meta-analysis. *Syst. Rev*, 9 :163.
- [101] J. Pascoe. Adding generic contextual capabilities to wearable computers. In *Digest of papers. second international symposium on wearable computers (cat. no. 98ex215)*, pages 92–99. IEEE, 1998.
- [102] N. Paspallis, A. Chimaris, and G. A. Papadopoulos. Experiences from developing a distributed context management system for enabling adaptivity. In *IFIP International Conference on Distributed Applications and Interoperable Systems*, pages 225–238. Springer, 2007.
- [103] J. Patterson. A taxonomy of architectures for synchronous groupware applications. *Journal of ACM SIGOIS Bulletin*, 15(3) :27–29, 1995.
- [104] D. Payet. *L’enrichissement de message comme support pour la composition logicielle*. PhD thesis, University of Montpellier 2, 2003.
- [105] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. Context aware computing for the internet of things : A survey. *IEEE communications surveys & tutorials*, 16(1) :414–454, 2013.
- [106] J. Piech and K. Czernicki. Virtual Reality Rehabilitation and Exergames—Physical and Psychological Impact on Fall Preventio among the Elderly—A Literature Review. *Appl. Sci*, 11 :4098.
- [107] N. Pinkwart and B. M. McLaren. *Educational technologies for teaching argumentation skills*. Bentham Science Publishers, 2012.
- [108] W. E. Piper, J. S. Ogrodniczuk, A. S. Joyce, M. Mccallum, and J. S. Rosie. Relationships among affect, work, and outcome in group therapy for patients with complicated grief. *American Journal of Psychotherapy*, 56(3) :347–361, 2002.

- [109] N. Preguiça, J. L. Martins, H. Domingos, and S. Duarte. Data management support for asynchronous groupware. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pages 69–78, 2000.
- [110] M. A. Razzaque, S. Dobson, and P. Nixon. Categorization and modelling of quality in context information. 2006.
- [111] I. M. Rezazadeh, X. Wang, M. Firoozabadi, and M. R. H. Golpayegani. Using affective human–machine interface to increase the operation performance in virtual construction crane training system : A novel approach. *Automation in Construction*, 20(3) :289–298, 2011.
- [112] K. Riege, T. Holtkämper, G. Wesche, and B. Fröhlich. The bent pick ray : An extended pointing technique for multi-user interaction. In *IEEE Symposium on 3D User Interfaces, 2006*, pages 62–65, 2006.
- [113] E. Rohn. Predicting context aware computing performance. *Ubiquity*, 2003(February) :1–17, 2003.
- [114] M. Roseman and S. Greenberg. Simplifying component development in an integrated groupware environment. In *Proceedings of the 10th annual ACM symposium on User interface software and technology*, pages 65–72, 1997.
- [115] N. Rossol, I. Cheng, W. F. Bischof, and A. Basu. A framework for adaptive training and games in virtual reality rehabilitation environments. In *proceedings of the 10th international conference on virtual reality continuum and its applications in industry*, pages 343–346, 2011.
- [116] D. Salber. *De l'interaction individuelle aux systèmes multi-utilisateurs. L'exemple de la Communication Homme-Homme-Médiatisée*. PhD thesis, Université Joseph Fourier, Grenoble, 1995.



- [117] A. Schäfer, G. Reis, and D. Stricker. A survey on synchronous augmented, virtual and mixed reality remote collaboration systems. *ACM Computing Surveys (CSUR)*, 2021.
- [118] B. N. Schilit. Disseminating active mop infonncition to mobile hosts. *IEEE network*, 1994.
- [119] J. Seo, T. H. Laine, G. Oh, and K.-A. Sohn. Eeg-based emotion classification for alzheimer’s disease patients using conventional machine learning and recurrent neural network models. *Sensors*, 20(24) :7212, 2020.
- [120] C. Shin, H. Kim, C. Kang, Y. Jang, A. Choi, and W. Woo. Unified context-aware augmented reality application framework for user-driven tour guides. In *2010 International Symposium on Ubiquitous Virtual Reality*, pages 52–55. IEEE, 2010.
- [121] R. Slagter. *Dynamic Groupware Services : Modular design of tailorable groupware*. PhD thesis, Telematica Instituut, the Netherlands, 2004.
- [122] R. Slagter, G.-J. Ter Hofte, and O. Stiemerling. Component-based groupware : An introduction. In *Proceedings of CBG2000, the CSCW2000 workshop on Component-Based Groupware*), volume 2, pages 394–400, 2000.
- [123] M. Slater, V. Linakis, M. Usoh, and R. Kooper. Immersion, presence and performance in virtual environments : An experiment with tri-dimensional chess. In *Proceedings of the ACM symposium on virtual reality software and technology*, pages 163–172, 1996.
- [124] J.-S. Sottet, G. Calvary, J.-M. Favre, J. Coutaz, A. Demeure, and L. Balme. Towards model driven engineering of plastic user interfaces. In *International Conference on Model Driven Engineering Languages and Systems*, pages 191–200. Springer, 2005.
- [125] L. Sternberger. *Interaction en réalité virtuelle*. PhD thesis, Université Louis Pasteur, 2006.

- [126] J. Stewart, B. Bederson, and A. Druin. Single display groupware : a model for co-present collaboration. In *Proceedings of the SIGCHI conference on Human factors in computing systems : the CHI is the limit*, pages 286–293, 1999.
- [127] O. Stiemerling, R. Hinken, and A. Cremers. Distributed component-based tailorability for cscw applications. In *Autonomous Decentralized Systems, 1999. Integration of Heterogeneous Systems. Proceedings. The Fourth International Symposium*, pages 345–352, 1999.
- [128] A. Streicher, J. Busch, and W. Roller. Dynamic cognitive modeling for adaptive serious games. In *International Conference on Human-Computer Interaction*, pages 167–184. Springer, 2021.
- [129] A. Streicher and J. D. Smeddinck. Personalized and adaptive serious games. In *Entertainment computing and serious games*, volume 9970, pages 332–377. Springer, Cham, 2016.
- [130] I. E. Sutherland. Sketch pad a man-machine graphical communication system. In *Proceedings of the SHARE design automation workshop*, pages 6–329, 1964.
- [131] I. E. Sutherland et al. The ultimate display. In *Proceedings of the IFIP Congress*, volume 2, pages 506–508. New York, 1965.
- [132] A. Syri. Tailoring cooperation support through mediators. In *Proceedings of 5th European Conference on Computer Supported Cooperative Work*, Lancaster, UK, 1997.
- [133] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface : Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

- [134] M. Taleb, A. Seffah, and A. Abran. Interactive systems engineering : A pattern-oriented and model-driven architecture. *Software Engineering Research and Practice*, pages 636–642, 2009.
- [135] F. Tarpin-Bernard and B. David. Amf : A new design pattern for complex interactive software? *Advances in Human Factors Ergonomics*, 21 :351–354, 1997.
- [136] G. Teege. A csw system model for classifying tailorability implementations. In *Proc. of Workshop on Implementing Tailorability in Groupware*, 1999.
- [137] G. Ter Hofte, H. van der Lugt, and M. Houtsma. Co 4, a comprehensive model for groupware functionality. In *Telematics in a multimedia environment : Proceedings of Euromedia*, volume 96, pages 19–21, 1996.
- [138] D. Thevenin and J. Coutaz. Plasticity of user interfaces : Framework and research agenda. In *Interact*, volume 99, pages 110–117, 1999.
- [139] J. Tisseau. *Réalité virtuelle : autonomie in virtuo*. Habilitation à diriger des recherches, Université de Rennes 1, 2001.
- [140] D. Torres, A. Fernandez, G. Rossi, and S. Gordillo. Fostering groupware tailorability through separation of concerns. *Lecture Notes in Computer Science*, 4715 :143, 2007.
- [141] K. Tsiakas, M. Huber, and F. Makedon. A multimodal adaptive session manager for physical rehabilitation exercising. In *proceedings of the 8th ACM international conference on pervasive technologies related to assistive environments*, pages 1–8, 2015.
- [142] T. Valenzuela, Y. Okubo, A. Woodbury, S. R. Lord, and K. Delbaere. Adherence to Technology-Based Exercise Programs in Older Adults : A Systematic Review. *Journal of Geriatric Physical Therapy*, 41(1), 2018.

- [143] D. Vallejo, C. Gmez-Portes, J. Albusac, C. Glez-Morcillo, and J. J. Castro-Schez. Personalized Exergames Language : A Novel Approach to the Automatic Generation of Personalized Exergames for Stroke Patients. *Appl. Sci*, 10 :7378, 2020.
- [144] S. Verma, D. Kumar, A. Kumawat, A. Dutta, and U. Lahiri. A low-cost adaptive balance training platform for stroke patients : a usability study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(7) :935–944, 2017.
- [145] L. Wang, S. Du, H. Liu, J. Yu, S. Cheng, and P. Xie. A virtual rehabilitation system based on eeg-emg feedback control. In *2017 Chinese Automation Congress (CAC)*, pages 4337–4340. IEEE, 2017.
- [146] L. Wang, X. Liu, and X. Li. Vr collaborative object manipulation based on viewpoint quality. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 60–68, 2021.
- [147] T. Weissker, P. Bimberg, and B. Froehlich. An overview of group navigation in multi-user virtual reality. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 363–369, 2021.
- [148] V. Wulf. Let’s see your search tool! collaborative use of tailored artifacts in groupware. In *Proceedings of the international ACM SIGGROUP conference on Supporting group work*, pages 50–59. ACM, 1999.
- [149] E. Yigitbas, J. Heindörfer, and G. Engels. A context-aware virtual reality first aid training application. In *Proceedings of Mensch und Computer 2019*, New York, NY, USA, Sept. 2019. ACM.
- [150] M. Yoshida, Y. A. Tijerino, S. Abe, and F. Kishino. A virtual space teleconferencing system that supports intuitive interaction for creative and cooperative work. In *Symposium on Interactive 3D graphics (I3D)*, pages 115–122, 1995.

