Approche neuronale profonde pour la reconnaissance conjointe audio-vidéo de violences dans un environnement ferroviaire embarqué

Thèse de doctorat de l'Université de Lille

École doctorale n° 632 : SCIENCE DE L'INGÉNIÉRIE ET DES SYSTÈMES (ENGYS)
Spécialité de doctorat : Micro-nanosystèmes et capteurs
COSYS-LEOST, Université Gustave Eiffel

Thèse présentée et soutenue à l'Université Gustave Eiffel, le 25/09/2023, par :

Tony MARTEAU

Composition du Jury

François BRÉMOND Directeur de recherche, INRIA	Président du jury
Catherine ACHARD Professeure des universités, Sorbonne Université	Rapportrice
Olivier LÉZORAY Professeur des universités, Université de Caen	Rapporteur
Claire NICODEME Docteure, Direction Technologies, Innovation et Projets Groupe - SNCF	Invité

Encadrement de la thèse

Fouzia BOUKOUR Directrice de recherche, Université Gustave Eiffel	Directrice de thèse
David SODOYER Chargé de recherche, Université Gustave Eiffel	Co-Encadrant de thèse
Sébastien AMBELLOUIS Ingénieur de recherche, Université Gustave Eiffel	Co-Encadrant de thèse
Sitou AFANOU Ingénieur, Centre D'ingénierie Du Matériel SNCF Voyageurs	Tuteur en entreprise

Remerciements

Cette thèse CIFRE représente un chapitre essentiel de mon parcours. Ce fut une période intense, riche en rencontres inspirantes et en défis stimulants qui ont grandement contribué au développement de mes compétences, tant sur le plan professionnel que personnel.

Je saisis cette opportunité pour exprimer ma profonde gratitude envers les acteurs principaux de cette aventure. Je crains d'omettre certains noms et m'en excuse par avance.

Tout d'abord, je tiens à remercier les personnes qui m'ont fait confiance et qui ont encadré mes travaux, que ce soit David SODOYER et Sébastien AMBELLOUIS, de l'Université Gustave Eiffel, ainsi que Sitou AFANOU de la SNCF. Qu'ils soient remerciés pour leurs bienveillances et leurs disponibilités. Ils m'ont fait confiance, j'espère avoir été à la hauteur de leurs attentes. Je tiens également à remercier chaleureusement Fouzia Boukour pour avoir accepté de diriger cette thèse.

J'adresse aussi tous mes remerciements à Monsieur François BRÉMOND, Directeur de recherche à l'INRIA, ainsi qu'à Madame Catherine ACHARD, Professeure des universités à Sorbonne Université, et Monsieur Olivier LÉZORAY, Professeur des universités à Université de Caen, de l'honneur qu'ils m'ont fait en acceptant d'être respectivement président du jury et rapporteurs de cette thèse. De même que Claire NICODEME pour sa présence dans mon jury.

Enfin, je tiens à exprimer ma gratitude envers toutes les personnes rencontrées au fil de ces trois années, celles qui ont contribué à l'avancement de mes travaux et celles avec qui j'ai échangé des idées enrichissantes. Une mention spéciale revient aux collègues de l'équipe ETF1 de la SNCF, Philippe, Cédrick et Gwenael, pour les moments partagés.

Ce document marque la fin d'un chapitre significatif de ma vie, mais il représente également le tremplin vers de nouveaux horizons et projets.

Encore merci.

Résumé

Faisant face à l'augmentation du nombre de coups et blessures volontaires recensés dans les rames ferroviaires depuis plusieurs années, la SNCF installe des systèmes de vidéo surveillance à l'intérieur de ses rames. Malheureusement, ces systèmes de vidéo surveillance font face à plusieurs difficultés : toutes les images ne sont pas transmises au sol pour être supervisées, la perception est contrainte par des scènes hors-champ, de nombreuses occultations, des phénomènes de reflet, de flou ou de vibration, et enfin la quantité croissante de flux vidéo devient complexe à superviser efficacement par des opérateurs. Cette thèse s'inscrit dans la problématique de reconnaissance automatique d'activités humaines et aborde spécifiquement le problème de la reconnaissance de situations violentes dans un environnement ferroviaire embarqué. L'objectif est d'ajouter le traitement du signal audio à celui de la vidéo afin de bénéficier des complémentarités et/ou cohérences de ces deux perceptions. Pour cela, ces travaux ont consisté à étudier le traitement conjoint de signaux audio et vidéo par des architectures neuronales profondes. Les architectures mises en place s'appuient sur des extracteurs de caractéristiques proposés dans la communauté tel que I3D pour le signal vidéo et OpenL3 pour le signal audio. La structure temporelle des caractéristiques extraites de chaque signal est ensuite modélisée à l'aide de couches récurrentes (LSTM). Enfin, la reconnaissance de violence est obtenue en combinant la sortie des LSTM à différents niveaux de fusion et avec différentes fonctions (concaténation, mécanisme à porte et attention). Pour mettre en œuvre ces architectures, un jeu de données a été enregistré à bord d'une rame ferroviaire en dynamique. Ce jeu de données est composé de scénarios d'agression joués par des comédiens professionnels dans de multiples contextes (densités de figurants, lieux et positions dans la rame etc.). L'annotation de ce jeu de données a été produite en dissociant complètement les modes audio et vidéo afin de tenir compte de la spécificité de la perception modale des violences. L'évaluation de ces modèles audio et/ou vidéo de violences est présentée en fonction des différentes architectures neuronales proposées et l'analyse des performances est réalisée en fonction de différents degrés de violence, de la durée de perception, différents degrés d'occultation et de la distance aux capteurs.

Abstract

Faced with an increase in the number of intentional assaults and injuries recorded on railway trains over the past few years, the SNCF has been installing video surveillance systems inside its trains. Unfortunately, these video surveillance systems face several difficulties : not all images are transmitted to the ground for supervision, perception is constrained by off-screen scenes, numerous occultations, reflection, blurring or vibration phenomena, and finally the increasing quantity of video streams is becoming complex for operators to supervise efficiently. This thesis is part of the problem of automatic recognition of human activities, and deals specifically with the problem of recognizing violent situations in an on-board railway environment. The aim is to add audio signal processing to video signal processing in order to benefit from the complementarities and/or coherences of these two perceptions. To this end, we have studied the joint processing of audio and video signals using deep neural architectures. The architectures implemented are based on feature extractors available in the community, such as I3D for the video signal and OpenL3 for the audio signal. The temporal structure of the features extracted from each signal is then modeled using recurrent layers (LSTMs). Finally, violence recognition is achieved by combining the output of the LSTMs at different fusion levels and with different functions (concatenation, gating and attention). To implement these architectures, a dataset was recorded on board a dynamic train. This dataset is composed

of aggression scenarios played out by professional actors in multiple contexts (density of extras, locations and positions in the train, etc.). The annotation of this dataset was produced by completely dissociating audio and video modes in order to take into account the specificity of modal perception of violence. The evaluation of these audio and/or video models of violence is presented in terms of the different neural architectures proposed, and performance analysis is carried out as a function of different degrees of violence, duration of perception, different degrees of occlusion and distance from the sensors.

Table des matières

Table des figures

Liste des tableaux

Liste des acronymes

AE Autoencoder Network. 25, 30

CNN Convolutionnal Neural Network. 9, 18, 53

FFT Fast Fourier Transformation. 31, 32

GAN Generative Adversarial Neural Network. 25, 30

GMM Gaussian Mixture Models. 17, 21, 25, 43, 51, 61, 62, 65

GRU Gated Recurrent Unit. 33, 37, 44, 53

HMM Hidden Markov Models. 25, 35, 43, 51

IA Intelligence Artificielle. 17, 21, 22, 28, 125

LSTM Long Short Term Memory. 9, 33, 37, 44, 45, 47, 50, 52, 53, 58, 60, 79, 81, 83, 93, 95–97, 103, 127

MFCC Mel-frequency cepstral coefficients. 31, 51, 52, 62, 65

MoSIFT Motion Scale-Invariant Feature Transform. 43

RGPD Règlement Général sur la Protection des Données. 22, 23, 25–27, 128

RNN Recurrent Neural Network. 33, 44, 52

RSB Rapport signal sur bruit. 62

RVB Rouge, Vert, Bleu. 30, 34, 48, 49, 54, 56, 57, 60, 69, 92, 94, 104

SNCF Société Nationale des Chemins de fer Français. 9, 17, 20–22, 67, 68, 72

STIP Space-Time Interest Points. 43

SVM Support Vector Machine. 17, 21, 25, 35, 43–45, 56, 57, 60–62, 65

TSN Temporal Segment Network. 9, 45, 47, 57

Publications

International Conferences

1. Tony MARTEAU, Sitou AFANOU, David SODOYER et Sébastien AMBELLOUIS, "Violence detection in railway environment with modern deep learning approaches and small dataset", Lisbonne, Portugal, 2022, Transport Research Arena (TRA)
2. Tony MARTEAU, David SODOYER, Sébastien AMBELLOUIS et Sitou AFANOU, "Level fusion analysis of recurrent audio and video neural network for violence detection in railway", Belgrade, Serbie, 2022, IEEE European Signal Processing Conf. (EUSIPCO), p. 563-567
3. Tony MARTEAU, Sitou AFANOU, David SODOYER, Sébastien AMBELLOUIS et Fouzia BOUKOUR, "Audio Events Detection in Noisy Embedded Railway Environments", Munich, Allemagne (Virtual), 2020, Springer European Dependable Computing Conf. (EDCC) - Workshops, p. 20-32.

Introduction

Depuis plusieurs années, la population française est encouragée à prendre les transports en commun afin de réduire l'empreinte carbone des déplacements par exemple. L'exploitation accrue des réseaux de transports en commun oblige les exploitants à faire face à de nombreux défis sur la régularité, la sécurité — événements accidentels — mais aussi la sûreté — actes de malveillance — pour les passagers comme pour les agents. Concernant ce dernier défi, pour accroître la sûreté dans ces trains la Société Nationale des Chemins de fer Français (SNCF) s'intéresse à la reconnaissance automatique des actes violents avec des modèles d'Intelligence Artificielle (IA) exploitant les flux audio et vidéo des systèmes de vidéo-protection.

Avant de définir le contexte industriel et scientifique de cette tâche, les problématiques que nous abordons et la méthodologie utilisée nous allons familiariser le lecteur avec le vocabulaire ferroviaire et le vocabulaire du domaine de l'IA.

Communément appelé train par le grand public ce terme regroupe différents types de matériel : le Train à Grande Vitesse connu sous l'acronyme TGV qui est exploité entre les grandes villes françaises pour des liaisons rapides, l'Intercité exploité entre les grandes villes françaises pour des liaisons lentes, le TER exploité entre les villes d'une même région, le Transilien exploité pour connecter la banlieue parisienne à Paris et le train de fret exploité pour le transport de marchandises. Ces trains sont composés d'une ou plusieurs rames attelées entre elles. Une rame est un ensemble de voitures à simple ou à double niveau qui contiennent des salles avec des aménagements différents (salle voyageur ou plateforme). Les rames peuvent être automotrices lorsqu'elles assurent elles-mêmes la propulsion ou tractées par une locomotive.

L'intelligence artificielle, communément appelée IA par le grand public, regroupe différentes approches (Figure 2) : les approches symboliques qui reposent sur des règles de décision définies et écrites par des humains en fonction de leur expérience du domaine d'application, et les approches par apprentissage automatique qui définissent un modèle de décision en intégrant de manière automatique des connaissances acquises au cours de nombreuses expériences. Ces deux approches de l'IA se sont développées parallèlement durant les premières années (1952-1969). Cependant, pour des problématiques de puissance de calcul, de disponibilité de données et de technologie jusque dans les années 80, le développement des systèmes à base de connaissances, plus simples à mettre en place, a connu une croissance plus rapide que les approches par apprentissage automatique. Avec l'émergence de calculateurs de plus en plus puissants, de plus en plus abordables et avec la numérisation de la société, les approches d'apprentissage automatique se sont développées. De nouveaux algorithmes tels que celui des K plus proches voisins, celui de l'apprentissage d'un mélange de gaussiennes (Gaussian Mixture Models (GMM)), d'une machine à vecteurs supports (Support Vector Machine (SVM)), l'algorithme des forêts aléatoires (random forest) ou celui du gradient boosting à la base de l'apprentissage d'un ensemble de modèles faiblement appris. Ces techniques permettent d'éviter l'expression explicite de règles par l'humain et de définir des modèles de décision plus complexes. Afin de réduire la complexité de ces apprentissages et face à des données généralement de grandes dimensions, des méthodes de réduction de dimension sont préalablement appli-

FIGURE 2 – Deux approches de l'intelligence artificielle.

quées rendant ainsi l'estimation du modèle plus simple et plus rapide. Parmi ces méthodes nous pouvons citer l'extraction de caractéristiques, l'analyse en composantes principales (ACP) ou l'analyse discriminante linéaire (ADL). C'est avec la croissance exponentielle de la puissance de calcul des processeurs et l'avènement des composants graphiques que nous assistons depuis la fin des années 90 au développement très rapide des techniques neuronales profondes [60].

Aujourd'hui, la communauté scientifique cherche à traiter les données en se passant de l'étape de réduction de la dimension décrite précédemment. Sur la base des recherches menées par les biologistes et des médecins sur la modélisation du cerveau humain et des neurones qui le composent et avec l'aide des informaticiens, nous sommes passés des réseaux de neurones totalement connectés (FCN) composés de quelques couches cachées à des réseaux de neurones convolutifs (Convolutionnal Neural Network (CNN)) beaucoup plus profonds. Dans les CNN, les couches originelles dont les neurones étaient totalement connectés à ceux des couches précédentes, se sont transformées en couches de convolutions pour lesquelles le nombre de paramètres à apprendre a été considérablement réduit. Une couche d'un CNN n'est d'ailleurs plus caractérisée par un nombre de neurones mais par un nombre de filtres eux-mêmes caractérisés par une taille de noyau. En quelques années, ces couches se sont complexifiées en intégrant par exemple certaines formes de récurrences, capables de prendre en compte la temporalité présente dans les signaux. Ces architectures neuronales associées à des techniques d'apprentissage profond ont été appliquées à de nombreux signaux (audio, image, séquence d'images, série temporelle etc.) dans divers contextes d'application tels que la robotique, la surveillance intelligente, la médecine par exemple. Les limites en termes de performance ont été repoussées et ces architectures sont aujourd'hui la référence dans l'état de l'art.

Concernant le contexte industriel du transport, la sûreté est une problématique omniprésente. Depuis plusieurs années, l'Institut Paris Régions [4] ou Interstats [5] éditent des enquêtes et des rapports sur le sujet de la violence dans les transports en commun, dont l'objectif est de mesurer l'évolution du nombre d'actes violents et le sentiment d'insécurité que les usagers peuvent ressentir dans les transports en commun (rames et gares) de France. L'évolution du nombre d'actes violents est calculé sur la base de données des infractions recensées par les forces de sécurité (services de police et unités de gendarmerie). Le sentiment d'insécurité quant à lui, est mesuré avec des enquêtes de terrain.