- [151] M. Zahabi and A. M. Abdul Razak. Adaptive virtual reality-based training : a systematic literature review and framework. *Virtual Reality*, 24(4) :725–752, 2020.

Article

# VR-PEER: A Personalized Exer-Game Platform Based on Emotion Recognition

Yousra Izountar <sup>1,2,\*</sup> , Samir Benbelkacem <sup>2</sup>, Samir Otmane <sup>3</sup>, Abdallah Khababa <sup>1</sup>, Mostefa Masmoudi <sup>2</sup> and Nadia Zenati <sup>2</sup>

<sup>1</sup> Département d'Informatique, Faculté des Sciences, Campus El Bez. Sétif, University Ferhat Abbas Setif 1, Sétif 19137, Algeria; akhababa@univ-setif.dz

<sup>2</sup> Centre de Développement des Technologies Avancées (CDTA), Dévision Productique & Rebotique (DDR), Equipe Interaction Homme Système Réalité Virtuelle & Augmentée-IRVA, Baba Hassen, Alger 16303, Algeria; sbenbelkacem@cdta.dz (S.B.); mmasmoudi@cdta.dz (M.M.); nzenati@cdta.dz (N.Z.)

<sup>3</sup> IBISC Lab, Université Paris-Saclay, University Evry, 91020 Evry, France; samir.otmane@univ-evry.fr

\* Correspondence: yousra.izountar@univ-setif.dz

**Abstract:** Motor rehabilitation exercises require recurrent repetitions to enhance patients' gestures. However, these repetitive gestures usually decrease the patients' motivation and stress them. Virtual Reality (VR) exer-games (serious games in general) could be an alternative solution to address the problem. This innovative technology encourages patients to train different gestures with less effort since they are totally immersed in an easy to play exer-game. Despite this evolution, patients, with available exer-games, still suffer in performing their gestures correctly without pain. The developed applications do not consider the patients psychological states when playing an exer-game. Therefore, we believe that is necessary to develop personalized and adaptive exer-games that take into consideration the patients' emotions during rehabilitation exercises. This paper proposed a VR-PEER adaptive exer-game system based on emotion recognition. The platform contain three main modules: (1) computing and interpretation module, (2) emotion recognition module, (3) adaptation module. Furthermore, a virtual reality-based serious game is developed as a case study, that uses updated facial expression data and provides dynamically the patient's appropriate game to play during rehabilitation exercises. An experimental study has been conducted on fifteen subjects who expressed the usefulness of the proposed system in motor rehabilitation process.

**Keywords:** 3D Exer- game; deep learning; adaptation; rehabilitation; emotion recognition; facial emotion recognition (FER); CNN; virtual reality (VR)



**Citation:** Izountar, Y.; Benbelkacem, S.; Otmane, S.; Khababa, A.; Masmoudi, M.; Zenati, N. VR-PEER: A Personalized Exer-Game Platform Based on Emotion Recognition. *Electronics* **2022**, *11*, 455. <https://doi.org/10.3390/electronics11030455>

Academic Editors: Juan M. Corchado and Michael Wehner

Received: 21 November 2021

Accepted: 28 January 2022

Published: 3 February 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Motor rehabilitation helps people to bring up dysfunctions in various parts of the body such as the lower or upper limbs [1]. The biggest challenge patients may face during training is having to repeat the exercises over and over. This may expose them to severe boredom, and later, affect their performance in exercises and lead to an imbalance in their treatment program that may be more and more extended. Recently, many researchers have been interested in exer-games and virtual reality, which they consider a viable solution to this issue. Exer-games are video games with user interfaces that require active participation and physical exertion from players. These exer-games are meant to detect the body movement and provide participants with both enjoyment and exercise. Exergames can be used to motivate older people to engage in physical activity [2,3]. Moreover, they can be used as an alternative to exercise because they are widely available, allow for autonomous practice, and provide control and personalized adjustment possibilities. Exer-games have been shown to be effective in enhancing motor and cognitive skills, minimizing the risk of falling [4], and increasing quality of life of old individuals [5], and people living with chronic diseases [6]. Furthermore, they may be used to enhance flexibility and balance [7],

and improve cognitive inhibition. These technologies have helped many patients with their motor rehabilitation programs. Researchers [8,9] have found that VR can improve motivation and engagement when used in combination with exer-games. The use of exer-games became indispensable in several clinical cases. Thus, several projects have been developed and experienced with patients [10–13]. Virtual reality exer-games facilitate the exercises and give patients an impulse to finish the exercises as a kind of pleasure rather than the pain and boredom during motor rehabilitation sessions [14]. Despite the great success of these exer-games, patients' behavior and responses vary, even if they play the same exer-game. This was the reason for the appearance of personalized or adaptive exer-games. It can be used in several fields (health, training, learning . . . ) for their ability to motivate the user [15]. According to [16], the differences in the physical and gaming capabilities of the players requires providing a personal experience to achieve the targeted positive results. In addition, there are many studies that have proved the effectiveness of personalized and adaptive exer-games by conducting experiments in motor rehabilitation clinics. The game proposed by [17] was well accepted by the patients, and the results showed that the adaptive game is stimulating and attractive. In other experiments carried out by [18], it was noted through that adapting the gameplay parameters improved the physiological performance of the participants. Patients are sometimes faced with hard games that may stress them. This forces them to give up the game quickly. To avoid this tedious and hard situation, adaptive exer-games can be a relevant solution. In general, an adaptive system refers to automatically fitting any context of use continuously; performances and capacities have to be guaranteed too. Consequently, an adaptive game becomes a breakthrough for rehabilitation specialists in clinical situations. This adaptation can be perceived from various perspectives. However, the biggest challenge facing researchers in this field is how to adapt it to individual patients. End-users need access to adaptive interfaces to meet these requirements [19,20]. In this regard, the personalized exer-game addresses this issue. Several studies in the literature focused on such games, some of them are interested in static adaptation [21,22], which makes changes only at the beginning of the game, as opposed to dynamic adaptation, which allows you to adapt the game during the long period of play [17,22–27]. Even if the adaptation is dynamic, it is according to parameters that do not really express the situation of the patient such as the score, or other systems that rely mainly on specialists to intervene in every adaptation process [20,24].

During training, many researchers, including [28–30], have confirmed the existence of a relationship between the patient's feelings and performance, as well as the results of physical therapy. In short-term group psychotherapy, [28] discovered that the experience and expression of positive emotions was related to positive therapeutic outcomes, whereas the intense expression of negative emotions was linked to negative outcomes. It was found in [29] that therapeutic change seemed to be related to the experience of positive emotions, while the experience of only negative emotions was associated with a bad therapeutic outcome. The goal of the research conducted by [30] is to look into the direct relationship between the patients' performance quality and a positive or negative emotional state that has been artificially induced. Statistical analysis of the resulting EMG and Accelerometer signals revealed a significant difference in the quality of each subject's physical exercise induced by various emotional triggers. All observer-based emotion recognition tools use facial expressions as a reliable indicator of positive and negative emotions. Seven basic emotions are universal to human beings, namely: neutrality, anger, disgust, fear, happiness, sadness, and surprise [31]. One study [32] proposed a dimensional approach, in which any emotion is represented in terms of three basic dimensions: valence (positive/pleasurable or negative/unpleasurable), arousal (engaged or not engaged), and dominance (degree of control that a person has over their affective states). In this paper, we will base on the valence dimensions (positive/negative). According to the above model, we will classified the emotions into three classes: positive (happy), negative (angry, disgust, fear, sad) and neutral (neutral). As for the feeling of surprise, some of them considered it positive [33,34], negative [35] or both [36,37]. Therefore, in this paper, we will divide the surprise probability

into two halves. Thus, we will add half to the positive emotion probability and the other half to the negative emotion probability.

Facial emotions can be recognized in different ways that are divided in the literature into two parts: electrical and non-electrical methods [38]. The proposed solution adapts the exer-game to the patient based on his feeling. If he shows negative emotions, the system makes changes to reduce the difficulty of the game. On the other hand, if he is happy (positive emotions) and constructed (i.e., he can easily do it), the system increases the difficulty. This is exactly what the doctor does; he asks the patient to decrease or increase the exercise speed.

Computer vision researchers are using this technology to develop approaches for automatically recognizing emotions from facial expressions, thanks to advancements in automatic facial expression analysis. Therefore, we proposed a case study that allows us to know the patient's emotion using an artificial intelligence method (CNN), by analyzing facial expressions of the patients.

Thus, our contribution through this work aims at developing the VR-PEER platform in which we implement facial expression and adaptive exer-games modules. The platform is divided into three phases: (1) the capturing and interpretation phase, in which we capture and interpret the contextual information of the patients; (2) analysis phase: in this phase, we analyze the interpreted information and recognize the emotions of the patient; (3) adaptation phase: the most important part of our architecture in which we make changes and adjustments to the exer-game according to the patient's mood. The difficulty level of the game varies according to the patient's facial expression during the exercise. For example, if we detect that the patient is sad, angry, fear or disgust, the system lowers the difficulty level of the exer-game, and vice versa if the patient is happy.

The upcoming pages will provide more details about our contribution. Firstly, we shall enumerate several related work. Secondly, we will present the proposed architecture. Thirdly, we will represent the study case of VR-PEER architecture. Finally, we will validate our solution with an experimental setup.

## 2. Related Work

The idea of introducing gamification into the therapeutic protocol of upper limbs rehabilitation was created to motivate patients during the rehabilitation programs. There are numerous studies looking for ways to develop serious games for rehabilitation, including what our team did, aiming at creating serious games in a virtual reality environment [39]. Several serious games for rehabilitation projects have appeared and showed a wide general interest in improving and sustaining virtual reality technology [8,40,41]. The core idea of VR-based rehabilitation is to use sensing devices to capture and quantitatively assess the movements of patients under treatment in order to track their progress more accurately [42]. Two of the sensing devices that can also be considered as natural user interfaces (NUI) are the Microsoft Kinect sensor and the Leap Motion Controller [43]. In the same context, there are several implementations that use, principally, these two types of devices to control the games and provide feedback to patients. On the other hand, we have two categories of developed games: non-personalized games and personalized games. In fact, non-personalized games represent those which could be used for a wide range of patients without operating changes on the exer-game according to the patients' states, while personalized games were conceived to be suitable for each patient's state changes.

Several works were carried out on non-personalized games. In [44], a Stable-Hand VR system was developed for hand rehabilitation. It uses a VR device and a leap controller to manage the hand's movement. The exer-game focused on pinching gestures, supporting three tasks: (a) open-hand; (b) pinch position and (c) open-hand. The researchers in [39] developed fruit-based games and a runner game for rehabilitation of post-stroke upper limbs using Kinect with Oculus Rift HMD. The researchers in [23] were interested in cognitive rehabilitation and used only Oculus Go HMD. The mechanics of the game require memorizing geometric shapes while the player works in two modes, autonomous or



manual. They defined two levels of difficulty depending on the number of elements to remember. The exer-game developed in [45] targeted to manage hand disabilities. Five hand gestures were selected to control a sphere in an environment composed of obstacles (increase speed, go backwards, jump, and make the sphere bigger or smaller). The goal is reached when the patient drops the ball in a basket. The system developed used CNN for hand gesture recognition via EMG signals.

Previous work on serious games has found the need for personalization to deliver appropriate levels of exercise to individuals and to make the game progressively more challenging to keep people engaged [46,47], and offer a personalized experience, i.e., adapting the game to the learners' characteristics, situation, and needs.

Consequently, some work was carried out on personalized games. The researchers in [26] proposed a VR framework for Parkinson's disease rehabilitation. The proposed system is designed to capture various movements of the upper limbs, unilateral or bilateral, and the captured motion is represented one-to-one in the virtual world of the video game. The difficulty of the game can be monitored remotely by the clinician. One study [22] investigated the use of free-hand interaction using gesture tracking sensors such as leap motion to support hand rehabilitation of patients with strokes or trauma. The researchers in [17] developed a Tetris-like game for motor rehabilitation after stroke. This system uses the Oculus Rift display device with Razer Hydra Motion Controllers, where patients perform pronation/supination movements with their hands. The difficulty of the exer-game changed dynamically according to the score obtained at a given time. The Rehab Bots game was created in [24]. It is based on the virtual assistant identified by therapists and contains three main levels that are: (1) a virtual assistance robot that shows patients how to properly perform the exercises, (2) an intelligent movement notifications module that analyzes and represents patients in 3D, then, if necessary, adjusts the game to the appropriate level to achieve the best outcome, and (3) a dynamic correction module that takes into account both the level of game complexity and the virtual assistant readings in order to create a series of exercises that are more suited to the patients' capacities. Personalized Exer-game Language (PEL) [20] was used to generate exer-games with a focus on young people [21]. Azure Kinect DK was used to analyze the movement of the entire body. One study [48] proposed an open-source framework to develop serious games using client/server architecture to facilitate the connection between different platforms. Two examples of serious games were played: the classical Gym-Tetris game and a problem-solving task game. The disadvantage of the last two strategies is that modification needs the involvement of kinesthetic rehabilitation experts. This means that the system will not be able to adapt on its own. To offer a personalized serious game for rehabilitation, [49,50] propose a recommender system, based on interaction analysis and preferences of the user. A lot of suggestions about serious adaptive games are directed at adults in general, without caring for young people in particular, as they quickly get bored of rehabilitation exercises. That is why [21] suggests a serious game adaptation for young people, where they develop serious games that support plug-and-play metaphors for natural interaction. The author in this study insists on the need to personalize the rehabilitation process for each individual. Despite the effort made in personalized exer-games development, the emphasis, however, is on the game difficulty progress, not on the patient's states. With personalized games proposed in the literature, it is difficult to know whether the patient is in good or bad situation. Thus, patients can present situations of stress, fatigue, or anxiety without the game detecting this. Here, the exer-game leads to a negative performance. Therefore, we are interested in studying the emotions of patients in the exer-game. Some studies have been interested in using this method to adapt serious games for learning. Thus, these studies did not specifically address motor rehabilitation. For example, in [51], the authors present a fruit-slicing exer-game to help a hemiparetic person, adaptive serious game in real-time. It is based on KINECT to recognize facial expressions and adjust the game's complexity according to the player's emotions. The real-time adaptation is executed using Facial Action Coding System (FACS) as described in [37]. FACS is a method of describing

facial movements which are used to find the emotion of the person. Another study [52], had been carried to predict the players' emotions. However, he did not use it for adaptation, but only to assess the learner's cognitive states during the educational video game, they used binomial logistic regression method to predict the cognitive-affective states of flow.

The table below presents some previous studies and compares them to the proposed solution in this paper:

As we mentioned earlier, we aspire to find a way that enables us to develop serious games for physical rehabilitation, which can be adapted to the patient's mood dynamically during exercise on a permanent basis. We notice from Table 1 that there are studies that have developed serious games for rehabilitation, but they are not adapted to the patient [3,40]. In the same context, if they are adapted they are personalized but not intended for motor rehabilitation [50], or the adaptation is only at the beginning of the exercise, regardless of what happens during it [17,19,24]. The previous works may be oriented to kinesthetic rehabilitation and adapted dynamically to the game's surroundings, but they do not take into account the virtual reality aspect [46,49]. There are works, however, that track the patient's state while playing, but they are unrelated to virtual reality or kinesthetic rehabilitation [51,52]. Therefore, the solution we propose will include all these shortcomings. It is suitable for virtual reality, directed to motor rehabilitation, and depends on the patient's emotions dynamically for exer-game adaptation.

**Table 1.** Comparison between our proposition VR-PEER and few existing methodology.

Reference	Year	VR Exergame?	Exergame for Rehabilitation?	Personalized Exergame?		Real-Time Patient's State?
				Static	Dynamic	
[22]	2017	✓	✓	✓	✓	
[3,40]	2017	✓	✓			
[46]	2018		✓		✓	
[51]	2018				✓	✓
[49]	2019		✓		✓	
[50]	2020			✓		
[17,24]	2020	✓	✓	✓		
[44]	2020	✓	✓			
[52]	2020					✓
[19]	2021		✓	✓		
VR-PEER	2021	✓	✓		✓	✓

Thus, we propose an approach for a dynamic exer-game depending on patients' facial emotions. We can switch from a difficult to an easy level of an exer-game according to the patients' face states (e.g., positive or negative). The main contribution of this paper will be detailed in the next section.

### 3. VR-PEER Architecture

Proposing an architecture for exer-games is a very challenging task, especially concerning the personalization aspect due to the complexity of virtual reality systems. VR-PEER (Figure 1) has a highly modular structure in which an adaptive exer-game can be easily generated. Our architecture is divided into three main phases, namely:

(i) Capturing and Interpretation, (ii) Analysis, and (iii) Adaptation.

Typically, the architecture of a personalized exercise includes observations of the environment, analyzing and evaluating the captured data, and finally, selection of adaptations and their executions.