4. L'institut Paris Régions, fondé en 1960, est une agence régionale qui réalise des études et des travaux pluridisciplinaires sur les thématiques d'urbanisme, des transports, de la mobilité, de l'environnement, de l'économie et des questions de société pour aider la région Île-de-France et ses partenaires à prendre des décisions sur les politiques d'aménagement.

5. Interstats, fondé en 2014, est le service statistique ministériel de la sécurité intérieure qui réalise des études sur la délinquance et la criminalité.

6. Source : Interstats - rapport "Les vols et violences enregistrés dans les réseaux de transports en commun en 2021"

	France					Île-de-France				
	Vols sans violence	Vols Violents	Coups et blessures volontaires	Violences sexuelles	Outrages et violences contre dépositaires de l'autorité publique	Vols sans violence	Vols violents	Coups et blessures volontaires	Violences sexuelles	Outrages et violences contre dépositaires de l'autorité publique
2021	96 653	10 727	7 104	2 039	5 645	61 091	8 060	3 105	905	2 345
2020	93 783	11 268	6 183	1 548	5 249	63 471	8 644	2 615	689	2 128
2019	133 023	11 767	8 205	2 128	5 333	90 287	8 951	3 520	984	2 251
2018	114 736	11 797	7 848	1 889	4 688	73 174	8 519	3 212	869	2 069
2017	110 616	12 584	7 838	1 375	4 918	70 323	9 559	3 369	657	2 295
2016	108 430	12 184	7 347	1 271	4 918	71 840	8 778	3 200	676	2 273
2020/2021	3%	-5%	15%	32%	8%	-4%	-7%	19%	31%	10%
2019/2020	-29%	-4%	-25%	-27%	-2%	-30%	-3%	-26%	-30%	-5%
2018/2019	16%	-0%	5%	13%	14%	23%	5%	10%	13%	9%
2017/2018	4%	-6%	0%	37%	-5%	4%	-11%	-5%	32%	-10%
2016/2017	2%	3%	7%	8%	0%	-2%	9%	5%	-3%	1%

TABLE 1 – Nombre de victimes de vols et de violences dans les transports en commun en France et Île de France entre 2016 et 2021 [6].

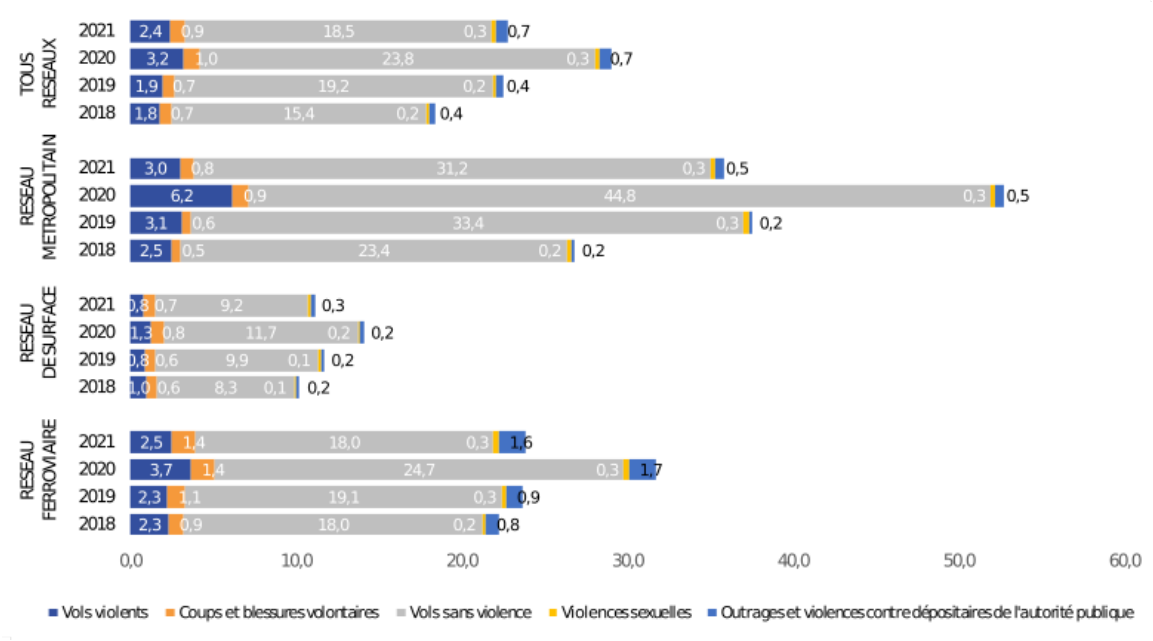


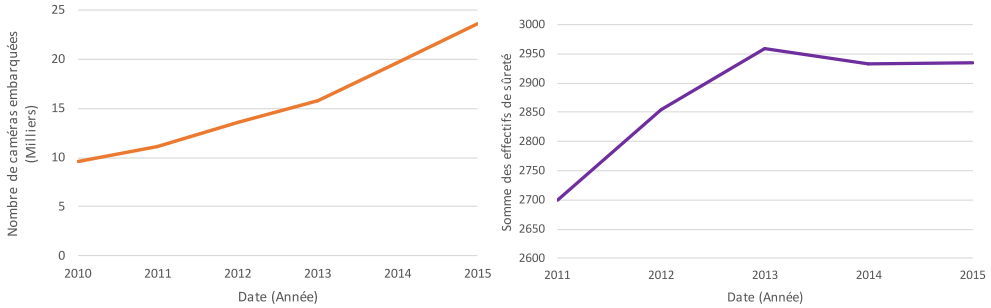
FIGURE 3 – Nombre de victimes de vols et de violences dans les transports en commun pour un million de voyages en Île-de-France entre 2019 et 2021 [7].

Le tableau 1 extrait du rapport d'Interstats de septembre 2022 [20] consolide les données de 2016 à 2021 pour la France et l'Île-de-France. Tout d'abord, on peut observer que les deux-tiers des victimes enregistrées dans les transports en commun ont lieu en Île-de-France. Une explication avancée dans les rapports est l'utilisation plus soutenue des transports en commun pour les trajets domicile-travail des Franciliens ainsi que par les nombreux touristes. Il apparaît clairement que le nombre de violences a augmenté sur la période 2016-2021 si on exclut la période sanitaire de 2020 et de 2021. Finalement, avec la reprise de l'activité économique, le nombre de violences est de retour à son niveau d'avant crise.

La figure 3 présente ces chiffres par million de voyages et en fonction du type de réseaux de transport en commun utilisé. Il apparaît que depuis 2019 le nombre de victimes de vols ou de violences par million de voyages n'évolue que très peu et cela quel que soit le réseau emprunté. Toutefois, ces chiffres sont à mettre en rapport avec un nombre de voyages toujours croissant d'une année à l'autre. Par ailleurs, il faut noter que le nombre absolu de victimes reste le même et cela même si le nombre de voyages diminue (année 2020 : crise COVID).

Année	Niveau d'insécurité
2001	43,8%
2003	43,7%
2005	45,2%
2007	42,2%
2009	40,6%
2011	45,5%
2013	43,7%
2015	42,3%
2017	38,1%
2019	40,9%

TABLE 2 – Niveau d'insécurité resenti par les passagers des transports en commun d'Île-de-France.



(a) Nombre de caméras embarquées[8]. (b) Effectif de sûreté[9].

FIGURE 4 – Évolution du nombre de caméras embarquées et du nombre d'agents à la SNCF entre 2010 et 2015.

Le sentiment d'insécurité est également un indicateur important de la qualité du service d'un transport collectif emprunté par des millions de voyageurs. Cet indicateur est mesuré tous les deux ans. Le tableau 2 présente sa valeur depuis 2001. On observe que depuis 2001, le niveau d'insécurité ressenti par les passagers évolue peu mais est élevé au-dessus de 40%. Quatre personnes sur dix craignent donc une agression ou un vol pendant leur voyage : les raisons évoquées sont la vétusté de nombreux sites, la densité très élevée de certaines lignes, l'absence de personnel notamment dans les rames et la présence de personnes droguées ou alcoolisées.

Dans ce contexte où le nombre de victimes de violences et le sentiment d'insécurité restent élevés, les exploitants de réseaux de transport doivent mettre en place des politiques pour améliorer ces indicateurs. Au sein de la Société Nationale des Chemins de fer Français (SNCF), deux principales stratégies sont mises en place : En premier lieu, l'augmentation du nombre de caméras embarquées (Figure 4a) et, en second lieu, l'augmentation du nombre d'agents de sûreté (Figure 4b).

Ces deux solutions sont complémentaires et ont pour principaux objectifs d'être dissuasifs, d'apporter des preuves lors des dépôts de plaintes et de rassurer les passagers et les agents. En région parisienne, les gares regroupent à elles seules environ 10000 caméras dont la majorité des flux ne sont pas visualisables depuis des postes de commande (PC). Lorsque les caméras peuvent être supervisées depuis ces PC, les éventuels événe-

7. Source : Interstats - rapport "Les vols et violences enregistrés dans les réseaux de transports en commun en 2021"

8. https://ressources.data.sncf.com/explore/dataset/equipement-surete

9. https://ressources.data.sncf.com/explore/dataset/effectif-agents-surete-ferroviaire

ments d'incivilité ou d'agression sont détectés par les agents en poste après un effort de concentration important, un temps devant l'écran très élevé et le niveau de fatigue qui en découle. Depuis quelques années, l'installation de matériels de vidéo-protection à bord des véhicules est systématique, augmentant ainsi le nombre de flux vidéo potentiel à transmettre au PC et à analyser par les opérateurs, rendant cette tâche non seulement non fiable mais humainement inacceptable.

Depuis de nombreuses années, la SNCF cherche à profiter des avancées en matière d'apprentissage machine pour spécifier et développer des outils d'interprétation automatique des images afin d'épauler les agents dans cette pénible tâche d'analyse des flux vidéo. Cette vidéo intelligente regroupe aujourd'hui, des algorithmes d'IA qui montrent chaque jour des performances remarquables. Toutefois, si de nombreux travaux ont été réalisés et publiés dans l'état de l'art, aucune des solutions proposées par les différentes équipes n'est aujourd'hui commercialisée. Les problématiques auxquelles sont confrontées ces recherches sont dans un premier temps, de définir la meilleure architecture d'IA capable d'extraire l'information pertinente pour modéliser les événements d'intérêt tout en garantissant, dans un second temps, qu'il est possible de traiter cet énorme volume de données dans un délai suffisamment court pour assurer le déclenchement rapide de l'alerte attendue.

Cependant, dans les véhicules ferroviaires, un système de vidéo-protection doit faire face à plusieurs difficultés :

— Tous les flux ne sont pas transmis au sol pour être visualisés, car la bande passante des connexions bord-sol est trop faible ;

— Le contenu des flux vidéo est soumis à des contraintes de perception liées à l'environnement particulier que représente l'enceinte d'une voiture : couverture insuffisante, occultation, reflet, flou, etc. ;

— Et, les flux audio ne sont pas visualisés par un agent.

Cette thèse s'inscrit dans le domaine de la reconnaissance de l'activité humaine et plus particulièrement dans le cadre de la reconnaissance d'actions violentes. C'est un domaine de recherche très actif dans la communauté. Le développement des approches par apprentissage automatique, particulièrement les techniques par apprentissage profond, et le développement des jeux de données ont permis d'atteindre des résultats intéressants ces dernières années. Aujourd'hui, les techniques développées exploitent principalement une ou plusieurs images provenant d'une séquence d'images couleurs, et/ou l'information de mouvements apparents contenue dans cette séquence. Les méthodes d'IA utilisées peuvent être séparées en deux groupes : celui des techniques d'apprentissage machine traditionnelles citées précédemment (SVM, GMM, HMM, Bag of Words etc.) et celui des réseaux profonds (CNN, 3D CNN, LSTM, ConvLSTM, etc.). Certains chercheurs se sont aussi intéressés à la détection d'actions violentes par l'analyse d'un ou plusieurs signaux audio, par la combinaison de plusieurs signaux hétérogènes tels que le signal vidéo et le signal audio pour profiter de la complémentarité de ces deux modes de perception. Même si l'analyse de l'activité humaine par analyse des images est un sujet courant dans la littérature, la reconnaissance d'actions violentes l'est un peu moins. Ce sujet a fait l'objet de quelques travaux depuis les années 90 en évoluant en fonction du champ d'étude de l'IA. L'état de l'art présente clairement que les méthodes profondes constituent aujourd'hui les solutions les plus performantes.

Ces méthodes ne peuvent cependant pas être exactement transférées à cause des contraintes géométriques, acoustiques et d'exploitation portées par le contexte "embarqué" des transports en commun (bus, métro, trains) qu'il faut pendre en compte pour assurer de manière fiable et robuste la reconnaissance d'actions violentes. La première de ces spécificités est qu'une telle enceinte constitue une structure longitudinale peu large

avec une faible hauteur, comportant de larges fenêtres et pouvant accueillir une densité de personnes très élevée. Du point de vue de la perception vidéo, cela implique un champ de vision plus profond que large, de nombreuses situations d'occultation, des reflets sur les vitres et les parois internes. D'un point de vue audio l'aspect longitudinal implique une dynamique sonore étendue, le rapport signal à bruit évoluant en fonction de la distance entre la source sonore et le microphone et d'un environnement fortement réfléchissant. Par ailleurs, l'environnement sonore est fortement variable à cause notamment de la variation de densité de personnes, passant d'un état plutôt silencieux à des états plus ou moins bruyants. La seconde spécificité de l'aspect embarqué est liée à la mobilité de cet environnement, impliquant dans premier temps une forte non-stationnarité d'un point de vue vidéo avec des variations accrues de la luminosité (transitions des passages en tunnels, en gares, variations rapides en zones urbaines) mais également avec des apparitions de mouvements plus ou moins rapides au travers des vitres de l'habitacle dus au défilement du paysage extérieur, variant en fonction de la vitesse de déplacement. La mobilité a également un impact sur la variabilité sonore de l'environnement notamment lorsque les portes sont ouvertes (bruit du quai) ou lorsque les fenêtres sont ouvertes et/ou que le train est en mouvement à vitesse élevée.

Dans ce contexte très particulier, les signaux sont donc sujets à des phénomènes non stationnaires et fortement aléatoires. La reconnaissance de violences dans un tel environnement transport est donc un grand challenge et les travaux l'ayant étudié sont assez peu nombreux. Il en résulte donc que peu de jeux de données et quasiment aucune base ne sont partagées. La reconnaissance d'une action violente dans un environnement ferroviaire est explorée depuis quelques années avant même l'essor des réseaux de neurones profonds [182, 125, 94, 188], avec dans la continuité de ces travaux [87] exploitant signal audio seul et réseaux de neurones profonds.

De plus, si l'intérêt pour cette problématique est évidente, il s'avère également très difficile d'acquérir des données les plus représentatives qu'il soit à cause de : la disponibilité des matériels, la possibilité d'équiper les matériels de caméras et de microphones, la nécessité de respecter le Règlement Général sur la Protection des Données (RGPD) mais surtout de rencontrer (fort heureusement) des événements violents en cours d'exploitation commerciale. Cette dernière raison font d'une action violente un événement rare.

Face à l'activité scientifique du domaine de recherche de l'IA, la marque Transilien de la SNCF a sollicité le Centre d'Ingénierie du Matériel pour réaliser une étude sur la reconnaissance d'actions violentes à l'intérieur de ses rames. L'objectif pour l'entreprise est d'avoir la capacité de reconnaître en temps-réel les situations de violence dans un environnement ferroviaire embarqué en utilisant la complémentarité des signaux audio et vidéo provenant du système de vidéo-protection embarqué afin de faire intervenir rapidement la police ferroviaire.

Notre objectif est de proposer une architecture neuronale profonde capable d'analyser conjointement les signaux audio et vidéo afin de reconnaître des actions violentes en prenant en compte les contraintes liées à l'environnement ferroviaire présentées par les points suivants :

- **Variation des échelles** : les capteurs étant installés aux deux extrémités des salles, la tâche de détection doit tenir compte de la variation de l'échelle (énergie et résolution) des scènes de violences perçues par la caméra ou le microphone ;
- **Les occultations** : l'aménagement des rames ferroviaires limitant la hauteur des salles et les capteurs étant installés proche de leur axe, de nombreuses occultations apparaissent dues aux passagers eux-mêmes ou au mobilier ;
- **Champ de vision limité** : sauf à utiliser des objectifs très grand angle, les dimensions des salles ne permettent pas aux caméras installées de capturer des images de tous les endroits de la salle ;

- **Bruit dans les signaux** : de nombreux bruits viennent s'ajouter au signal audio utile notamment ceux liés au mouvement du véhicule. De la même manière, le flou de bougé, l'effet des vibrations, des changements de luminosité etc. viennent altérer le contenu des images ;
- **Le manque de données et déséquilibre des classes** : les bases d'images et de sons acquis en environnement ferroviaire sont rares et difficiles à obtenir à bord d'un train en mission commerciale avec des voyageurs. Et lorsqu'elles existent, nous sommes confrontés à un déséquilibre du nombre d'échantillons entre "action sans violence" et "action avec violence" qui sont des événements plus rares.

Dans ce travail, pour résoudre la tâche de reconnaissance d'une action violente, sont proposées des architectures neuronales profondes uni-modales et multi-modales acceptant en entrée des échantillons audio et/ou des images en couleur. Ces architectures se basent sur des extracteurs de caractéristiques de la littérature entraînés sur de grands jeux de données uni-modaux qui n'ont pas été enregistrés dans un contexte ferroviaire. Les architectures proposées cherchent à modéliser la cohérence temporelle des données placées en entrée afin de finalement acter sur la présence ou non d'un acte violent. Chaque architecture uni-modale a été évaluée et plusieurs niveaux de combinaisons multi-modales ont été étudiés et comparés.

Comme nous le précisions auparavant, face à l'absence d'une base de données audio/vidéo spécifiquement ferroviaire, pour mener à bien cette étude, nous avons donc fait le choix de constituer notre propre jeu de données à partir de scènes jouées par des comédiens dans un véhicule circulant sur une ligne commerciale. Tous les événements violents ont été annotés afin de réaliser l'apprentissage des poids et des biais des architectures de façon supervisée sur une partie de la base et d'utiliser l'autre partie pour leur évaluation. Il n'était pas envisageable d'utiliser uniquement, même si cela avait été possible, des données acquises en exploitation réelle (*i. e.* avec de véritables usagers). En effet, même si les statistiques mettent en évidence un nombre toujours trop élevé d'événements violents depuis plusieurs années, de tels événements sont rares et il est statistiquement impossible d'en récupérer quelques instances sauf à sauvegarder le flux vidéo de toutes les caméras du réseau ferroviaire ou de croiser les enregistrements avec des déclarations officielles d'agression (mais cette fois sous la contrainte de la RGPD)

Ce mémoire est organisé en 5 chapitres. Le premier chapitre détaillera les concepts généraux qui seront abordés dans les chapitres suivants. Le second chapitre présentera l'état de l'art sur la reconnaissance d'activité humaine, la reconnaissance d'événement sonore, la combinaison des modes audio et vidéo et la reconnaissance d'actions violentes. Ensuite, le troisième chapitre présentera la base de données produite et les architectures proposées. Les chapitres 4 et 5 exposeront respectivement notre méthodologie et les résultats des évaluations et leur analyse. Finalement, nous conclurons ce travail et donnerons quelques perspectives.

Chapitre 1

Concepts

Ce chapitre est une introduction générale aux concepts que nous allons rencontrer et manipuler dans les chapitres suivants. Ce chapitre est organisé en partant des généralités sur l'apprentissage machine, en passant sur tout ce qui est en rapport avec la donnée utilisée pour les phases d'apprentissage ou d'évaluation, et en terminant avec la présentation des architectures neuronales profondes.

1.1 Modélisation et apprentissage

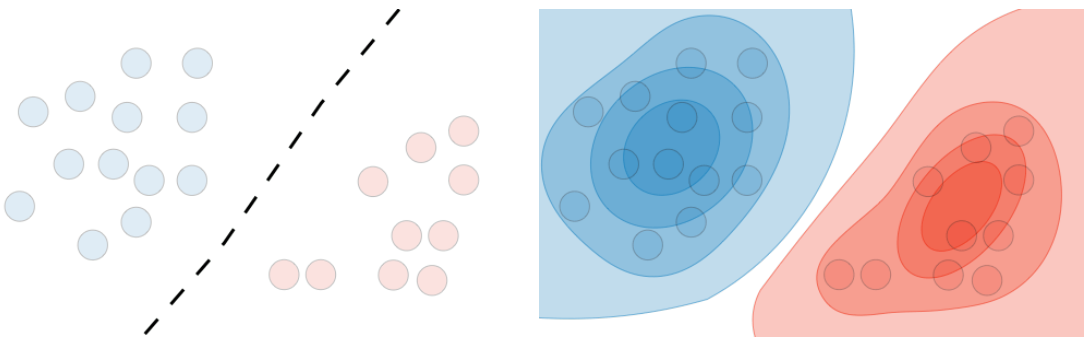
1.1.1 Modèle génératif *vs.* discriminatif

Les modèles génératifs et discriminatifs, illustrés dans la figure 1.1, sont les deux grandes familles de modèles en apprentissage machine.

La modélisation par modèle génératif est une technique d'apprentissage automatique qui consiste à utiliser des modèles probabilistes pour estimer la distribution de probabilité des données d'entraînement, permettant ensuite de générer de nouvelles données. Dans le cadre d'une application de classification, ces méthodes consistent à estimer la distribution de probabilité conditionnelle $p(x|y)$, x étant une donnée d'entrée et y l'étiquette de sa classe. À partir de l'estimation de cette distribution de probabilité conditionnelle, le calcul du maximum de vraisemblance $\max_y p(x|y)$ (équivalent au maximum *a posteriori* $\max_y p(y|x)$ sous certaines hypothèses [2]) permet de déterminer pour toute nouvelle donnée d'entrée x son appartenance à l'une des classes y du problème.

1. https://stanford.edu/%7Eshervine/teaching/cs-229/cheatsheet-supervised-learning

2. A partir de la formule de Bayes, $p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x|y)p(y)}{p(x)}$, ainsi sous l'hypothèse que y est équiprobable $\max_y p(y|x) \equiv \max_y \frac{p(x|y)}{p(x)} \equiv \max_y p(x|y)$



(a) Modèle discriminatif. (b) Modèle génératif.

FIGURE 1.1 – Illustration des modèles discriminatifs et génératifs [1].

Les approches discriminatives, quant à elles, sont des approches qui estiment une frontière de décision à partir des données. L'objectif est de trouver une fonction $f(x)$ qui prend en entrée une observation x et lui attribue l'étiquette y de la classe à laquelle elle peut appartenir. Cette méthode utilise un ensemble de données d'entraînement avec des observations étiquetées pour apprendre la relation entre les caractéristiques des données et l'étiquette correspondante. Dans un contexte de classification cette dernière approche est souvent plus performante lorsqu'on a peu des données par classe, ou que les distributions de probabilité des données sont difficilement modélisables de manière probabiliste. Ces approches appliquées sans noyau sont souvent utilisées lorsque les données sont linéairement séparables ou lorsque les caractéristiques qu'on en extrait sont déjà dans un espace de dimension appropriée pour une séparation linéaire. Cependant, si dans cet espace de base, le problème de classification est non linéairement séparable, l'utilisation de noyaux (*kernel trick*) peut être nécessaire pour transformer les données en un espace de dimension supérieure où une séparation linéaire est possible.

Nous rappelons ci-dessous quelques exemples de modèles :

— **génératifs** : les réseaux bayésiens naïfs (*Naive Bayes Classifier*), les modèles de mélange de gaussiennes (*Gaussian Mixture Models* (GMM)), les modèles de Markov cachés (*Hidden Markov Models* (HMM)), les architectures neuronales profondes de type auto-encodeur (*Autoencoder Network* (AE)) ou les architectures neuronales profondes de type antagoniste génératif (*Generative Adversarial Neural Network* (GAN)).

— **discriminatifs** : les arbres de décisions (*Decision Tree*), les forêts aléatoires (*Random Forest*), les machines à vecteurs supports (*Support Vector Machine* (SVM)) dans leur version avec et sans noyau, les architectures neuronales profondes de type convolutif (CNN) ou récurrent (RNN).

1.1.2 Les philosophies d'apprentissage

L'apprentissage automatique peut être divisé en deux grandes catégories : l'apprentissage supervisé et l'apprentissage non supervisé.

• **L'apprentissage supervisé** : résoudre une tâche avec cette approche nécessite que toutes les données soient annotées précisément (par exemple, borne de début et de fin, boîte englobante spatiale et identification des classes). Le jeu de données contient donc pour chaque donnée observée x une étiquette y. Il s'agit alors d'estimer des relations entre les observations d'entrée et les étiquettes correspondantes et d'aboutir à leurs généralisations pour prédire les étiquettes de nouvelles entrées i.e. non utilisées lors de l'apprentissage.

• **L'apprentissage non-supervisé** : résoudre une tâche avec cette approche ne nécessite pas que les données soient étiquetées. Il s'agit de découvrir des structures ou des relations cachées dans les données d'apprentissage, sans aucune connaissance sur leur appartenance à une classe. Cette approche est utilisée pour réaliser des tâches de segmentation, de *Clustering* ou des tâches de réduction de dimensions de données, etc.

Cependant, il n'est pas toujours possible d'obtenir des données étiquetées pour l'apprentissage supervisé. En effet, les données sont coûteuses à étiqueter ou difficiles à obtenir pour des raisons techniques ou réglementaires (CNIL, RGPD, etc.). Dans ces cas, les méthodes d'apprentissage semi-supervisé et faiblement supervisé peuvent être utilisées.

• **L'apprentissage semi-supervisé** : cette approche considère que seule une partie des données est annotée précisément. Le jeu de données contient donc des observations

avec des étiquettes et d'autres observations pour lesquelles l'étiquette n'est pas présente. Les données non étiquetées apportent des informations supplémentaires permettant de capturer la structure latente des données. Cela permet au modèle d'obtenir une représentation plus généralisable, améliorant ainsi la performance du modèle. Les données non étiquetées permettent également d'augmenter la quantité de données disponibles pour la phase d'entraînement, notamment dans le cadre des réseaux de neurones profonds.

- **L'apprentissage faiblement supervisé** : cette méthode d'apprentissage utilise des données avec des annotations moins précises ou incomplètes (par exemple, identification de classes sans bornes temporelles précises ou annotation par d'autres modèles).

1.1.3 Les stratégies d'apprentissages

En fonction de la taille du jeu de données à disposition pour résoudre une tâche, il peut être nécessaire d'adapter les stratégies d'apprentissages pour estimer les paramètres des modèles. Il existe trois principales stratégies d'apprentissage possibles :

- **À partir de zéro** : les paramètres des modèles sont initialisés aléatoirement et sont ensuite estimés sur le jeu de données à disposition. Cette stratégie est pertinente lorsque les données à disposition sont en nombre suffisantes vis-à-vis de la complexité du modèle.
- **Ajusté** : les paramètres de toute ou une partie du modèle sont pré-estimés sur un jeu de données de grande envergure et sont ensuite ajustés sur le jeu de données plus petit à disposition. On parle "d'adaptation de domaine" lorsque les classes sont les mêmes mais que les données proviennent d'environnements différents et de "transfert de connaissance" quand les environnements sont identiques mais que la tâche est différente.
- **Fixé** : les paramètres d'une partie d'un modèle, estimés sur un jeu de données de grande envergure, sont récupérés et fixés pour être introduit dans un nouveau modèle. Seuls les paramètres "non récupérés" du nouveau modèle sont estimés sur un plus petit jeu de données à disposition. Cette stratégie est pertinente lorsque le jeu de données à disposition est très petit.

1.2 Les données

1.2.1 La collecte des données

Aujourd'hui, grâce à la numérisation de la société et le développement de technologies de capteurs très différentes, de nombreux jeux de données sont disponibles dans la communauté. Ceci permet de développer, d'entraîner et/ou d'évaluer divers modèles liés à des tâches de détection, de reconnaissance, etc. Ces jeux de données peuvent appartenir à l'une de ces 4 catégories que nous décrivons en nous plaçant dans notre contexte applicatif :

- **Données réelles** : les données proviennent d'un système opérationnel composé d'un ou plusieurs capteurs de technologies pouvant être différentes. L'avantage de telles données est d'avoir des observations acquises directement dans l'environnement de l'étude. Cela peut être par exemple des images issues d'un système de vidéo-protection ou encore des images de matériels médicaux. Outre les règles de CNIL/RGPD à respecter, cette approche ne permet pas toujours de contrôler le contenu des observations acquises au regard des hypothèses de recherches spécifiées : par exemple,

sauf à regarder l'ensemble des images acquises, il est difficile de vérifier et d'identifier l'instant d'apparition d'une ou plusieurs actions violentes dans l'enregistrement d'un flux de vidéo-protection. L'annotation de ces données est chronophage, dépend de la sensibilité des annotateurs et présente la plupart du temps un déséquilibre entre les classes. Dans notre cas d'usage et notre contexte d'étude, il paraît évident que la classe "anormale" (activité avec violence) sera moins représentée que la classe "normale" dans des données provenant d'un système opérationnel.

- **Données scénarisées** : pour ce contexte, les données sont acquises dans l'environnement cible avec un système opérationnel ou expérimental avec un contrôle des enregistrements. Dans notre contexte par exemple, ceci est réalisé par un jeu d'acteurs suivant des scénarios définis. L'avantage est de pouvoir contrôler le contenu des données acquises autant que possible, tout en profitant de certaines contraintes de l'environnement cible. L'annotation de ces données est moins coûteuse car elle est conditionnée par des scénarios pré-définis : il est donc plus simple de savoir quand commence et quand se termine un scénario qui a été joué. Par ailleurs, il est possible de jouer à plusieurs reprises des actions de violence et donc d'en créer un grand nombre d'instances. Enfin cette mise en œuvre permet de respecter facilement les règles de la RGPD/CNIL.
- **Données simulées** : dans ce type de jeu de données, les données sont acquises par simulation numérique. Le premier avantage est de contrôler parfaitement le contenu et les conditions d'acquisitions afin de vérifier des hypothèses simples. Le deuxième avantage est qu'il est très facile d'obtenir les annotations puisque les données ont été produites à partir d'une description précise de chaque scénario. Enfin le dernier avantage est de ne poser que peu de problème de RGPD/CNIL. Dans le cadre de la vidéo et l'audio surveillance intelligente, ce type de données a été créé dans le projet DéGIV [188]. Même si d'énormes avancées ont été réalisées dans le cadre de la synthèse des images, de très grosses différences subsistent avec des données réelles équivalentes.
- **Données agrégées** : Ce type de jeu de données est construit en agrégeant des données de plusieurs origines. Souvent, ces jeux de données sont construits avec des données partagées publiquement sur internet (par exemple, des vidéos sur YouTube) ou issues de bases de données scénarisées. L'avantage de ce type de jeu de données est d'avoir des environnements et des contraintes variés permettant d'apprendre des modèles qui généralise mieux. Par contre, les données collectées ne sont pas contrôlées. Il n'est donc pas possible d'évaluer finement les limites de robustesse des modèles.