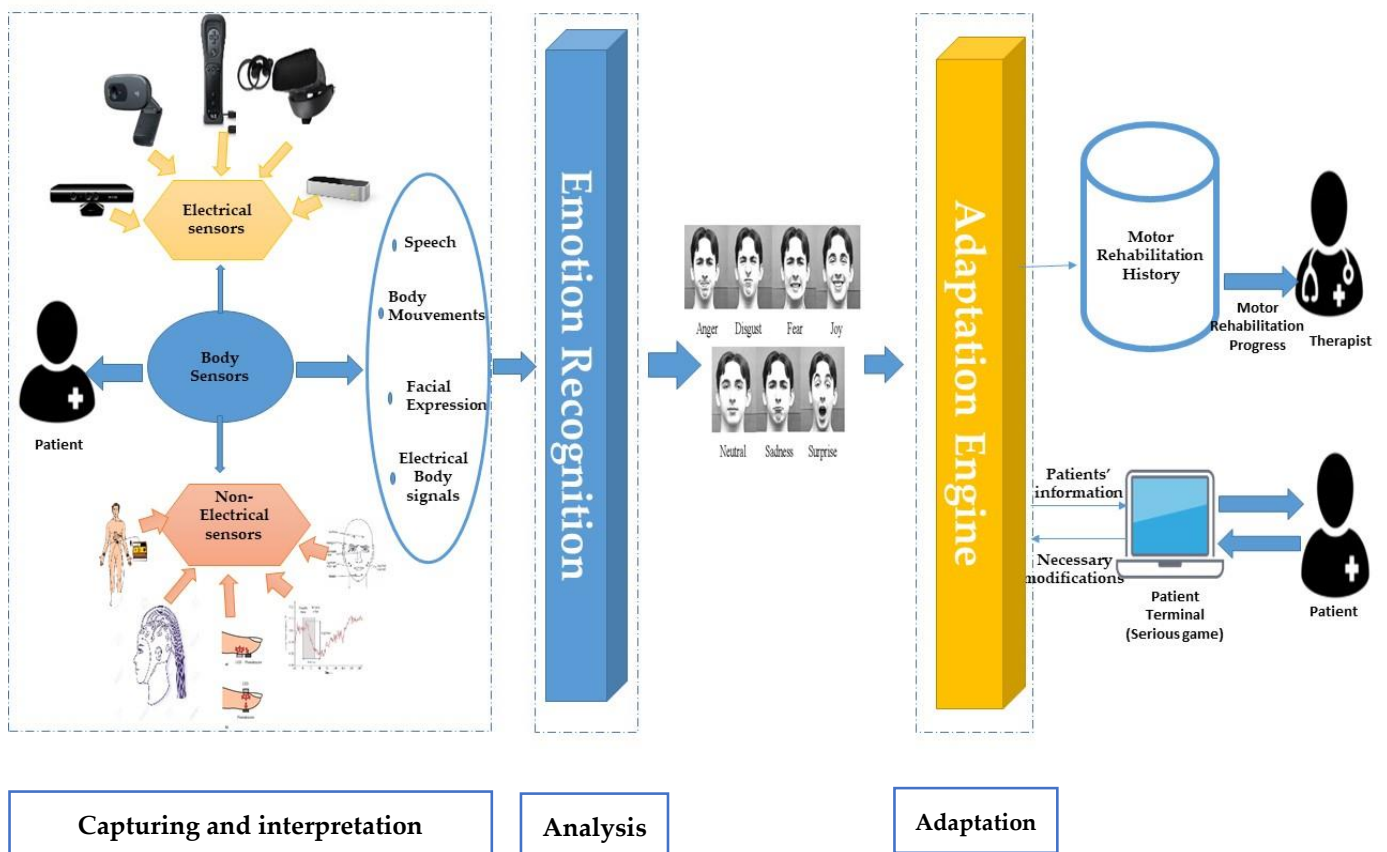


Figure 1. VR-PEER architecture.

### 3.1. Capturing and Interpretation Phase

This phase forms the basis of our system. In order to be able to intercept context data, the capture layer presents a collection of sensors. One study [38] proposes a classification of these sensors to recognize human emotions automatically: (i) electrical sensors such as EEG, ECG and EMG signals and (ii) non-electrical sensors like Kinect, Web Camera and Oculus. Basing on this classification in our architecture, each type of sensor must be attached to a software component allowing access to the intercepted data. This layer contains two parts: a data provider and a data interpreter. The data provider collects contextual information from the user’s runtime environment, called “meta information” (e.g., captured patient photo, patient speech), while the data interpreter uses this information to generate a high-level representation to describe the context information and make it more exploitable and easier to manipulate (e.g., ECG signal, photo pixels).

### 3.2. Analysis Phase

In this paper, we seek to propose a system solution to adapt exer-games according to the patient’s situation. During the rehabilitation exercises, the patient performs several exercises for his limbs, which may change his psychological state and his feelings. He can be in pain, fear, or happiness. We thought of recognizing the patient’s emotions to adapt to the exer-game he was going to have. Emotion recognition is the process of identifying human emotions, most typically from facial expressions, human speech, etc. The use of multiple sources can lead to conflicting situations and contradictory results, which lead to imprecise or even totally incorrect situations. Therefore, this layer must have some form of intelligence in order to intercept and resolve these conflicts. To accomplish this process, we propose the use of one of the artificial intelligence methods such as recurrent neural networks (RNN) or convolutional neural networks (CNN) (which is the method that we used in our case study), etc.

### 3.3. Adaptation Phase

This is the most important stage. The objective of this phase is to adapt the exer-game according to the emotion transmitted by the analysis phase. It contains three parts: (i) Adaptation Engine: responsible for defining the various adaptation mechanisms as well as the various reactions that the system must perform following a change in the patient's emotions. In the field of rehabilitation, several decisions and changes must be made in the metrics of motor rehabilitation (rehabilitation time, number of repetitions, and the difficulty of the exercise . . . ). (ii) Motor rehabilitation history: every change will be saved in this part of the system, with the goal of following the progress of patients, preparing reports, and finally sending them to the therapists. Indeed, with this part, we can have access to various data whenever needed. (iii) Exercise or application: this part must be subscribed to the adaptation engine layer in order to make the appropriate changes in the game, to adapt to his psychological state of the patient.

#### Mathematical Description of Adaptation Phase

We will briefly describe the adaptation process in the form of a mathematical equation that contains a number of inputs. Each input represents the law that is applied during the adaptation process, provided that it is approved by the therapist so that the development of the game in terms of difficulty is logical and appropriate for patients during motor rehabilitation. Each input contains, in its turn, two inputs that are the parameters to be changed and the patient's emotions at the moment of the adaptation process.

The program contains three exits: what to do if the patient feels pain (negative emotions), the second, if he does not feel pain (positive emotions), and the third is the exit condition. It is useful when the exercise must be changed, i.e., the condition will be executed if the patient is happy and has completed the exercise. We propose using some other parameters to decide if the patient has completed the exercises or not. In our case, we will take average of the gesture numbers completed in every 30 s. Thus, the adaptation in the same exercise will be based on the patient's emotions when the passage relies on another parameter chosen by the therapist. It should be noted that each exercise has this adaptation equation (see Equation (1)) that changes the number of inputs and outputs according to it, and that it is implemented at every moment of the system adaptation to the patient in real time. In our case, for example, we chose the time range for executing this equation to be every 10 s. This value can be increased or decreased depending on the exercise, game, or training period. An example of how to apply this function will be given in the next session.

$$\text{Adaptation}(F_1(P_1, E_1), F_2(P_2, E_2), \dots, F_n(P_n, E_n)) = (AP_1, AP_2, \dots, AP_n) \quad (1)$$

where,  $F_n$ : equation using to change the parameter to get an adaptive exer-game;  $P_n$ : adaptation parameter;  $E_n$ : patient's emotion in the moment of adaptation;  $AP_n$ : the value of the parameter after the adaptation process.

## 4. VR-PEER Case Study

In interpersonal communication, facial expressions (FE) are one of the most important information conduits. We can indicate (and recognize) a variety of emotions through our facial expressions. We have applied the VR-PEER architecture presented in the previous section, in which we chose the device used in the capturing phase and the method for recognizing emotions based on facial expressions.

### 4.1. Adaptation Based Facial Emotion Recognition

The architecture consists of three modules, as shown in Figure 2:

- Capturing and analysis module: At this stage, a video of the patient's face is captured during the rehabilitation program, in the form of a set of images (frames). Each image is then analyzed to detect and extract the face that will be used later for recognition.
- Module for Facial Emotion Recognition (FER)