1.2.2 Les différents type d'annotation

En fonction de la stratégie d'apprentissage retenue pour aborder la tâche à résoudre, les données récoltées doivent être associées à des annotations. Ces annotations permettent de discriminer les données observées en fonction des classes traités dans la tâche. Il s'agit ici de définir comment chaque annotation est codée de manière informatique. Le codage sera différencié en fonction du type de classification visé :

- **Classification à une classe** : pour une tâche binaire, l'annotation consiste à attribuer la valeur 0 ou 1 à une observation. La valeur 1 est généralement attribuée aux observations appartenant à la classe d'intérêt.
- **Classification multi-classes** : pour cette tâche l'objectif est d'attribuer une classe parmi N à chaque observation à annoter. L'étiquette peut être encodée sous la forme d'un entier compris entre 0 et N ou sous la forme de vecteur de dimension N composé de 0 et dont seul la valeur correspondante à l'indice de la classe aura la valeur 1 (*One-Hot Encoding*).

- **Classification multi-étiquettes** : cette classification autorise à une observation d'appartenir simultanément à plusieurs classes. Ainsi, pour une observation donnée et dans le cas de N classes, l'étiquette est codée sous la forme de vecteur de dimension N composé de 0 et dont seuls les valeurs correspondantes aux indices des classes présentes dans l'observation auront la valeur 1.

1.2.3 La mise en œuvre de l'annotation des données

Dans le cadre de données enregistrées à partir d'un système d'audio et de vidéo protection, l'annotation des données peut être réalisée soit de manière manuelle où chaque observation est regardé et/ou écouté par une ou plusieurs personne(s) qui lui attribue(nt) l'une des classes considérées ou de manière automatique en appliquant des modèles d'IA pré-entraînés pour reconnaître l'ensemble ou une partie des classes considérées. Une association de ces deux premières mises en œuvre est possible avec une annotation semi-automatique où les classes reconnues sont vérifiées manuellement. Lorsque les données proviennent de plates-formes de partage en ligne, les mots-clés associés peuvent permettre de produire une annotation.

Que cela soit pour le signal sonore ou les images, les informations annotées dépendent de la tâches envisagées i.e. détection, identification (classification), localisation, segmentation etc. Ainsi, pour une image, il peut s'agir d'ajouter une étiquette aux objets d'intérêt qui y sont présents, éventuellement en ajoutant sa position (boîte englobante), son orientation, etc. Dans le cas ou chaque objet doit être segmenté de manière précise, chaque pixel de l'objet pourra alors être associé à une même étiquette. Dans le cas d'un signal sonore, il s'agit d'annoter le type de son, sa localisation, sa durée, etc. Dans le cas d'un signal temporel ou d'une succession d'images dans le temps (une vidéo) il faut tenir compte du fait que de telles informations nécessitent d'être associées à une durée i.e. une étiquette est nécessairement associée à une portion du signal avec un début et une fin.

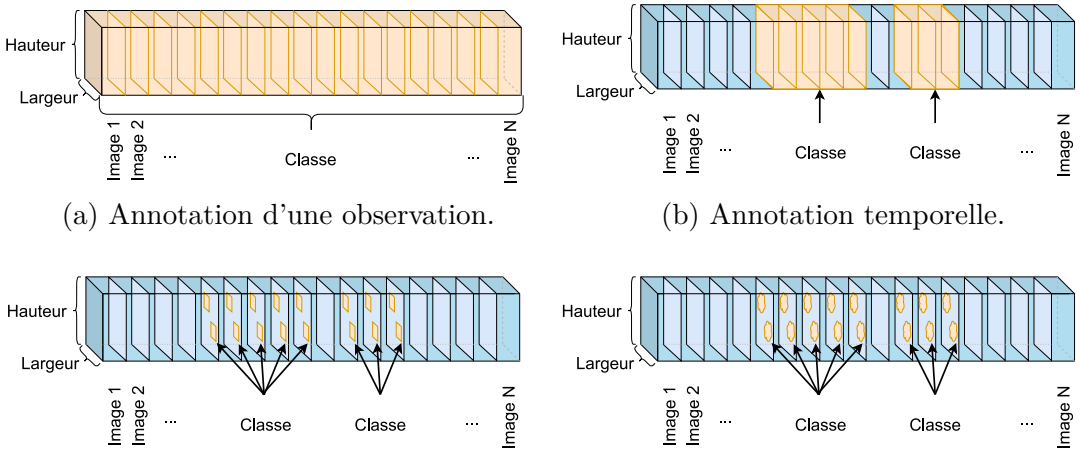
L'annotation des données peut alors prendre plusieurs formes que nous illustrons dans le cas d'une séquence d'images dans la figure 1.2 :

- **Une annotation d'observation** : une ou plusieurs classe(s) est (sont) assignée(s) à une observation i.e. une image ou un segment sonore. Ce type d'annotation informe seulement de la présence (sans localisation spatiale pour une image et sans localisation temporelle pour un segment sonore) d'une ou plusieurs classes dans l'observation considérée. Par exemple la présence de deux personnes dans une image ou la présence de paroles sur un segment sonore.

- **Une annotation temporelle** : l'étiquette est posée sur plusieurs images ou plusieurs échantillons sonores consécutifs. L'annotation est alors associée à un début et une fin. Ce type d'annotation s'applique par exemple lorsqu'il s'agit d'étiqueter une activité humaine dans un ensemble d'images ou présence de sons les cas d'une succession d'échantillons sonores.

- **Une annotation spatiale** : dans une observation une boite englobante est définie pour chacune des classes présentes dans une image. Appliquée à une séquence d'images, ce type d'annotation forme une annotation spatio-temporelle qui vient préciser l'annotation décrite précédemment.

- **Une annotation pixellique** : dans une observation chaque pixel d'une image ou d'une vidéo est associé à une classe. Ce type d'annotation permet une annotation plus détaillée que l'annotation spatiale décrite précédemment. Elle complète de la même manière une annotation temporelle.



(a) Annotation d'une observation.

(b) Annotation temporelle.

(c) Annotation spatio-temporelle.

(d) Annotation segmentation.

FIGURE 1.2 – Illustration de la mise en œuvre de l'annotation d'une séquence d'images (vidéo).

Afin de réduire l'imprécision des annotations et qu'elles soient aussi cohérentes que possible, les règles d'annotation doivent être clairement exprimées. Dans le cas idéal, deux annotateurs minimum sont requis afin de limiter la subjectivité inhérente à chacun et de rejeter d'éventuelles erreurs : leurs annotations sont alors fusionnées en utilisant des méthodes telles que le vote majoritaire ou les annotations peuvent être le fruit d'une interpolation s'il s'agit par exemple d'annoter la position d'un objet.

1.2.4 L'augmentation de données

Les architectures neuronales profondes que nous allons mettre en œuvre montrent des performances élevées dans de nombreux domaines si elles sont comparées aux méthodes d'apprentissage plus traditionnelles. Ce gain de performance est dû notamment à leur structure qui nécessitent d'estimer un très grand nombre de paramètres et qui doit donc faire appel à des jeux de données de grande taille (de plusieurs milliers à plusieurs millions d'exemples) afin d'assurer leur apprentissage. Si la taille du jeu de données est insuffisante, on s'expose à du "sur-apprentissage" i.e. le réseau apprend "par cœur" les données utilisées pour son apprentissage ce qui limite sa capacité à généraliser la tâche pour laquelle il est entraîné. Lorsqu'un grand jeu de données n'est pas disponible, une stratégie est d'augmenter les données pour obtenir des exemples supplémentaires tout en étendant sa variabilité.

Dans un jeu de données, l'augmentation est appliquée uniquement sur l'ensemble d'entraînement en venant compléter la base d'apprentissage avec des données jamais observées. Ainsi un modèle appris avec ces nouvelles observations a une meilleure capacité de généralisation. En fonction de la technique d'augmentation utilisée, l'annotation peut rester la même après une augmentation. Cependant, dans certains cas l'augmentation de données peut ne pas préserver le contenu d'une observation et provoquer des nouvelles classes [129] (par exemple, une séquence vidéo peut-être retournée pour créer une autre action). Pour ne pas modifier l'annotation associée à la donnée observée, il est nécessaire de faire attention à appliquer des opérations d'augmentation qui respecte le contenu des données initiales. Plus globalement, il est primordial d'avoir une bonne connaissance des signaux mis en jeu dans la base afin d'appliquer une augmentation en évitant d'introduire un biais dans les images ou le son tel que l'annotateur n'aurait peut être pas réussi à identifier l'étiquette initiale.

Voici quelques exemples d'opérations applicables aux images permettant d'augmenter une base de données initiale : augmentation ou diminution de la luminosité, découpage, retournement vertical, horizontal, temporel. Dans le cas d'un signal sonore, l'augmentation peut consister à jouer avec sa caractéristique temporelle comme par exemple sa durée, sa fréquence d'échantillonnage.

Dans un cadre plus général applicable à plusieurs types de signaux, il est possible d'augmenter le nombre de données en ajoutant du bruit ou en générant de nouvelles données synthétiques avec des réseaux de neurones génératifs (transfère de style, AE, GAN), etc.

1.2.5 Représentation des données en entrée

Historiquement, les modèles construits par apprentissage ne prenaient pas directement en entrée des données sorties des capteurs, dites "brutes". En fonctions des signaux d'intérêt, du contexte applicatif et du but recherché, il était généralement nécessaire de réaliser des (pré)traitements sur ces signaux afin de les uniformiser et d'en faire ressortir l'information pertinente. Le traitement de ces données, faisant appel à une certaine expertise et des analyses approfondies, fournissaient ce que l'on appelait des paramètres d'entrée "haut niveau" aux modèles que nous cherchions à estimer. Aujourd'hui, les architectures neuronales profondes prennent en entrée des représentations des données de plus "bas niveau" voire même des données "brutes". Nous précisons un peu plus ces notions ci-après.

1.2.5.1 Représentation brutes

Une donnée brute est une donnée provenant directement d'un capteur et qui n'a subi aucune transformation. Dans le contexte de la vidéo, ces données peuvent être une ou une succession d'images en niveau de gris ou en couleur (RVB) acquises par un appareil photo (Figure 1.3a), ou par par une caméra. Dans le contexte de l'audio, les données brutes sont relatives à une onde sonore acquise par un microphone (Figure 1.3b).

1.2.5.2 Représentation de bas niveau

Une donnée de bas niveau est obtenue en appliquant un (pré)traitement sur de sa version brutes. Ces (pré)traitements ont pour objectifs de mieux conditionner les données lors de la phase d'apprentissage afin d'améliorer la convergence des modèles et de limiter la présence de données aberrantes. En fonction du problème posé, ils peuvent également projeter les signaux dans un nouvel espace plus descriptif.

* **La standardisation** : cela consiste à supprimer la moyenne et à établir l'écart-type des données à 1.
* **La mise à l'échelle** : cela consiste à fixer l'amplitude des données en entrée. Par exemple, en image cela revient à passer l'amplitude des pixels de [0-255] à l'amplitude [0-1] ou [-1−1]. La normalisation d'une variable est une forme de mise à l'échelle qui tient compte de sa valeur maximale et minimale sur l'ensemble de la base d'apprentissage.
* **La différence entre deux signaux** : lorsqu'il s'agit de définir une architecture neuronale en vue d'analyser les mouvements apparents dans une séquence d'images, il peut être intéressant de procéder à une différence entre des images successives ou entre une image courante et une image du fond (*background removal*). Cette opération permet de détecter les objets en mouvement entre des images successives lorsque la caméra est fixe [145]. Une illustration de cette transformation est présentée dans la figure 1.3c.

* **Le flot optique** : appliqué un algorithme d'extraction du flot optique permet d'estimer le champ des vecteurs du mouvement apparent présent dans une séquence d'images. De tels algorithmes estiment l'amplitude et la direction du mouvement apparent en chaque pixel d'une image. Une illustration de cette transformation est présentée dans la figure 1.3e.
* **La transformée de Fourier** (Fast Fourier Transformation - FFT) : il s'agit de la transformation d'un signal du domaine temporel au domaine fréquentiel. Cette opération permet de représenter le signal dans une forme compacte. Dans le cas du signal sonore, cette représentation est calculée sur le signal numérisé avec l'opération Discrete Fourier Transform (DFT) (Équation 1.1). Une illustration de cette transformation est présentée dans la figure 1.3d.

$$\hat{x}(k) = \sum_{n=0}^{N-1} x(n) e^{-i\frac{2\pi k}{N}n}, k = 0, ..., N-1 \tag{1.1}$$

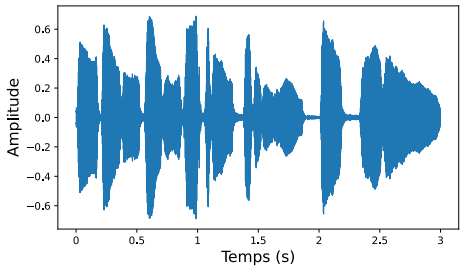
  où $x(n)$ représente le signal numérique temporel de 0 à $N-1$ échantillons.
* **Le spectrogramme** : est une représentation en temps et en fréquence du signal permettant d'observer son évolution temporelle et fréquentielle. Cette représentation est construite à partir des valeurs consécutives des modules des FFT calculées successivement au cours du temps. Dans le cas du signal sonore la fenêtre de calcul est de 20 à 30 ms, durée sur laquelle le signal est considéré comme stationnaire. Cette fenêtre est ensuite déplacée avec un chevauchement généralement de 50%. Cette représentation peut éventuellement être exprimée en échelle log.
* **Le mel-spectrogramme** : cette représentation, inspirée de la perception humaine, est une conversion du spectrogramme avec une échelle de type logarithmique sur l'axe des fréquences. Cette conversion est réalisée en calculant les énergies d'un banc de filtres appliqué au spectre original. Les filtres sont des filtres passe bande triangulaires espacés selon l'échelle des mel [149]. Le mel-spectrogramme est obtenu en effectuant cette transformation sur chacun des spectres composant le spectrogramme. Une illustration de cette transformation est présentée dans la figure 1.3f.
* **Les Mel-frequency cepstral coefficients (MFCC)** : ces coefficients sont obtenus à partir du mel-spectrogramme en appliquant une opération de cosinus discret (DCT) [37]. Cette opération est une transformation orthogonale qui mathématiquement permet d'obtenir entre autre une décorrélation des caractéristiques fréquentielles.