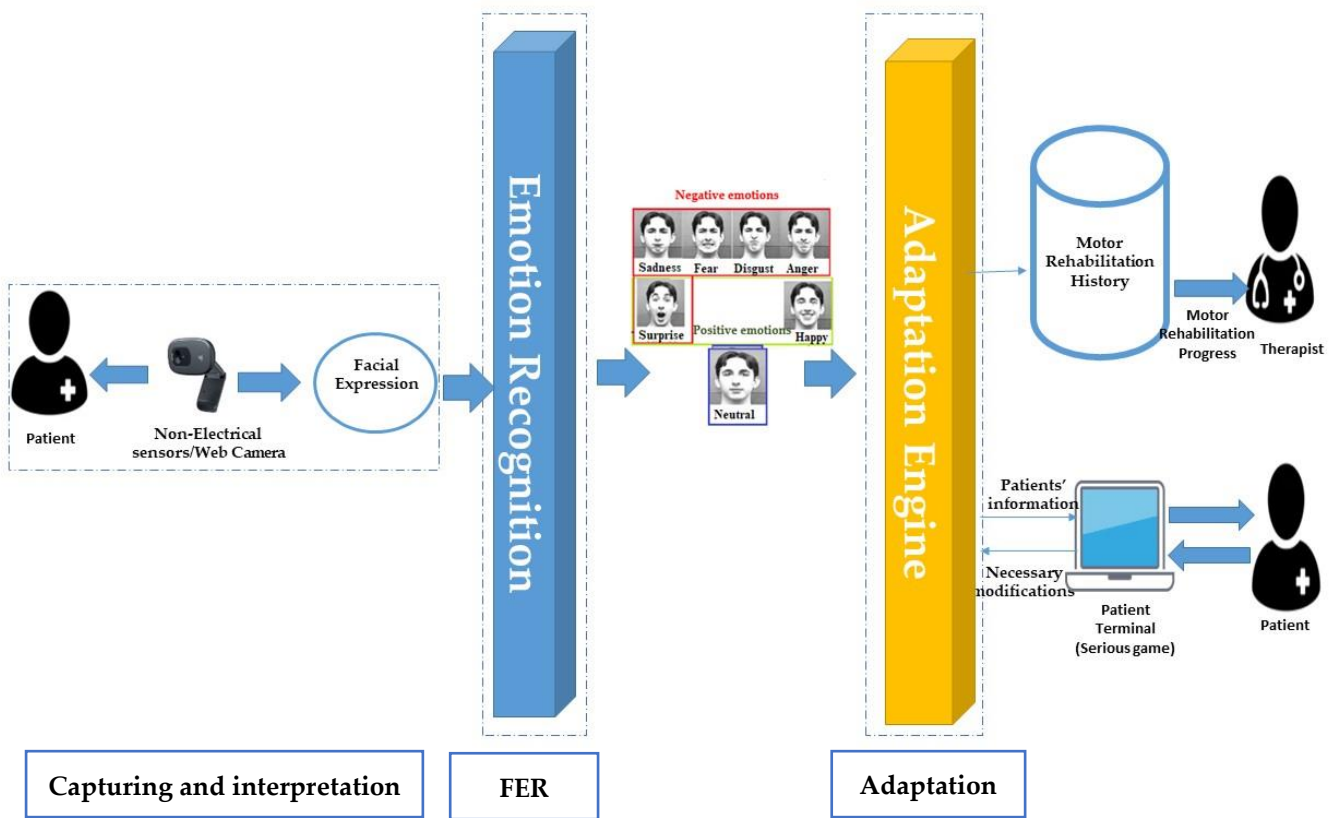


Figure 2. Applying VR-PEER architecture with facial emotion recognition using web camera.

We focus in this process on recognizing facial emotions of patients using the convolutional neural networks (CNN) and the DeepFace model proposed by [53]; knowing that we used in our solution its training set.

DeepFace is one of the most popular facial recognition models. It consists of 9 layers. The first layer consists of an input image with dimensions of  $152 \times 152 \times 3$ . It is convolved with 32 filters of size  $11 \times 11 \times 3$ , resulting in dimensions of  $32 \times 11 \times 11 \times 3 \times 152 \times 152$ . The second layer is a pooling operation with filter size  $3 \times 3$  and stride of 2 separately for each channel. Hence, the dimensions of resulting image will be  $14 \times 14 \times 6$ . Similarly, the third layer also involves a convolution operation with 16 filters of size  $9 \times 9 \times 16$ . The purpose of these three first layers is to extract texture and edges, followed by three locally connected layers (L4, L5 and L6). Finally, DeepFace is followed by two fully connected convolutional layers (F7 and F8). The output of the last fully connected layer is fed to a K-way softmax (where K is the number of classes), which produces a distribution over the class labels. In our case, the final layer will be a softmax output layer with '7' possible classes. Moreover, we will look at these seven emotions that can be classified into three classes, called 1) "positive emotion" that includes happy emotions (Equation (1)), "negative emotion" that includes (angry, sad, surprised, Disgusted, and Afraid) emotions described in (Equation (2)), and "neutral emotion".

$$P(\text{positive}) = P(\text{happy}) + P(\text{surprised})/2 \tag{2}$$

We have, negative = angry  $\cup$  sad  $\cup$  surprised  $\cup$  disgust  $\cup$  afraid, so the equation will be:

$$P(\text{negative}) = P(\text{sad}) + P(\text{angry}) + P(\text{disgust}) + P(\text{surprised})/2 + P(\text{Afraid}) \tag{3}$$

During training, the web cam captures facial expressions and calculates P (negative), P (positive), P (normal) in each frame, and the adaptation is done every Time = T,



we must therefore calculate the average emotion (positive, negative, neutral) at each T. (see Equations (3)).

$$\text{Average} = \left( \sum_{i=1}^{i=n} P_i(\text{Emotion}) \right) / n \quad (4)$$

where, Average = {APos, ANeg, ANeu}, Emotion = {positive, negative, normal},  
N: Number of frames in T.

Thus, the result of process will be, Max(APos, ANeg, ANeu).

- Real-time-adaptation module:

This part receives the patient's mood. There are three possibilities for these outcomes that are either positive, neutral, or negative emotions. Real-time adaptation module depends on this result to change the game to be suitable and adapted to the patient's emotion; it becomes more difficult if the patient is in a good condition, and the difficulty decreases if the patient feels physical pain. However, if the patient is in a normal condition, the game is kept as it is. Based on the result of the previous process to adapt the 3D exergame automatically, the patient does not need to change the parameters of the exergame manually. Our system will be intelligent, as it will pay attention to every change that will occur in the patient's face emotion in real time. The adaptation process will be executed with every change in the patient's emotion.

#### 4.2. Experimental Setup

In order to evaluate the system that we previously suggested, we developed a simple 3D exergame for the rehabilitation of the upper limbs. We, first, determine the patient's mood based on the detection of facial emotions (positive, negative, or neutral), and then increase or decrease the game difficulty.

##### 4.2.1. Logical Architecture of Our Serious Game

A serious game was developed using Unity 3D (Figure 3). The exergame is developed based on the architecture proposed in advance. We have used a web camera as a sensor whose role is to track the patient's face. The video will be used by Unity to detect real-time patient emotions based on the combination of the Mood Me [54] asset and the DeepFace framework [53]. Deepface is a framework to be used in Python, and Mood me can be used in Unity. We have extracted the neural network (weight model) from DeepFace, and then, included the model in the Mood Me asset. We based our system on this asset to preprocess the real-time photos of the patient, and to detect his face during training using the Barracuda package [55]. Then, Mood Me used the DeepFace model to predict the real-time users' emotions. The emotion result will be the entry of the adaptation equation to decide which actions will be executed in the game.

Our game, developed as a 3D application for rehabilitation, is based on 3D interaction (I3D) using the Leap Motion Controller [56]. In this work, the virtual headset is not planned. The idea is to have an easy-to-use application in hospitals and reeducation centers. Thus, to avoid cognitive overload, we used a large screen and a leap motion to create a virtual environment with semi-immersion, and ensure the 3D interaction of the patient.

In this game, we considered only one gesture. The table below represents the description of the gestures and adaptation parameter (Figure 1). The gamer will extend the arm completely and move it from top to bottom and similarity from bottom to top, to drop the apple into the floor. One apple will appear every  $\text{Time}_{\text{interville}}$ .

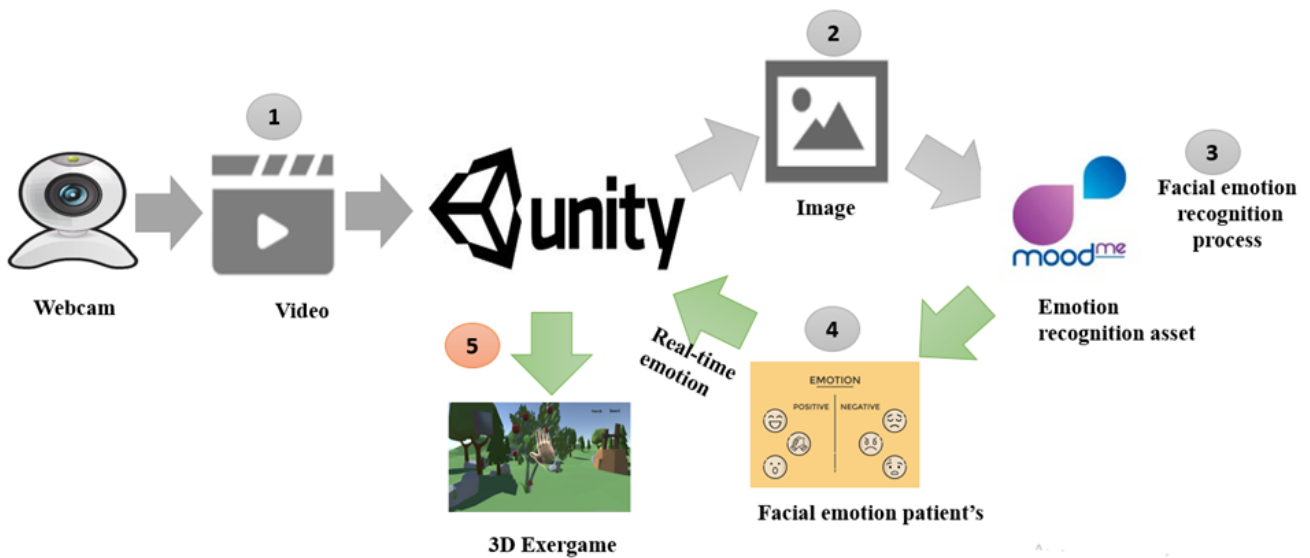


Figure 3. The logical architecture of the developed game.

As mentioned earlier, the patient in this game tries to extend his arm to drop the apples to the ground (see Figure 4). We derived from the equation previously cited in the adaptation process another adaptation equation with predefined parameter  $Time_{interval}$  and function (see Equation (6)), containing one entry and one exit (see Equation (5)).  $Time_{interval}$  changed by the adaptation’s law (Equation (6)).

$$Adaptation(F(Time_{interval})) = Time'_{interval} \tag{5}$$

$$F(Time_{interval}) = \left\{ \begin{array}{l} Time\_interval = 5 \\ Time_{interval} - 1 \quad \text{if emotion} = \text{positive} \\ Time_{interval} + 1 \quad \text{if emotion} = \text{negative} \\ Time_{interval} = 15 \quad \text{if } Time_{interval} > 15 \\ Time_{interval} = 3 \quad \text{if } Time_{interval} < 3 \end{array} \right\} \tag{6}$$

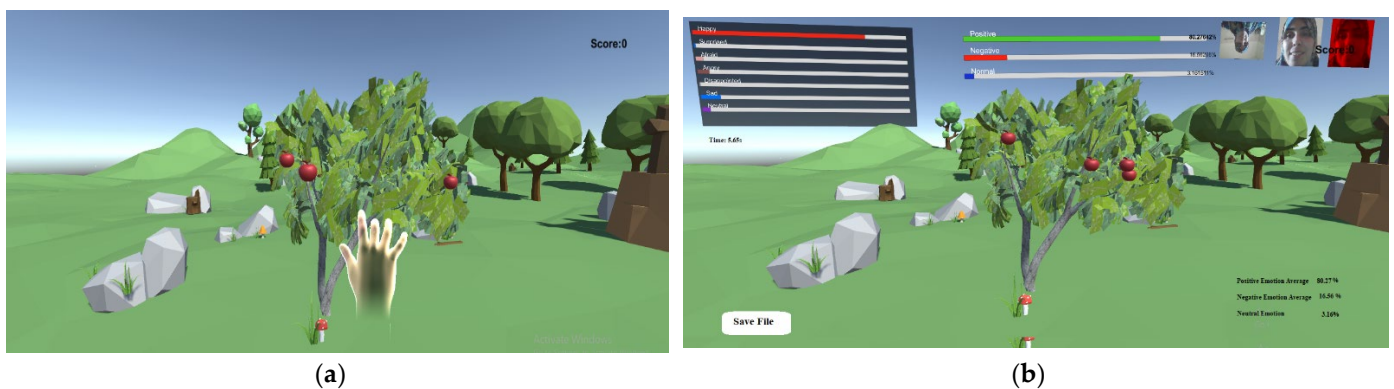


Figure 4. Adaptive serious game developed based on VR-PEER architecture. (a): participant’s interface, (b): interface used to test the facial emotion recognition.

Our purpose is to make the patient feel that the game is designed for them because it adapts to their facial emotions, whether they feels pain caused by pressure on their limbs and wants exercises to be easier, or is willing to do more difficult exercises.

### 4.2.2. Tests and Evaluation

We carried out two types of tests to prove the efficiency and performance of our system. The first is to prove the efficacy of the first part of the Facial Emotion Recognition process, while the second is for testing the adaptation process. This first phase of our system evaluation is done to evaluate the performance and robustness of the adaptation system by taking into account facial emotions. We, therefore, seek to answer the following question: is the adaptation robust or not. Thus, the goal of this evaluation is to know if the adaptation is done independently of the patients. We used a laptop with the following characteristics: I7–9th generation processor, with a GeForce NVidia graphics card, a Web Camera integrated into the pc, and a leap motion controller.

- Emotion test

We made a 10-min test. Facial emotion recognition (FER) using the DeepFace method is done in each frame since there are 24 frames in every second, i.e., the number of emotional recognitions during this period will be 14,400. The adaptation will be executed every 10 s. As a result, we will calculate the average of the positive, negative, and neutral emotions (results represented in Figure 5). We will obtain 60 values calculated with the equation below:

$$\text{Average} = \left( \sum_{i=1}^{i=240} P_i(\text{Emotion}) \right) / 240 \tag{7}$$



Figure 5. Emotion-state progression values for case study subject.

During testing, we found that the system succeeded in recognizing the predominant user emotions every 10 s, and 57 out of 60 tests were correct. From the three faulty tests, which were represented by the confusion between happy and neutral emotions, we conclude that our system is effective in most cases, except a few ones, which is logical as it is difficult to know whether a person is happy or normal.

- Adaptation test

Our system is divided into two parts: a part to determine the patient’s emotion and a part to adapt the game according to his facial expressions. In the first, we used a method that was proven in the previous study, meaning that there is no need to re-test it. As for the second part, in this table, we tested the response of the adaptation process to the part of the emotion process.



We have tested the adaptation process on a group of 15 subjects who have tried the game (6 men and 9 women ranging from 21 to 60 years old). Overall, 70% of them have never used VR serious games and 15% are not used to VR games but tried them before; the rest are used to such games. The duration of each test is 5 min. Subjects change their facial expressions during exercises randomly.

The adaptation process is executed every 10 s; thus, the number of adaptations (points check) during the test is 30. Every 10 s, the game will be adapted according to the subject emotion (positive, negative, or neutral). We have to save every 10 s each subject's emotion, the tested adaptation parameter, and theoretical adaptation parameter (see Table 2).

**Table 2.** Example of the data saved for each subject during the training.

Emotion	Time Interval Tested	Theoretical Time Interval	True or False Adaptation?
Neutral	5	5	True
Positive	4	4	True
Positive	3	3	True
positive	3	3	True
Negative	4	4	True
Positive	3	3	True
Neutral	3	3	True
Neutral	3	3	True
Positive	3	3	True
Negative	4	4	True
Negative	5	5	True
Negative	6	6	True
Neural	6	6	True
Neural	6	6	True
Neural	6	6	True
Positive	5	5	True
Negative	6	6	True
Negative	7	7	True
Positive	6	6	True
Positive	5	5	True
Negative	6	6	True
Negative	7	7	True
Neural	7	7	True
Neural	7	7	True
Neural	7	7	True
Positive	6	6	True
Negative	7	7	True
Negative	8	8	True
Positive	7	7	True
Positive	6	6	True
True results:30			False results: 0

In the chart below (Figure 6), we have changed the Time<sub>interval</sub> value when applying the adaptation system (tested and theoretical) and when not applying it too.

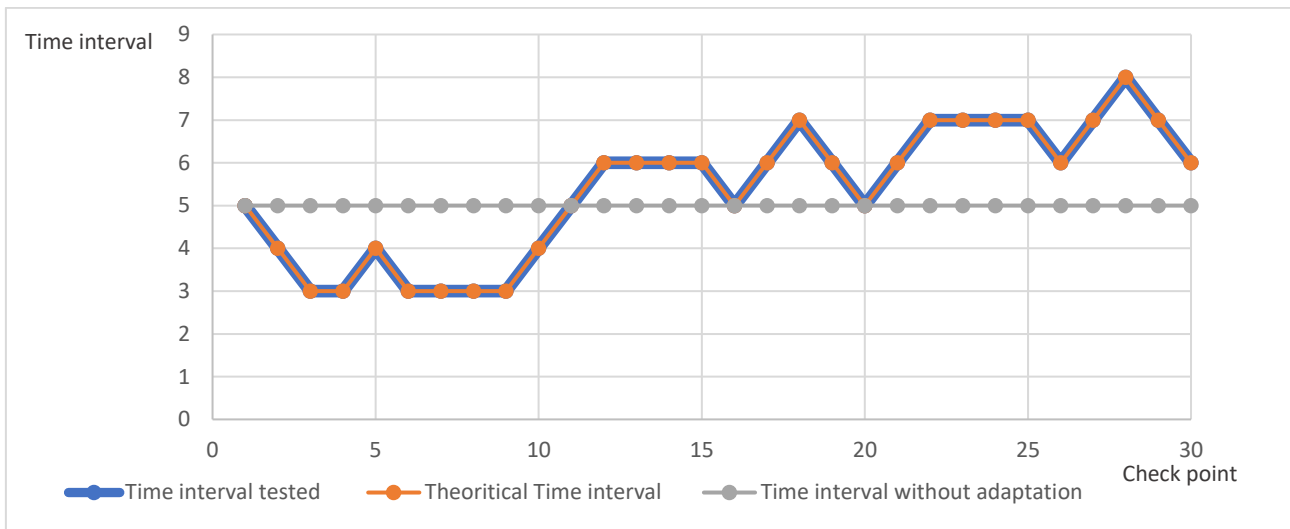


Figure 6. Time interval value every check point with applying and without applying adaptation.

We note that the time interval changes dynamically after each execution of the adaptation process, while it remains constant if it is not applied. We also note that the values we obtain in the test are the same as those calculated manually. Thus, we conclude that the function we used for adaptation is applied correctly in the game.

As we said, the information mentioned in the Table 3 below will be included for each experimenter in our system. Based on the information from the 15 subjects, we will create a comprehensive table, which will contain the sum of successful adjustments as well as the accuracy which is calculated in the following way:

$$\text{accuracy} = \frac{\text{correct adaptations}}{\text{total adaptations}} \times 100 \tag{8}$$

Table 3. Experimenter about system.