1.2.5.3 Représentation de haut niveau

Les représentations de haut niveau sont obtenues à partir des données brutes sur lesquelles des traitements plus approfondis ont été appliqués. Ces traitements qui n'ont pas été appris sur des données ont pour objectif d'extraire et de structurer l'information utile et nécessaire propre aux signaux et à la tâche à réaliser. En conséquence, les performances du système ne reposent pas seulement sur le modèle appris, mais aussi sur le modèle extrayant la représentation depuis les données brutes ou de bas niveau :

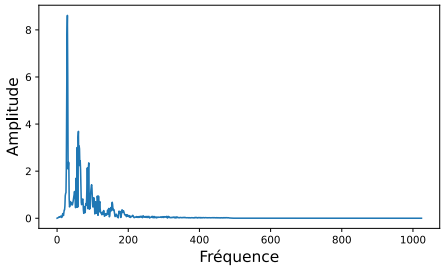
* **Extraction de caractéristiques génériques** : les projections intermédiaires de modèles appris, qui traitent des représentations brutes ou de bas niveau, peuvent être utilisées comme des représentations fournies en entrée d'un autre modèle. En fonction du niveau d'extraction des projections, l'information est plus ou moins structurées.
* **L'analyse en composantes principales** : l'objectif est de réduire le nombre de variables en les projetant linéairement dans un espace de plus faible dimension appelés "composantes principales"

(a) Image originale.

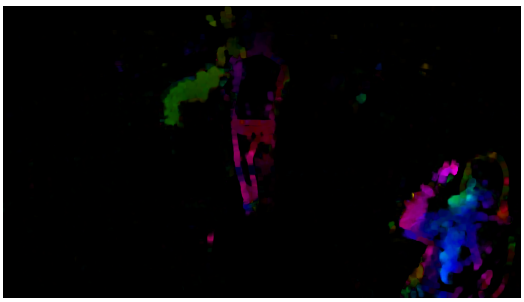


(b) Signal sonore original.

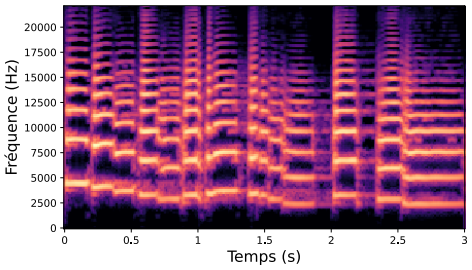
(c) Différence entre deux images.



(d) FFT calculée sur 20ms d'un signal sonore.



(e) Flot optique.



(f) Mel-Spectrograme.

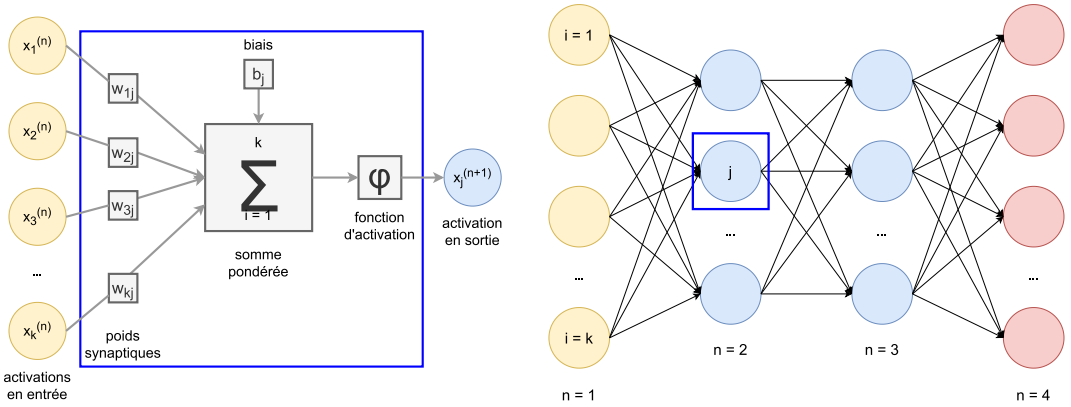
FIGURE 1.3 – Exemples de différents niveaux de représentation de signaux. (a) et (b) : Représentation "brutes". (c), (d), (e) et (f) : Représentation "bas niveau".

1.3 Les architectures neuronales profondes

Inspirées par les systèmes de traitement de l'information biologique, les architectures neuronales profondes permettent aujourd'hui et pour certaines tâches, d'atteindre des performances comparables aux performances humaines. Les architectures sont une succession de couches (Figure 1.4b) composées de plusieurs neurones artificiels effectuant des opérations sur des données d'entrée. Une couche est connectée à la suivante au travers de ses résultats de sortie permettant ainsi d'extraire et de structurer, depuis des données fournies en entrée de la première couche, des caractéristiques pour finalement fournir le résultat attendu en sortie du réseau.

Les architectures peuvent être composées des types de réseaux suivants :

Un réseau entièrement connecté (ou fully connected en anglais) : ces réseaux sont adaptés au traitement d'entrée de type vecteur unidimensionnel. Ils sont composés de quelques couches dont les neurones d'une couche sont totalement connectés aux neurones de la couche suivante. La sortie de chaque neurone est la somme pondérée de ses entrées et d'une valeur de biais (Figure 1.4a). La sortie d'un tel réseau est un vecteur unidimensionnel (Figure 1.4b).



(a) Cellule simple neurone.

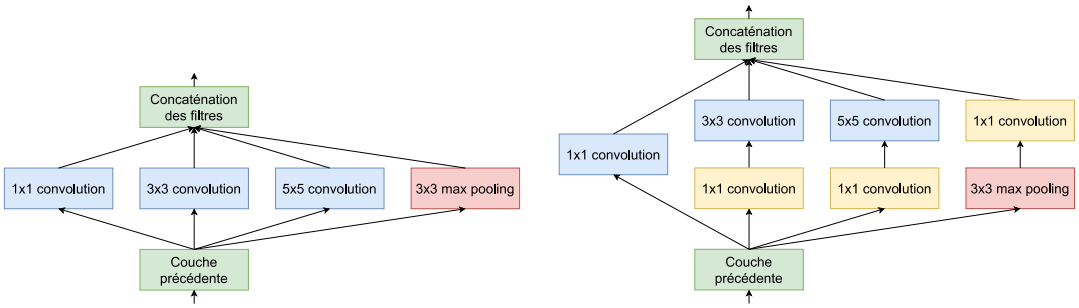
(b) Architecture composée de couches "entièrement connectées".

FIGURE 1.4 – Exemple d'une cellule "simple neurone" composant une couche "entièrement connectées".

Les réseaux de neurones convolutifs (convolutional neural networks) [93] : ce sont des réseaux de neurones qui sont conçus pour être appliqués sur des données à plusieurs dimensions telles que des images. Ils sont composés d'une succession de couches convolutives. Chaque couche est composée de cellules opérant des filtres appliqués localement aux données d'entrée. Ainsi les paramètres d'un filtre permettent d'extraire une certaine caractéristique quelle que soit la position dans les données d'entrée. L'usage de convolutions permet de réduire considérablement le nombre de paramètres à estimer lors de l'apprentissage comparativement à un réseau entièrement connecté équivalent. Des couches de sous-échantillonnage (pooling) peuvent être utilisées pour réduire la dimension de la sortie d'une couche en préservant les caractéristiques les plus importantes. Ces réseaux sont souvent utilisés pour produire un espace de caractéristiques (feature map) dans lequel l'entrée du réseau est projetée et sont combinés à des réseaux entièrement connectés pour assurer par exemple des tâches de reconnaissance ou de classification.

Les réseaux de neurones récurrents (recurrent neural networks) : ce sont des réseaux de neurones où l'information ne circule pas uniquement de la couche d'entrée vers la couche de sortie. La sortie de certaines couches reviennent en arrières pour être connectées à l'entrée de ces couches et ainsi former une boucle. Ces réseaux sont vus comme des modules capables d'extraire des caractéristiques présentes dans des séquences temporelles de données (*Recurrent Neural Network* (RNN) *vs. Long Short Term Memory* (LSTM) [74], *Gated Recurrent Unit* (GRU) [24]). Ils sont très souvent associés à des réseaux convolutifs et entièrement connectés afin de remplir des tâches liées au langage naturel, à la traduction, à la reconnaissance de la parole, ou à la reconnaissance d'activités humaines.

Un réseau de neurones peut être aussi modulaire. Les "modules" sont constitués d'une succession de couches connectées entre elles telles qu'elles ont été décrites précédemment : couche de convolution, couche de pooling, couche récurrente. Les modules peuvent être répétés plusieurs fois pour former une architecture plus complexe et profonde. Le module *Inception* introduit dans [154] en 2014 en est une illustration. Le module Inception a été proposé pour résoudre le problème de convergence lors la phase d'apprentissage des réseaux très profonds dont le nombre de paramètres est très élevé. Le module Inception utilise une combinaison en parallèle en en série de convolutions de différentes tailles de filtre et de pooling pour extraire des caractéristiques pertinentes à différentes échelles spatiales tout en réduisant le nombre de paramètres. La première version du module



(a) Module *Inception* original. (b) Module *Inception* amélioré.

FIGURE 1.5 – Modules *Inception*.

Inception est présentée à la figure 1.5 (a). Sa version améliorée présentée sur la figure 1.5 (b) tire bénéfice des convolutions (1×1). Ce type de module, introduit avec des convolutions 2D pour de la classification d'images, a été porté avec des convolutions 3D [19] pour de la classification de vidéos.

En 2015, le module *ResNet* a été proposé par He *et al.* dans [68] pour résoudre le problème de l'évanouissement et l'explosion du gradient lors de l'apprentissage des architectures très profondes. Pour résoudre ce problème, ce nouveau module propose d'ajouter une connexion "résiduelle" entre la sortie de la couche précédente et la sortie de deux couches en avant comme cela est présenté sur la figure 1.6. Ce module a permis de définir des architectures beaucoup plus profondes. Comme précédemment, ce type de module introduit avec des couches de convolutions 2D pour de la classification d'images a été porté avec des convolutions 3D [65] pour de la reconnaissance d'actions dans des vidéos.

1.4 Les combinaisons multimodales

Inspiré du monde du vivant, qui utilise plusieurs perceptions sensorielles dans ses observations, ses compréhensions, ses décisions (la vue, l'ouïe, l'odorat, le touché, le goût), le champ de recherche du traitement automatique s'est intéressé à la combinaison de plusieurs signaux dans des architectures dites "multi-branches". Ces sources peuvent provenir de mêmes signaux comme des séquences images RVB adjointes à des flots optiques, des squelettes, etc. Ils peuvent aussi provenir de signaux différents comme la séquence d'images RVB d'un signal vidéo et des spectres d'un signal audio. Les intérêts de combiner des sources provenant de différents capteurs sont multiples : augmenter la précision des décisions, augmenter la résilience à la présence de bruit sur un signal, être résilient lorsque qu'un signal est absent, etc.

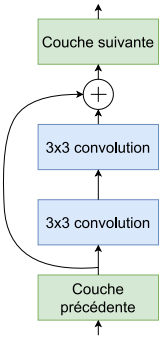


FIGURE 1.6 – Module *ResNet*.

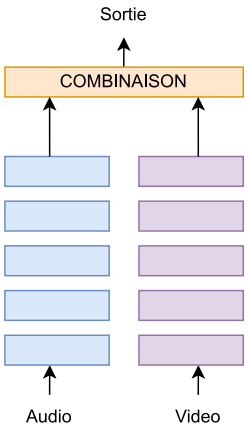


FIGURE 1.7 – Combinaison de modèles uni-modaux traitant des signaux audio (a) et vidéo (v)

1.4.1 Les niveaux de combinaisons

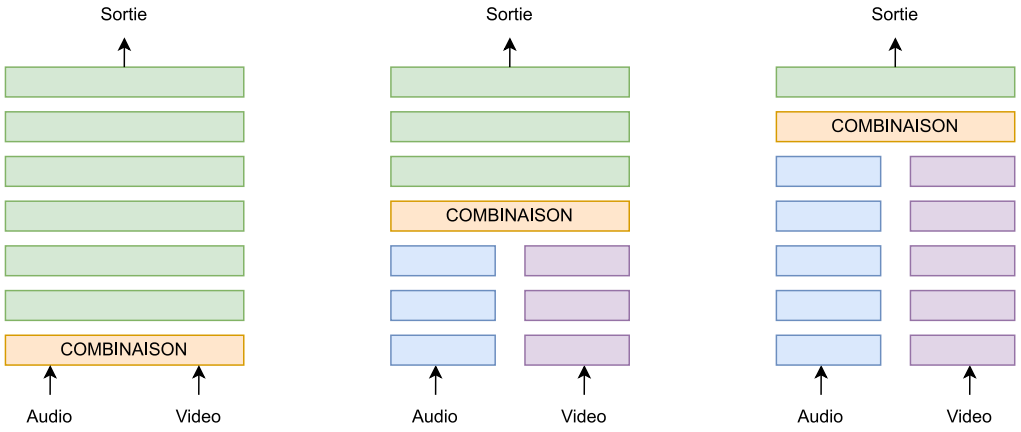
Dans la communauté, la catégorisation des types de combinaisons est principalement définie sur leur "niveau" de combinaison [168, 12, 43], c'est-à-dire la profondeur dans le réseau à laquelle les branches se combinent dans l'architecture neuronale, pour être traitées simultanément. La figure 1.8 présente différents niveaux de combinaison.

Une première approche consiste à considérer la combinaison de modèles uni-modaux (Figure 1.7). Dans ce cas, une architecture spécifique à chaque branche est définie et apprise indépendamment l'une de l'autre. La combinaison la plus élémentaire se réalise alors par la combinaison des décisions de chaque architecture, par des fonctions logiques de types "ET", "OU", etc. Une autre combinaison possible est d'établir et d'apprendre un nouveau modèle, défini par de simples valeurs de seuils ou par des modèles plus évolués d'apprentissage machine (réseau entièrement connecté, SVM, HMM, etc.). Dans ce dernier cas les entrées du modèle de combinaison peuvent être directement les valeurs de décision de chacune des architectures, soit les scores amonts ayant abouties à ces décisions.

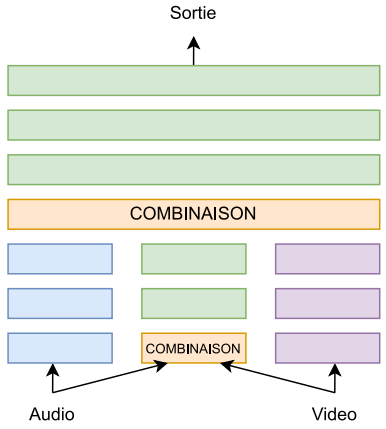
Une seconde approche consiste à considérer une architecture multi-modale dont la couche de décision et l'ensemble des branches sont apprises conjointement. Dans ce contexte, on recense dans [168, 12, 43] 3 types de combinaisons : la combinaison précoce (*early*), la combinaison moyenne (*middle*), la combinaison tardive (*late*) :

La combinaison précoce (Figure 1.8a), a pour objectif de combiner les données au plus près des capteurs. Dans le cas de signaux de même nature, cette combinaison peut déjà être effectué sur des représentations de bas niveau. Dans le cas où les données sont hétérogènes, il est nécessaire d'effectuer des traitements sur chacun des signaux pour leur donner un sens les uns par rapport aux autres ; cette combinaison précoce se réalise alors sur une représentations haut niveau (cas de signaux audio et vidéo).

La combinaison moyenne (Figure 1.8b) et tardive (Figure 1.8c) sont quant à elles bien plus souvent considérées. En effet les représentations bas niveaux ne permettent pas toujours de révéler efficacement les cohérences et les complémentarités des signaux. La combinaison de paramètres de haut niveau devient alors nécessaires à la modélisation des relations entre signaux. Ces niveaux de combinaisons, plus ou moins proches des couches de décisions, sont fonction de la nature des signaux, de la tâche à réaliser et de l'environnement dans lequel cette dernière est considérée.



(a) Combinaison précoce (early). (b) Combinaison moyenne (middle). (c) Combinaison tardive (late).



(d) Combinaison multi-niveaux.

FIGURE 1.8 – Niveaux de combinaisons [168] sur des architectures neuronales combinant des représentations provenant des branches audio (a) et vidéo (v).

Les combinaisons peuvent aussi être réalisée à différents niveaux pour une même architecture et devenir hybrides comme le présente la figure 1.8d.

Les études portées sur ces différents niveaux de combinaisons, uni-modales ou mutlti-modales, dans [168, 12, 43] aboutissent à la conclusion générale que le niveau de combinaison optimal est spécifique à la tâche et au type de signaux combinés. De plus, Vielzeuf dans [168] a expérimentalement observé qu'une combinaison précoce est plus intéressante pour les signaux de faibles dimensions, et qu'une combinaison tardive est plus intéressante pour des signaux de grandes dimensions.

1.4.2 Les mises en œuvre des combinaisons

Un autre point clé des combinaisons concerne les différentes structures pour leur mises en œuvre [12] : la combinaison par concaténation, la combinaison par opérations mathématiques (somme, multiplication, convolutions, etc.), la combinaison par mécanismes à portes ou encore la combinaison par attentions croisées. Nous proposons dans ce qui suit d'expliciter ces diverses techniques en considérant un signal audio (noté a) et vidéo (noté v). Le résultat de la combinaison sera notée z.

1.4.2.1 Combinaison par concaténation

La combinaison la plus élémentaire est celle par concaténation et consiste à assembler côte à côte les représentations de caractéristiques provenant de chaque branche (Équation 1.2). Les cohérences entre les branches sont modélisées par les couches suivantes de l'architecture.

$$z = concat([v, a]) \tag{1.2}$$

1.4.2.2 Combinaison par opérateurs mathématiques

La combinaison par opérateurs mathématiques consiste à effectuer des opérations élémentaires terme à terme entre les éléments de chaque branche. Les opérations les plus courantes sont l'addition (Équation 1.3) et la multiplication (Équation 1.4).

$$z = v + a \tag{1.3}$$

$$z = v \times a \tag{1.4}$$

Pour ce type de combinaison il est préférable que les espaces de représentations de chaque branche soient similaires et de même dynamique, et qu'ils soient de même dimension dans le cas de la multiplication. Dans le cas contraire, une transformation de chaque branche est nécessaire avant opérations :

$$z = W_v \cdot v + W_a \cdot a \tag{1.5}$$

$$z = W_v v \times W_a a \tag{1.6}$$

Chacune de ces deux dernières équations représente une fonction distincte : la première traite l'information de façon cumulative, la seconde de façon sélective.

1.4.2.3 Combinaison par mécanisme à portes

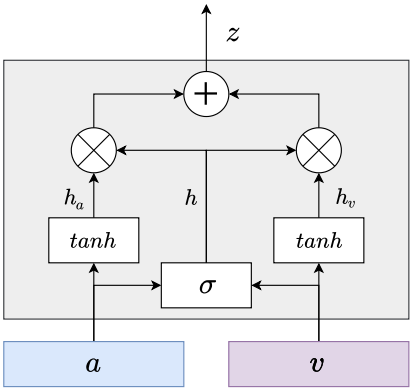
Cette structure de combinaison (Figure 1.9a) est inspirée des mécanismes à portes développés dans les couches récurrentes GRU et LSTM. L'idée est de cumuler les deux opérations vu précédemment : sélectionner et additionner les caractéristiques les plus pertinentes de chacune des branches. Ce type de combinaison est exprimé par l'équation :

$$\begin{aligned} h_v &= tanh(W_v \cdot v) \\ h_a &= tanh(W_a \cdot a) \\ h &= \sigma(W_h \cdot [v, a]) \\ z &= h \times h_v + h \times h_a \end{aligned} \tag{1.7}$$

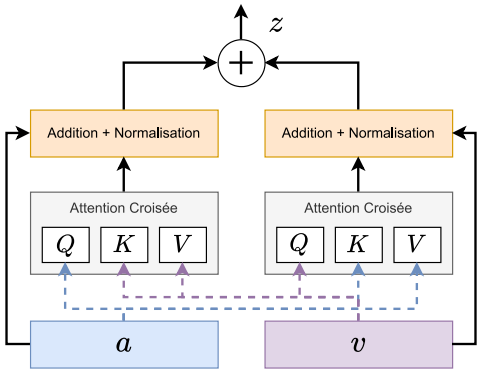
où $\times$ est la multiplication terme à terme.

1.4.2.4 Combinaison par attention croisée

La notion d'attention a été introduite dans les architectures neuronales dédiées au traitement du langage naturel [165]. L'objectif de l'attention est de mettre en avant les caractéristiques les plus intéressantes de chaque branche et de mettre en retrait les caractéristiques les moins intéressantes de chaque branche pour la prise de décision finale. Ainsi l'attention permet aux couches suivantes de se concentrer sur les caractéristiques importantes mêmes si elles sont petites. Appliqué à la combinaison, l'attention croisée entre les branches permet de mettre en avant des caractéristiques cohérentes entre les représentations issues de chacune des branches. L'attention croisée (Figure 1.9b) est réalisée



(a) Module de combinaison par mécanisme à porte.



(b) Module de combinaison par attention croisée.

FIGURE 1.9 – Modules de combinaison par mécanisme à porte et par attention croisée.

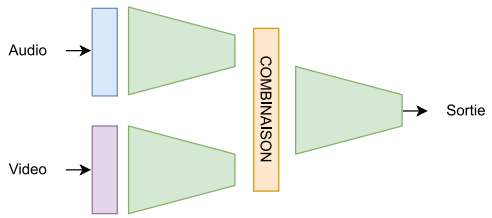
entre les représentations de chacune des branches. Ce type de combinaison est exprimé par l'équation :

$$\begin{aligned} Attention(Q,K,V) &= softmax(\frac{QK^\top}{\sqrt{d_k}})V \\ h_{av} &= Attention(a,v,v) \\ h_{va} &= Attention(v,a,a) \\ h_a &= Norm(h_{av}+a) \\ h_v &= Norm(h_{va}+v) \\ z &= h_a + h_v \end{aligned} \tag{1.8}$$

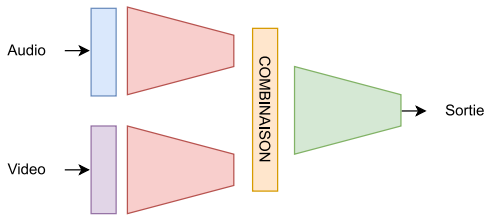
où Q, K et Vreprésentent respectivement la "requête", la "clé" et la "valeur" dans une couche d'attention comme définis dans [165]. Cette technique de combinaison permet de comparer chaque étape de temps de la représentation audio avec chaque étape de temps de la représentation vidéo avec l'attention h_{av} ; et inversement avec l'attention h_{va}.

1.4.3 Stratégies d'apprentissage des architectures multi-modales

Comme nous l'avons vu en section 1.4.1, l'apprentissage des architectures mutli-modales consiste à estimer l'ensemble des paramètres de l'architecture avec une stratégie de "bout



(a) Apprentissage des architectures de combinaisons de bout en bout.



(b) Apprentissage des architectures de combinaisons avec des extracteurs de caractéristiques uni-modaux fixés.

FIGURE 1.10 – Stratégies d'apprentissage d'architectures multi-modales combinant des branches audio et vidéo. Les paramètres des branches vertes de l'architecture sont conjointement appris, les paramètres des branches rouges sont appris sur un autre jeu de données et ne sont pas ré-appris.

en bout" (Figure 1.10a). Pour mettre en œuvre cette stratégie, il est nécessaire d'avoir à disposition un large jeu de données mutli-modales qui se caractérisent nécessairement par des données synchronisées (*cf.* les architectures multi-branches vidéo [143, 171, 172, 101] où chacun des modes sont issus du même signal vidéo)

A défaut d'avoir à disposition un large jeu de données la seconde stratégie est d'exploiter des extracteurs de caractéristiques uni-modaux, comme dans [157, 15]. Les paramètres de ces extracteurs sont estimés séparément sur différents jeux de données indépendants et fixés lors de l'estimation de l'architecture réalisant la combinaison (Figure 1.10b). Pour jouer correctement leur rôle d'extracteurs de caractéristiques, il est nécessaire que ces architectures soient suffisamment génériques pour ne pas mettre en défaut les cohérences et complémentarités de la multi-modalité.

Lorsque les caractéristiques sont issues de signaux de natures différentes comme l'audio et la vidéo, Xiao *et al.* dans [181] ont montré qu'il pouvait exister un déséquilibre de l'apprentissage conjoint en fonction des modes. Plus précisément l'estimation des paramètres de l'architecture peut converger vers une solution pour laquelle une des caractéristiques n'est pas correctement modélisée. Ceci peut être dû à un mode plus prépondérant qu'un autre pour certaines observations ou qu'une branche de l'architecture est constituée de moins de paramètres. Pour faire face à cette difficulté, Xiao *et al.* ont donc proposé de désactiver l'une ou l'autre des deux branches (audio ou vidéo) au cour de la phase d'estimation des paramètres, aléatoirement et réciproquement.

1.5 L'évaluation

L'évaluation des performances d'un réseau de neurones est une étape essentielle dans le développement d'un modèle en apprentissage machine. Elle permet de mesurer la capacité de généralisation du modèle à de nouvelles données et donc à fournir des prédictions précises i.e. conformes aux annotations. Pour cela, plusieurs métriques et outils peuvent être utilisés. L'évaluation est menée à partir des éléments d'une base de données totalement inconnues de la phase d'apprentissage. Nous présentons simplement ci-dessous les métriques classiques que nous utilisons dans nos travaux ou que nous comptions utiliser dans la suite de nos travaux.

1.5.1 Matrice de confusion

La matrice de confusion (Figure 1.11) est un moyen classique et simple pour apprécier la performance d'un système de classification. Elle se décline de la même manière dans un cadre de classification binaire ou multi-classe. Dans le cadre d'une reconnaissance à 2 classes (une classe positive et une classe négative) telle que la détection d'un évènement anormal (la classe "positive"), la matrice de confusion est composée des métriques suivantes (les valeurs exprimées en pourcentage sont indiquées entre parenthèses) :

- Les **Vrais Positifs** (VP) : Nombre d'éléments annotés "positifs" détectés comme "positifs" ($100 \times \mathrm{VP}/(\mathrm{VP}+\mathrm{FN})$) ;
- Les **Vrais Négatifs** (VN) : Nombre d'éléments annotés "négatifs" détectés comme "négatifs" ($100 \times \mathrm{VN}/(\mathrm{VN}+\mathrm{FP})$) ;
- Les **Faux Positifs** (FP) ou Fausse alarme : Nombre d'éléments annotés "négatifs" détectés comme "positifs" ($100 \times \mathrm{FP}/(\mathrm{VN}+\mathrm{FP})$) ;
- Les **Faux Négatifs** (FN) ou Détection ratée : Nombre d'éléments annotés "positifs" détectés comme "négatifs" ($100 \times \mathrm{FN}/(\mathrm{VP}+\mathrm{FN})$).

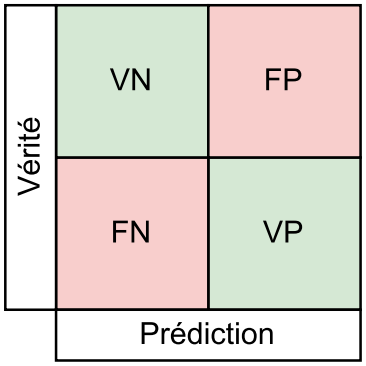


FIGURE 1.11 – Matrice de confusion

Il faut noter que dans ce cadre binaire, la matrice de confusion dépendra de la valeur de seuil définie pour décider de la présence ou non d'une anormalité.

Dans le cadre multi-classe, une seule matrice de confusion existe mais cette fois avec un nombre de lignes et de colonnes égal au nombre de classes. La matrice regroupe alors, par classe, les faux positifs, les faux négatifs et les vrais positifs.

Notons qu'une architecture qui montre des performances parfaites est associée à une matrice de confusion diagonale.

1.5.2 L'exactitude

L'**Exactitude** (*Accuracy*) est le rapport du nombre de prédictions correctes (VN et VP dans le cas binaire) sur l'ensemble de données et s'exprime ainsi :

$$\text{Exactitude} = \frac{VP + VN}{VP + VN + FP + FN} \tag{1.9}$$

Elle est bornée entre 0 et 1. La valeur maximale "1" exprime une absence totale d'erreur ($FP = FN = 0$). Cette métrique peut s'exprimer en pourcentage.

1.5.3 La précision

La **Précision** (*Precision*) est le rapport du nombre de vrais positifs sur le nombre total de classifications positives et s'exprime ainsi :

$$\text{Précision} = \frac{VP}{VP + FP} \tag{1.10}$$

Elle est bornée entre 0 et 1, où la valeur "1" exprime qu'aucun faux positif n'a été identifié($FP = 0$). Comme précédemment, elle peut s'exprimer en pourcentage.

1.5.4 Le rappel

Le **Rappel** (Sensibilité ou *Recall*) est le rapport du nombre de vrais positifs sur le nombre total d'éléments positifs contenu dans la base. Elle s'exprime ainsi :

$$\text{Rappel} = \frac{VP}{VP + FN} \tag{1.11}$$

Le rappel est borné entre 0 et 1. La valeur maximale "1" exprime l'absence de positifs oubliés, autrement dit l'algorithme a détecté l'ensemble des éléments qu'il devait détecter ou reconnaître comme positif ($FN = 0$).