Subject	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Correct Adaptaion	60	60	60	60	60	60	60	60	60	60	60	60	60	60	60
Accuracy	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

The table shows that all the adjustments we made to each person given by the system exactly match what was calculated manually. The average number of adaptation operations for each person is 30, and the total number is 450. From these results, we conclude that the percentage of its total effectiveness is 100.

To find the performance of the proposed system, we will calculate its accuracy. Our system is divided into two parts: (1) Emotion recognition process (ERP) using DeepFace Framework. It will be applied at every check point (10 s). We found in the first test 57 correct results out of 60 tests. (2) Adaptation process (AP), which means that the overall system’s accuracy will be the average of the adaptation process accuracy and the DeepFace accuracy Framework, which represented 97.35% [16], as well as real-time facial emotion recognition (Rt-FER).

$$\text{Accuracy} = \frac{\text{FER accuracy} + \text{Fer}_{\text{RT}} \text{ accuracy} + \text{AP accuracy}}{3} \tag{9}$$

$$\text{Accuracy} = \frac{97.35 + 100 + 95}{3} = 97.45\% \quad (10)$$

The accuracy comes out to 97.45%. This means that our solution is doing a great job in adapting the exer-game according to the patient's emotions.

## 5. Conclusions

Developing a personalized exer-game is a very challenging problem because it requires a heavy effort to enhance the performance measures of motor rehabilitation. This field of adaptive serious games is gaining attention owing to its applications to motivate people during training. This paper presented a detailed global architecture based on emotion recognition. It is divided into three main phases: (1) computing and interpretation, (2) emotion recognition, (3) and adaptation. Furthermore, a case study using facial emotion recognition and web camera has been presented in this work. Finally, we have presented an experimental setup in order to validate the proposed solution. We also conducted two types of tests where we, first, tested the effectiveness of the system to track changes that occur in the patient and, second, tested 15 people to determine the effectiveness of adaptation based on facial emotions.

The results were very satisfying as we reached accuracy. This system is not suitable for the use of a virtual reality headset. It requires other techniques for emotion recognition (ECG signal, EMG signal, speech, body movement, etc.). Our system has not been tested in rehabilitation centers. These shortcomings provide opportunities for future work to evaluate the therapeutic side of the system by testing our application in real situations, introducing other methods of adaptation, and comparing them with the method presented in this paper. We plan to use other devices such as the VR headset, and other techniques for emotion recognition such as body movements, electrical body signals, and the history of the gestures made by the patient. Thus, our future work will be an extension of the global architecture presented in this paper.

**Author Contributions:** Conceptualization, Y.I.; methodology, Y.I.; software, Y.I.; formal analysis, S.O., S.B.; resources, Y.I., S.B., S.O.; writing—original draft preparation, Y.I., S.B.; writing—review and editing, Y.I., S.B., S.O., M.M.; supervision, S.O., S.B., A.K.; project administration, S.O., S.B., N.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** We would like to thank the IBISC Laboratory and the Ile de France region and CDTA research center for their funding support. This material is based upon work supported by the FEDER CESAAR-AVC project under Grant N° IF 001 1053.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Tanaka, K.; Parker, J.R.; Baradoy, G.; Sheehan, D.; Holash, J.R.; Katz, L. A Comparison of Exergaming Interfaces for use in Rehabilitation Programs and Research. *Loading* . . . **2012**, *6*.
2. Valenzuela, T.; Okubo, Y.; Woodbury, A.; Lord, S.R.; Delbaere, K. Adherence to Technology-Based Exercise Programs in Older Adults: A Systematic Review. *J. Geriatr. Phys. Ther.* **2018**, *41*, 49–61. [[CrossRef](#)] [[PubMed](#)]
3. Meekes, W.; Stanmore, E.K. Motivational Determinants of Exergame Participation for Older People in Assisted Living Facilities: Mixed-Methods Study. *J. Med. Internet Res.* **2017**, *19*, e238. [[CrossRef](#)] [[PubMed](#)]
4. Alhagbani, A.; Williams, A. Home-Based Exergames for Older Adults Balance and Falls Risk: A Systematic Review. *Phys. Occup. Ther. Geriatr.* **2021**, *39*, 1–17. [[CrossRef](#)]
5. Piech, J.; Czernicki, K. Virtual Reality Rehabilitation and Exergames—Physical and Psychological Impact on Fall Prevention Among the Elderly—A Literature Review. *Appl. Sci.* **2021**, *11*, 4098. [[CrossRef](#)]
6. Cugusi, L.; Prosperini, L.; Mura, G. Exergaming for Quality of Life in Persons Living with Chronic Diseases: A Systematic Review and Meta-Analysis. *PM R* **2020**, *13*, 756–780. [[CrossRef](#)]
7. Pacheco, T.B.F.; De Medeiros, C.S.P.; De Oliveira, V.H.B.; Vieira, E.; De Cavalcanti, F.A.C. Effectiveness of exergames for improving mobility and balance in older adults: A systematic review and meta-analysis. *Syst. Rev.* **2020**, *9*, 1–14. [[CrossRef](#)]
8. Yoo, S.; Kay, J. Vrun: Running-In-Place Virtual Reality Exergame. In Proceedings of the 28th Australian Conference on Computer-Human Interaction, Association for Computing Machinery, Launceston, Australia, 29 November–2 December 2016; Volume 29, pp. 562–566.

9. Mütterlein, J. The Three Pillars of Virtual Reality? Investigating the Roles of Immersion, Presence, And Interactivity. In Proceedings of the 51st Hawaii International Conference on System Sciences, Kauai, HI, USA, 3–6 January 2018.
10. Kappen, D.L.; Mirza-Babaei, P.; Nacke, L.E. Older Adults' Physical Activity and Exergames: A Systematic Review. *Int. J. Hum. Comput. Interact.* **2019**, *35*, 140–167. [[CrossRef](#)]
11. Skjæret, N.; Nawaz, A.; Morat, T.; Schoene, D.; Helbostad, J.L.; Vereijken, B. Exercise and Rehabilitation Delivered through Exergames in Older Adults: An Integrative Review of Technologies, Safety and Efficacy. *Int. J. Med. Inform.* **2016**, *85*, 1–16. [[CrossRef](#)]
12. Barry, G.; Galna, B.; Rochester, L. The Role of Exergaming in Parkinson's Disease Rehabilitation: A Systematic Review of the Evidence. *J. Neuroeng. Rehabil.* **2014**, *11*, 1–10. [[CrossRef](#)]
13. Reis, E.; Postolache, G.; Teixeira, L.; Arriaga, P.; Lima, M.L.; Postolache, O. Exergames for Motor Rehabilitation in Older Adults: An Umbrella Review. *Phys. Ther. Rev.* **2019**, *24*, 84–99. [[CrossRef](#)]
14. Chen, Y.; Zhang, Y.; Guo, Z.; Bao, D.; Zhou, J. Comparison Between the Effects of Exergame Intervention and Traditional Physical Training on Improving Balance and Fall Prevention in Healthy Older Adults: A Systematic Review and Meta-Analysis. *J. Neuroeng. Rehabil.* **2021**, *18*, 1–17. [[CrossRef](#)]
15. Gamboa, E.; Ruiz, C.; Trujillo, M. Improving Patient Motivation Towards Physical Rehabilitation Treatments with Playtherapy Exergame. *Stud. Health Technol. Inform.* **2018**, *249*, 140–147.
16. Streicher, A.; Smeddinck, J.D. Personalized and Adaptive Serious Games. In *Entertainment Computing and Serious Games*; Springer: Cham, Switzerland, 2016; Volume 9970, pp. 332–377.
17. Ferreira, B.; Menezes, P. An Adaptive Virtual Reality-Based Serious Game for Therapeutic Rehabilitation. *Int. J. Online Biomed. Eng.* **2020**, *16*, 63–71. [[CrossRef](#)]
18. Palaniappan, S.M.; Suresh, S.; Haddad, J.M.; Duerstock, B.S. Adaptive Virtual Reality Exergame for Individualized Rehabilitation for Persons with Spinal Cord Injury. In *Computer Vision—Eccv 2020 Workshops*; Bartoli, A., Fusiello, A., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 518–535.
19. Cristian, G.-P.; David, V.; Ana-Isabel, C.-S.; Rodríguez-Hernández Marta, L.M.-C.J.; Santiago, S.-S. A Platform Based on Personalized Exergames and Natural User Interfaces to Promote Remote Physical Activity and Improve Healthy Aging in Elderly People. *Sustainability* **2021**, *13*, 7578. [[CrossRef](#)]
20. Vallejo, D.; Gmez-Portes, C.; Albusac, J.; Glez-Morcillo, C.; Castro-Schez, J.J. Personalized Exergames Language: A Novel Approach to The Automatic Generation of Personalized Exergames for Stroke Patients. *Appl. Sci.* **2020**, *10*, 7378. [[CrossRef](#)]
21. Cristian, G.-P.; Lacave Carmen, I.M.A.; David, V. Home Rehabilitation Based on Gamification and Serious Games for Young People: A Systematic Mapping Study. *Appl. Sci.* **2020**, *10*, 8849. [[CrossRef](#)]
22. Afyouni, I.; Qamar, A.M.; Hussain, S.O.; Ur Rehman, F.; Sadiq, B.; Murad, A. Motion-Based Serious Games for Hand Assistive Rehabilitation. In Proceedings of the 22nd International Conference on Intelligent User Interfaces Companion, Limassol, Cyprus, 13–16 March 2017; pp. 133–137.
23. Varela-Aldás, J.; Palacios-Navarro, G.; Amariglio, R.; García-Magariño, I. Head-Mounted Display-Based Application for Cognitive Training. *Sensors* **2020**, *20*, 6552. [[CrossRef](#)]
24. Afyouni, I.; Murad, A.; Einea, A. Adaptive Rehabilitation Bots in Serious Games. *Sensors* **2020**, *20*, 7037. [[CrossRef](#)]
25. Guimarães, V.; Oliveira, E.; Carvalho, A.; Cardoso, N.; Emerich, J.; Dumoulin, C.; Swinnen, N.; De Jong, J.; de Bruin, E.D. An Exergame Solution for Personalized Multicomponent Training in Older Adults. *Appl. Sci.* **2021**, *11*, 7986. [[CrossRef](#)]
26. Paraskevopoulos, I.; Tsekelves, E. Use of Gaming Sensors and Customised Exergames for Parkinson's Disease Rehabilitation. In Proceedings of the 2013 5th International Conference on Games and Virtual Worlds for Serious Applications, Poole, UK, 11–13 September 2013; pp. 1–5.
27. Li, X.; Han, T.; Zhang, E.; Shao, W.; Li, L.; Wu, C. Memoride: An Exergame Combined with Working Memory Training to Motivate Elderly with Mild Cognitive Impairment to Actively Participate in Rehabilitation. In *Human Aspects of It for the Aged Population. Supporting Everyday Life Activities*; Gao, Q., Zhou, J., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 90–105.
28. Piper, W.E.; Ogradniczuk, J.S.; Joyce, A.S.; McCallum, M.; Rosie, J.S. Relationships Among Affect, Work, And Outcome in Group Therapy for Patients with Complicated Grief. *Am. J. Psychother.* **2002**, *56*, 347–361. [[CrossRef](#)]
29. Mergenthaler, E. Shifts from Negative to Positive Emotional Tone: Facilitators of Therapeutic Change. In Proceedings of the Presentation at the 34th Annual Meeting of The Society for Psychotherapy Research, Weimar, Germany, 25–29 June 2003.
30. Kritikos, J.; Caravas, P.; Tzannetos, G.; Douloudi, M.; Koutsouris, D. Emotional Stimulation During Motor Exercise: An Integration to The Holistic Rehabilitation Framework. In Proceedings of the 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, 23–27 July 2019. [[CrossRef](#)]
31. Ekman, P. Universal Facial Expressions of Emotion. *Calif. Ment. Health* **1970**, *8*, 151–158.
32. Russell, J.A.; Mehrabian, A. Evidence for A Three-Factor Theory of Emotions. *J. Res. Personal.* **1977**, *11*, 273–294. [[CrossRef](#)]
33. Toisoul, A.; Kossaiji, J.; Bulat, A.; Tzimiropoulos, G.; Pantic, M. Estimation of continuous valence and arousal levels from faces in naturalistic conditions. *Nat. Mach. Intell.* **2021**, *3*, 42–50. [[CrossRef](#)]
34. Bălan, O.; Moise, G.; Petrescu, L.; Moldoveanu, A.; Leordeanu, M.; Moldoveanu, F. Emotion Classification Based on Biophysical Signals and Machine Learning Techniques. *Symmetry* **2020**, *12*, 21. [[CrossRef](#)]