Lorsque le nombre d'instances dans les classes est déséquilibré, d'autres métriques sont proposées telles que :

Précision et rappel pondérés : ces métriques attribuent des poids à chaque classe en fonction de leur fréquence dans l'ensemble de données. Ainsi, les classes minoritaires ont un poids plus élevé que les classes majoritaires ;

F-mesure pondérée : cette métrique combine la précision et le rappel pondérés ;

Courbe ROC et AUC : la courbe ROC (*Receiver Operating Characteristic*) est définie dans le cas binaire. Elle représente la performance par le taux de vrais positifs en fonction du taux de faux positifs pour différents seuils de décision. Ce sont des métriques qui mesurent la capacité d'un modèle à discriminer les classes positives et négatives. L'AUC (*Area Under the Curve*) est l'aire en dessous de cette courbe ROC et elle quantifie la performance globale d'un modèle de classification binaire en considérant tous les seuils de classification possibles. Une AUC de 1 correspond à un modèle parfait, tandis qu'une AUC de 0,5 correspond à un modèle aléatoire ;

Courbe PR et AUC : la courbe PR (*Precision/Recall*) est définie dans le cas binaire. Elle met en relation la précision et le rappel pour différents seuils de décision. Là encore l'AUC est l'aire sous la courbe PR (AUC−PR) qui quantifie la performance globale d'un modèle de classification binaire en considérant tous les seuils de classification possibles. Une valeur d'AUC−PR de 1 signifie que l'évaluation est parfaite. La courbe PR est plus adaptée que la courbe ROC lorsque qu'il s'agit d'évaluer la capacité de l'architecture à détecter la classe minoritaire (la classe positive) car elle prend en compte le nombre de vrais positifs par rapport au nombre total de prédictions positives.

Conclusions

Pour conclure, dans ce chapitre nous avons introduit les concepts généraux qui seront manipulés dans les chapitres suivants. Cette introduction s'est déroulée en 5 parties en commençant par les différents types de modélisation et d'apprentissage qu'il peut exister dans le champ de recherche de l'apprentissage machine. Puis nous avons pu introduire la matière première des systèmes se basant sur les techniques d'apprentissage : les données. Ensuite nous avons abordé plus spécifiquement les architectures neuronales profondes et les différents types de combinaisons possibles. Enfin la dernière partie a été consacrée à la présentation de techniques d'évaluation que nous reprendrons dans nos travaux.

Chapitre 2

État de l'art

La reconnaissance d'activité humaine (HAR pour *Human Activity Recognition*) est un champ de recherche actif dans la communauté scientifique grâce notamment à la multiplication des capteurs tels que des caméras, les microphones, les capteurs embarqués dans les téléphones portables et l'apparition sur le marché des nombreux autres objets connectés ainsi que grâce aux nombreuses applications possibles dans les domaines de la santé [48, 115], de la sécurité/sûreté [29, 75, 189, 124] et de l'interaction homme/machine [78, 134]. Grâce à tous ces capteurs, il est possible de décrire par des signaux très différents un grand nombre d'activités humaines, signaux dont l'analyse du contenu permet de les reconnaître, de les détecter ou encore de les localiser.

Ce chapitre se concentre sur l'analyse des signaux audio et vidéo pour remplir une tâche de HAR et plus particulièrement de la détection d'un comportement humain violent. Nous commencerons par présenter l'ensemble des bases de séquences d'images et de signaux audio utilisées pour apprendre et évaluer les modèles sous-jacents. Puis, nous présenterons un état de l'art des méthodes de HAR fondées sur l'analyse du contenu d'une séquence d'images, sur celle du signal audio et sur l'analyse conjointe de ces deux signaux. La dernière partie introduira la problématique de reconnaissance de la violence pour laquelle nous avons proposé des architectures neuronales profondes.

2.1 Jeux de données de la communauté

Les jeux de données sont indispensables en apprentissage machine pour la mise en œuvre de modèle statistique, tant pour les estimations que pour les évaluations. Ainsi, avant de présenter un état de l'art des méthodes de HAR ou de reconnaissance d'évènements sonores, nous proposons de présenter brièvement les bases de données qui seront utilisées dans les futures sections.

Les principaux jeux de données proposés dans le domaine de la reconnaissance d'activité humaine sont constitués du signal vidéo : [141, 86, 148, 80, 69, 142, 61, 81]. La reconnaissance d'activité humaine n'est pas définie en tant que tel dans les champs de recherche du signal sonore. Parmi ces derniers, les plus proches sont ceux de la reconnaissance de scènes acoustiques, de motifs ou d'évènements sonores, dans lesquels les évènements à reconnaître ne sont pas forcément liés à une action humaine particulière : [150, 51, 49, 105, 38, 53, 44, 161, 50]. Les études portant sur l'analyse conjointe du signal audio et vidéo sont beaucoup moins répandues, d'où un nombre de jeux de données multi-modales moins vaste. Nous pouvons citer dans ce contexte [157, 13]. L'ensemble de ces jeux de données sont repris dans le tableau 2.1. Nous y retrouvons les principales caractéristiques comme la nature de son(ses) signal(aux) le(les) constituant, le type d'annotation associée (Tel que définie en section 1.2.2). Une présentation détaillée de ces jeux de données est fournie en Annexe A.

Nom		Année	Audio	Vidéo	Annotation	Nb. classes	Nb. classes violences	Nb. exemples	Durée des exemples	Durée totale
RWCP	[110]	2000	✓		Observation	3	0	100	-	-
ESC-10 / ESC-50	[128]	2015	✓		Observation	50	1	2.000	5s	2h46min
DCASE2013-Event	[150]	2015	✓		Temporel	16	0	398	1min	6h38
CHiME-Home	[51]	2015	✓		Observation	7	0	6.137	4s	6h49
MIVIA AED	[49]	2015	✓		Temporel	3	3	580	3min	28h28
TUT Sound Events	[105]	2016	✓		Temporel	18	0	42	3-5min	1h53
SINS	[38]	2017	✓		Temporel	16	0	1.095	-	1937h
AudioSet	[53]	2017	✓		Observation (moins précise)	527	4	2.084.320	10s	5.800h
CURE	[44]	2019	✓		Observation (moins précise)	13	0	7.000	5s	9h43
NIGENS-SE	[161]	2019	✓		Temporel	14	4	1.017	1s-5min	4h45
USM-SED	[2]	2021	✓		Observation (moins précise)	27	2	20.000	5s	27h46
FSD50K	[50]	2022	✓		Observation (moins précise)	200	4	51.197	0.3-30s	108h
KTH	[141]	2004		✓	Observation	6	1	599	8s-1min / ≈ 19s	3h13
HMDB51	[86]	2011		✓	Observation	51	7	6.849	0.63s-35s / ≈ 3,14s	5h55
UCF-101	[148]	2012		✓	Observation	101	5	13.320	1s-1min10s / ≈ 7,20s	27h
Sports1M	[80]	2014		✓	Observation (moins précise)	487	66	1.133.100	1s-10h / ≈ 4min8s	100.000h
ActivityNet	[69]	2015		✓	Temporel	200	8	19.994	1s-16min / ≈ 1min56	648h
Charades	[142]	2016		✓	Observation	203	0	9.848	2s-3min14 / ≈ 29s	81h15
EPIC-KITCHENS	[35]	2018		✓	Spatial	472	0	432	-	55h
AVA	[61]	2018		✓	Spatial	80	2	430	15min	107h
Kinetics	[81, 17, 18, 146]	2019		✓	Exemple	700	8	650.000	10s	1800h
AVE	[157]	2018	✓	✓	Temporel	28	0	4.143	10s	11h30
AVECL-UMONS	[13]	2020	✓	✓	Temporel	11	0	5.386	3s & 4s	21h

TABLE 2.1 – Listes de jeux de données dédiés à la reconnaissance d'actions et d'évènements sonores et leurs principales caractéristiques (Les "-" indiquant que les informations ne sont pas communiquées).

Ayant servi à de nombreux développements de modèles de référence en HAR ou en reconnaissance sonore, ces jeux de données considèrent peu de classes appartenant à la définition d'une violence. Des jeux de données spécifiques ont donc été développés lors d'études traitant de reconnaissance d'actions violentes avec le signal audio [49], avec le signal vidéo [114, 67, 136, 153, 147, 124, 189, 22] ou avec le signal audio et vidéo [42, 180]. Le tableau 2.2 détaille ces jeux de données. Comme précédemment, nous y retrouvons les principales caractéristiques comme la nature de son(ses) signal(aux), le type d'annotation associée, etc. et une présentation détaillée de ces jeux de données est également fournie en Annexe A.

2.2 Reconnaissance d'activité humaine par analyse vidéo

Les premiers travaux sur la HAR ont proposé des méthodes qualifiées d'*handcrafted*, c'est-à-dire fondées sur l'extraction de caractéristiques et de descripteurs dans les images à partir desquels il était ensuite possible d'estimer un modèle capable de reconnaître l'activité présente dans la séquence. Parmi ces caractéristiques, nous pouvons citer par exemple : STIP, SIFT et MoSIFT. L'étape de classification est assurée par des approches telles que GMM, SVM, HMM ou K-Means [91, 21, 170]. Bien que ces méthodes aient offert

Nom		Année	Environnement	Audio	Vidéo	Annotation	Nb. classes	Nb. exemples	Durée des exemples	Durée totale
MIVIA Audio Events Dataset	[49]	2015	Surveillance	✓		-	3	42	3-5min	1h53
Hockey Fight	[114]	2011	Match de hockey		✓	Observation	2	1 000 vidéos	2s	33min20s
Violent-Flow - Crowd Violence	[67]	2012	Divers		✓	Observation	2	246 vidéos	1,04-6,52s / ≈ 3,60s	14min09s
RE-DID	[136]	2015	Divers		✓	Spatial	2	73 (30 vidéos)	20s-4min / ≈	Non disponible
UCF-Crime	[153]	2018	Surveillance		✓	Observation (moins précis)	14	1900	60s-10min / ≈	128h
RLVS	[147]	2019	Divers		✓	Observation	2	2000	3s-6min16 / ≈ 5s	2h55
CCTV-Fights	[124]	2019	Surveillance		✓	Temporel	2	1000 vidéos	5s-12min	18h
Surveillance Camera Fight	[189]	2019	Surveillance		✓	Observation	2	300 vidéos	2s	10min
RWF-2000	[22]	2021	Surveillance		✓	Observation	2	2000 vidéos	5s	2h46
VSD	[42]	2011-2015	Films	✓	✓	Temporel	10	1 317	55,3s-13min49 / ≈ 3,93s	2h43min
XD-Violence	[180]	2020	Divers	✓	✓	Observation (moins précis)	6	4754 vidéos	variables	217h
BOSS	[90]	2009	Ferroviaire	✓	✓	Temporel	-	-	variables	
Yang *et al.*	[182]	2009	Ferroviaire	✓	✓	Temporel	-	-	variables	1h30

TABLE 2.2 – Listes des jeux de données dédiées à la reconnaissance de violences et leurs principales caractéristiques (Le "-" indiquant que l'information n'est pas communiquée)

des résultats intéressants sur des bases d'images de référence, elles sont très difficiles à généraliser, c'est-à-dire à appliquer à d'autres environnements, à d'autres points de vue, en présence d'occultations et d'autres personnes [140, 185].

Pour répondre à ces limites, des approches basées sur l'apprentissage profond ont été récemment proposées. Ces architectures profondes montrent des résultats prometteurs en matière de reconnaissance d'activité humaine et exploitent des réseaux de neurones convolutifs, des réseaux de neurones récurrents (RNN) pour extraire des caractéristiques d'une séquence temporelle ou encore des réseaux de neurones multi-branches. Les RNN à mémoire à court terme (LSTM) et les RNN à portes (GRU) sont des variantes qui ont montré des performances remarquables pour la reconnaissance d'activité humaine par analyse vidéo. Dans la suite de cette section, nous présentons un état de l'art de ces éléments d'architecture.

2.2.1 Les approches convolutionelles 2D

L'introduction des architectures neuronales profondes prenant en entrée des représentations brutes ou pré-traitées a permis d'améliorer les performances de nombreuses tâches de reconnaissance ou de classification. Bien qu'elles nécessitent une très grande quantité de données, l'intérêt d'une architecture neuronale profonde est de laisser le modèle extraire les caractéristiques les plus pertinentes à la prise de décision et d'accroître ainsi ses capacités de généralisation.

Karpathy *et al.* dans [80] proposent d'exploiter une architecture neuronale profonde fondée sur des couches de convolution 2D s'inspirant du travail de [85] dans le cadre du challenge *ImageNet*. Cette architecture est évaluée sur la base de vidéos *Sports1M* composée d'un million de vidéos annotées selon 487 classes. Cette architecture de base a permis aux auteurs d'évaluer la contribution de l'apparence contenue dans une image sur une tâche de reconnaissance d'activité humaine. Comme illustré dans la figure 2.1, les auteurs proposent de combiner les caractéristiques extraites pour plusieurs images selon plusieurs stratégies (*early*, *late* et *slow*) pour apprendre des motifs spatio-temporels des actions humaines considérées. Une première évaluation des architectures proposées sur *Sports1M* a montré qu'elles permettaient de capturer des motifs spatio-temporels permettant d'identifier ces actions et que la combinaison *slow* fournissait les meilleurs résultats. L'architecture *slow* a donc été utilisé pour vérifier sa généralisation à d'autres contextes telle que celui de la base *UCF-101*. L'apprentissage à partir de zéro des paramètres de l'architecture qu'ils proposent ne permet pas de faire de grands progrès sur le jeu de données *UCF-101* par rapport aux performances des modèles de type SVM avec des représentations de haut-

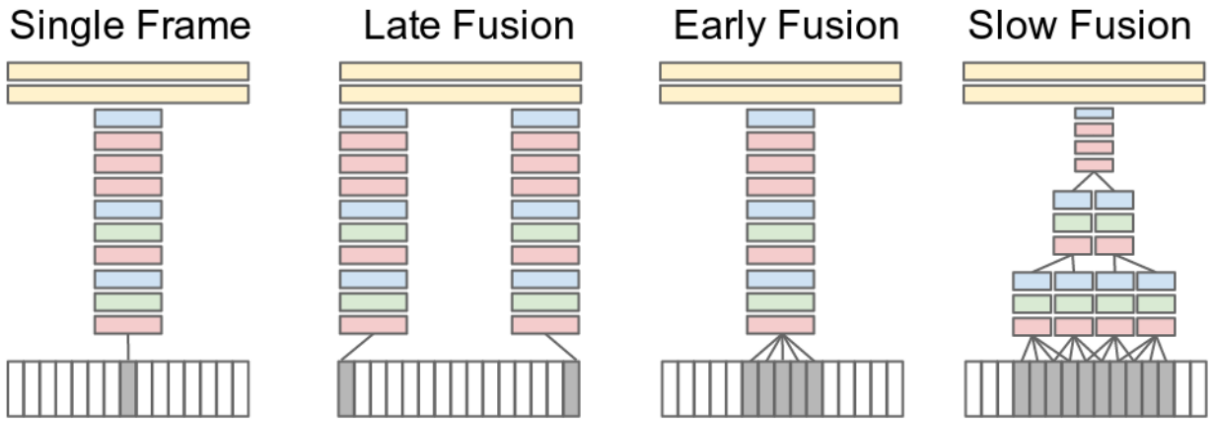


FIGURE 2.1 – Approches proposées par Karpathy *et al.* dans [80] pour reconnaître des activités humaines à partir de couches de convolutions 2D.

niveau des données [148]. Cependant, le réseau pré-entraîné sur *Sports1M* et transféré sur le jeu de données *UCF-101* montre des performances largement améliorées tout particulièrement lorsque les caractéristiques spatio-temporelles des premières couches apprises sur *Sports1M* sont gardées et que seules les 3 dernières couches sont ajustées sur *UCF-101* (fine-tuning). Il faut noter que dans ce travail, les auteurs proposent d'analyser le contenu de chaque image à plusieurs résolutions en définissant une architecture neuronale à deux branches. L'entrée de la première branche est constituée de l'image dont la résolution a été divisée par 2 et celle de la deuxième branche est un découpage centré sur le centre de l'image dans sa résolution initiale, zone où se déroulent généralement les actions à identifier. Ce choix permet de réduire le temps de l'apprentissage tout en sauvegardant les performances.