35. Hussain, S.; Alzoubi, O.; Calvo, R.; D'mello, S. Affect Detection from Multichannel Physiology During Learning Sessions with Autotutor. In *International Conference on Artificial Intelligence in Education*; Biswas, G., Bull, S., Kay, J., Mitrovic, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 131–138. [[CrossRef](#)]
36. Jin, X.; Wang, Z. An Emotion Space Model for Recognition of Emotions in Spoken Chinese. In *Affective Computing and Intelligent Interaction*; Tao, J., Tan, T., Picard, R.W., Eds.; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3784, pp. 397–402. [[CrossRef](#)]
37. Ekman, P.; Friesen, W.V. *Facial Action Coding System*; American Psychological Association: Washington, DC, USA, 1978.
38. Dzedzickis, A.; Kaklauskas, A.; Bucinskas, V. Human Emotion Recognition: Review of Sensors and Methods. *Sensors* **2020**, *20*, 592. [[CrossRef](#)]
39. Benrachou, D.E.; Masmoudi, M.; Djekoune, O.; Zenati, N.; Ousmer, M. Avatar-Facilitated Therapy and Virtual Reality: Next-Generation of Functional Rehabilitation Methods. In Proceedings of the 2020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP), El Oued, Algeria, 16–17 May 2020; pp. 298–304.
40. Trombetta, M.; Henrique, P.P.B.; Brum, M.R.; Colussi, E.L.; De Marchi, A.C.B.; Rieder, R. Motion Rehab Ave 3d: A Vr-Based Exergame for Post-Stroke Rehabilitation. *Comput. Methods Programs Biomed.* **2017**, *151*, 15–20. [[CrossRef](#)]
41. Miclaus, R.; Roman, N.; Caloian, S.; Mitoiu, B.; Suciuc, O.; Onofrei, R.R.; Pavel, E.; Neculau, A. Non-Immersive Virtual Reality for Post-Stroke Upper Extremity Rehabilitation: A Small Cohort Randomized Trial. *Brain Sci.* **2020**, *10*, 655. [[CrossRef](#)]
42. Kim, W.-S.; Cho, S.; Ku, J.; Kim, Y.; Lee, K.; Hwang, H.-J.; Paik, N.-J. Clinical Application of Virtual Reality for Upper Limb Motor Rehabilitation in Stroke: Review of Technologies and Clinical Evidence. *J. Clin. Med.* **2020**, *9*, 3369. [[CrossRef](#)]
43. Bachmann, D.; Weichert, F.; Rinkenauer, G. Review of Three-Dimensional Human-Computer Interaction with Focus on the Leap Motion Controller. *Sensors* **2018**, *18*, 2194. [[CrossRef](#)]
44. Pereira, M.F.; Prahm, C.; Kolbenschlag, J.; Oliveira, E.; Rodrigues, N.F. A Virtual Reality Serious Game for Hand Rehabilitation Therapy. In Proceedings of the 2020 IEEE 8th International Conference on Serious Games and Applications for Health (SeGAH), Vancouver, BC, Canada, 12–14 August 2020; pp. 1–7.
45. Nadia, N.; Sergio, O.-E.; Miguel, C. An Semg-Controlled 3d Game for Rehabilitation Therapies: Real-Time Time Hand Gesture. *Sensors* **2020**, *20*, 6451.
46. Dharia, S.; Eirinaki, M.; Jain, V.; Patel, J.; Varlamis, I.; Vora, J.; Yamauchi, R. Social Recommendations for Personalized Fitness Assistance. *Pers. Ubiquitous Comput.* **2018**, *22*, 245–257. [[CrossRef](#)]
47. Hagen, K.; Chorianopoulos, K.; Wang, A.I.; Jaccheri, L.; Weie, S. Gameplay as Exercise. In Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, San Jose, CA, USA, 7–12 May 2016; pp. 1872–1878.
48. Bernava, G.; Nucita, A.; Iannizzotto, G.; Capri, T.; Fabio, R.A. Proteo: A Framework for Serious Games in Tele-Rehabilitation. *Appl. Sci.* **2021**, *11*, 5935. [[CrossRef](#)]
49. González-González, C.S.; Toledo-Delgado, P.A.; Muñoz-Cruz, V.; Torres-Carrion, P.V. Serious Games for Rehabilitation: Gestural Interaction in Personalized Gamified Exercises Through a Recommender System. *J. Biomed. Inform.* **2019**, *97*, 103266. [[CrossRef](#)] [[PubMed](#)]
50. Zhao, Z.; Arya, A.; Orji, R.; Chan, G. Effects of A Personalized Fitness Recommender System Using Gamification and Continuous Player Modeling: System Design and Long-Term Validation Study. *JMIR Serious Games* **2020**, *8*, E19968. [[CrossRef](#)]
51. Tadayon, R.; Amresh, A.; Mcdaniel, T.; Panchanathan, S. Real-Time Stealth Intervention for Motor Learning Using Player Flow-State. In Proceedings of the 2018 IEEE 6th International Conference on Serious Games and Applications for Health (SeGAH), Vienna, Austria, 16–18 May 2018; pp. 1–8.
52. Verma, V.; Rheem, H.; Amresh, A.; Craig, S.D.; Bansal, A. Predicting Real-Time Affective States by Modeling Facial Emotions Captured During Educational Video Game Play. In Proceedings of the GALA 2020: Games and Learning Alliance, Laval, France, 9–10 December 2020; Springer: Cham, Switzerland; pp. 447–452.
53. Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. Deepface: Closing the Gap to Human-Level Performance in Face Verification. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1701–1708.
54. Mood Me. Available online: <https://www.Mood-Me.Com/Products/Unity-Face-Ai-Ar-Sdk/> (accessed on 10 November 2021).
55. Unity-Technologies. Available online: <https://Github.Com/Unity-Technologies/Barracuda-Release> (accessed on 10 November 2021).
56. Ultraleap. Available online: <https://www.Ultraleap.Com/Product/Leap-Motion-Controller/> (accessed on 10 November 2021).