Dans la continuité de l'étude de Karpathy *et al.* afin de prendre en compte la structure temporelle du contenu d'une séquence d'images, Ng *et al.* [112] expérimentent différents niveaux de regroupement, comme illustré dans la figure 2.2, ou considère une couche récurrente de type LSTM. Les caractéristiques sont extraites par une architecture convolutive 2D, basée sur l'architecture *AlexNet* ou *GoogleNet*. Tout d'abord, ils concluent qu'il est plus performant de combiner les caractéristiques avec une couche de regroupement par maximum juste après les couches convolutionnelles 2D (Figure 2.2 (a)), plutôt que d'utiliser quelques couches entièrement connectées avant de combiner les caractéristiques (Figure 2.2 (b, c, d)). Ensuite, ils observent que l'architecture *GoogleNet*, qui implémente des modules *Inception*, est plus performante que l'architecture *AlexNet*. Enfin, ils expérimentent une dernière architecture multi-branches où une branche traite la séquence d'images et une nouvelle branche traite une séquence de flot optique. À cette architecture à deux branches, une couche récurrente de type LSTM est ajoutée. Cette dernière architecture est la version la plus performante. Ce type d'architecture multi-branches est détaillé dans la section ci-dessous.

2.2.2 Les approches multi-branches 2D

Les architectures dites multi-branches vidéo, traitent en parallèle et de manière différente le même signal et/ou le signal ayant subi des pré-traitements différents. L'approche la plus populaire s'inspire du cortex visuel humain qui traite l'information par deux branches [59], une branche ventrale qui traite l'information sémantique et une branche dorsale qui traite l'information de mouvement. Les architectures proposées sont composées d'une branche qui traite une séquence d'images en couleur et d'une branche qui traite une séquence du flot optique.

Simonyan et Zisserman dans [143] proposent en 2014 unes des premières architectures neuronales profondes multi-branches qui traite en parallèle une image et une séquence du flot optique pour faire de la reconnaissance d'action. L'intérêt de combiner ces deux représentations du signal vidéo est de saisir à la fois l'apparence contenue dans les images et l'information de mouvement. Comme illustré dans la figure 2.3, dans l'architecture qu'ils proposent, les deux branches sont une succession de couches de convolutions 2D. Chacune des branches fournit un score pour chacune des actions pour lesquelles elles ont été entraînées. Le score final est calculé en les combinant par une moyenne ou par une méthode plus complexe telle qu'un SVM.

Dans la continuité des travaux de Simonyan et Zisserman, Wang *et al.* [171, 172] proposent une architecture multi-branches plus profonde basée sur l'architecture *VGG16* [144]. Afin d'assurer la modélisation des structures temporelles étendues, les auteurs proposent une approche appelée *Temporal Segment Network* (TSN) qui découpe une vidéo en petits segments de même durée comme illustré sur la figure 2.4. Chaque segment est analysé indépendamment par la même architecture multi-branches qui prend ainsi en en-

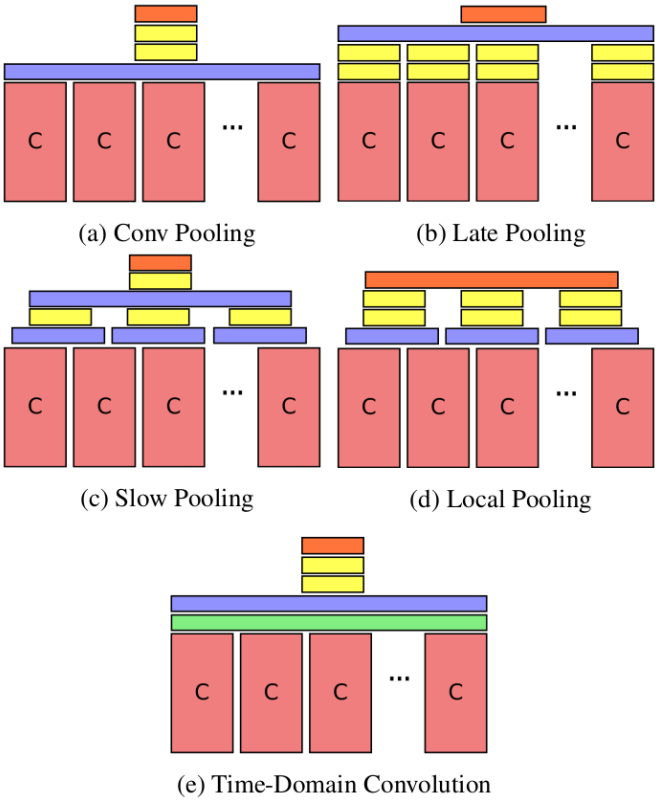


FIGURE 2.2 – Architectures pour tenir compte de la structure temporelle du contenu d'une séquence d'images proposées dans [112]. Les couches de convolutions empilées sont désignées par "C". Les rectangles bleus, verts, jaunes et orange représentent respectivement les couches de *max-pooling*, de convolution 1D, entièrement connectées et de *softmax*.

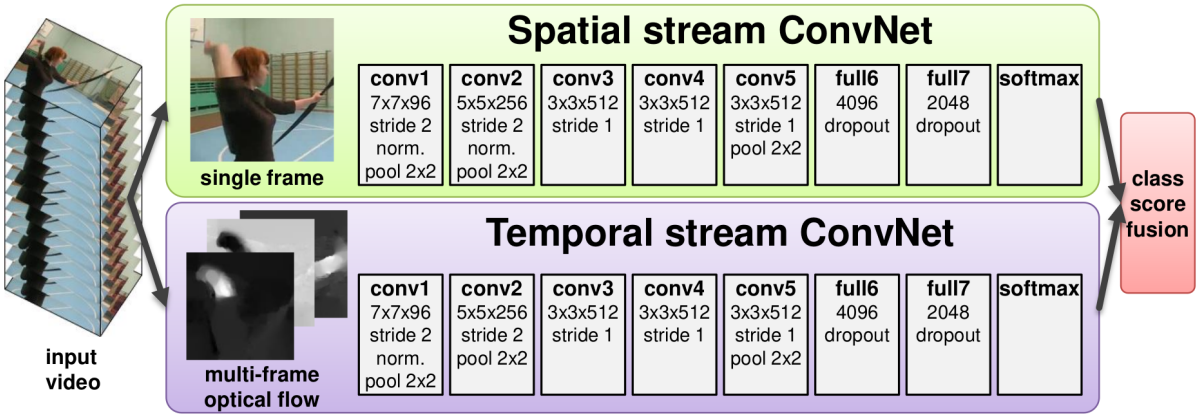


FIGURE 2.3 – Approches multi-branches proposée par Simonyan et Zisserman dans [143].

trée une image sélectionnée aléatoirement sur le segment et les autres modalités qui lui correspondent telle que son flot optique calculé à partir de son image précédente. Les paramètres de l'architecture sont ainsi partagés entre tous les segments ce qui permet de réduire le coût d'apprentissage en préservant suffisamment d'information. Chaque image et modalités correspondantes extraites de tous les segments sont alors agrégées pour décider sur la présence ou non d'une des actions d'intérêt. Plusieurs fonctions d'agrégation ont été évaluées et les auteurs montrent qu'en plaçant en entrée du réseau un échantillonnage de la séquence entière, cette nouvelle architecture est capable de modéliser des structures aux temporalités longues et dépasse les performances de l'état de l'art sur les 4 bases *HMDB51*, *UCF-101*, *THUMOS14*, *ActivityNet*.

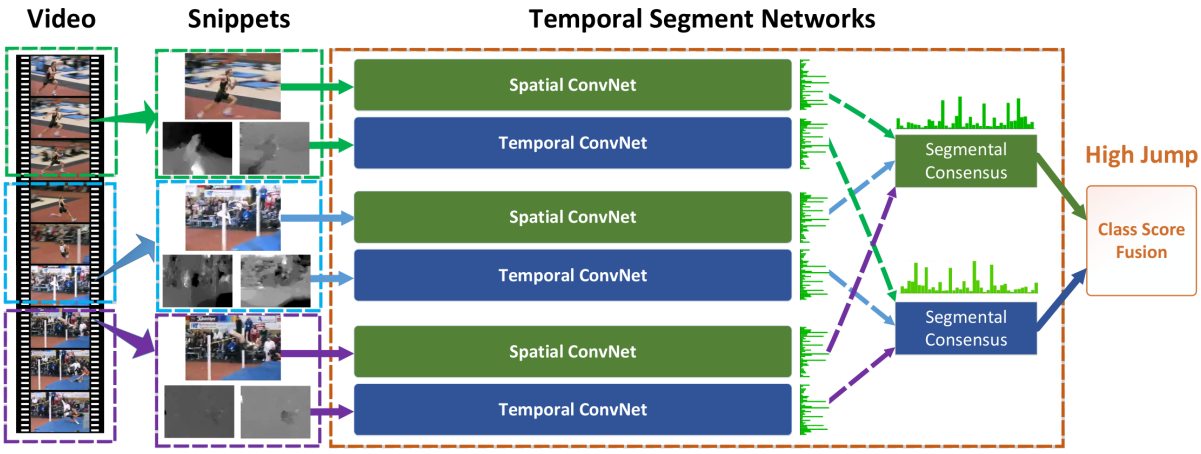


FIGURE 2.4 – Cadre *Temporal Segment Network* proposé par Wang *et al.* dans [172].

Par la suite, cette approche a été améliorée dans les travaux [186] et [96] pour rechercher une stratégie de combinaison non plus des scores comme dans les travaux précédents, mais des espaces de caractéristiques. Les auteurs opèrent avec des couches entièrement connectées appliquées sur différentes durées temporelles ou avec des couches de convolutions 1D.

Dans [101], Ma *et al.* proposent une étude approfondie sur l'utilisation de couches récurrentes de type LSTM ou de couches de convolutions 1D, permettant à l'architecture multi-branches de modéliser plus finement et sur des durées plus ou moins longues les structures temporelles des actions. Comme illustré dans la figure 2.5, dans l'architecture qu'ils proposent, les caractéristiques extraites des séquences d'images et celles extraites des flots optiques correspondants sont concaténées et envoyées à une couche récurrente de type LSTM. Tout d'abord, les résultats qu'ils obtiennent montrent que des couches de convolutions 1D (encapsulées sous forme d'un module *Inception*) permettent de modéliser les temporalités de manière aussi efficace que les couches récurrentes de type LSTM. Ils montrent également que le traitement de la vidéo avec l'architecture multi-branches et une couche LSTM, n'est pas beaucoup plus performante que l'architecture TSN sans LSTM [172].

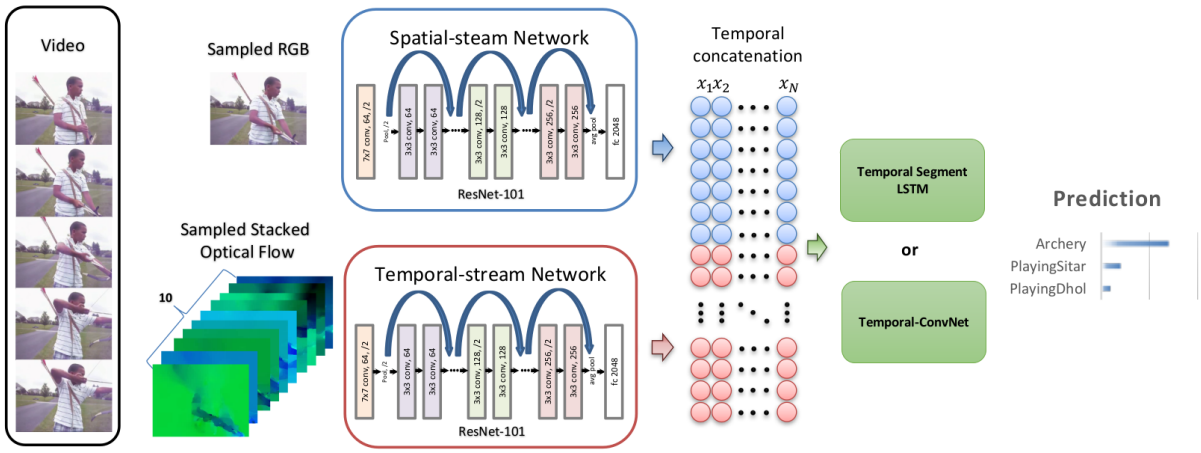


FIGURE 2.5 – Approche multi-branches avec LSTM proposée par Ma *et al.* dans [101].

Des architectures multi-branches décrites dans la littérature exploitent d'autres modalités que celle de l'image brute ou du flot optique. Dans [23], les auteurs exploitent conjointement les images d'une séquence et les squelettes qui y sont extraits. [174] associe des objets détectés sur des images et [187] exploite la carte des disparités correspondante (RVB-D). De plus, les approches multi-branches peuvent également combiner des signaux différents, par exemple le signal vidéo et le signal audio, nous détaillons ces approches dans la section 2.4 dédiée à la combinaison de la vision et de l'écoute.

Alors que dans cette section, nous avons présentés des architectures acceptant en entrée des données 2D tel qu'une image brute ou une image du flot optique, dans la suite de ce chapitre, nous présentons les approches capables d'analyser un volume de données grâce à des convolutions 3D. Elles constituent désormais une partie importante dans les architectures multi-branches profondes récentes.

2.2.3 Les approches convolutionnelles 3D

Ces approches permettent d'analyser un volume de données et notamment celui constitué par une succession des images d'une séquence vidéo. Les couches de convolutions possèdent alors des filtres à 3 dimensions permettant de prendre en compte les mouvements contenus dans la séquence.

Ces techniques ont déjà été expérimentées dans plusieurs travaux [83, 156, 7, 79], cependant le manque de grands jeux de données ne permettait pas d'estimer correctement les paramètres de ces réseaux de neurones convolutifs 3D. Ainsi, plus récemment Tran *et al.* dans [158], avec leur architecture *C3D*, ont montré que les filtres 3D pouvaient apprendre des motifs spatio-temporels. L'architecture qu'ils proposent est inspirée de l'architecture *VGG16* où les filtres 2D sont remplacés par des filtres 3D. Compte tenu du nombre de paramètres que possède ce type d'architecture, l'apprentissage du réseau a été assuré grâce au jeu de données à large échelle *Sports1M* qui a permis, dans un premier temps, de pré-estimer les paramètres de l'architecture. Dans un second temps, ces paramètres ont été ajustés sur le jeu de données *UCF-101*. Les résultats obtenus ont montré que l'architecture était pertinente pour faire de l'extraction de caractéristiques génériques, caractéristiques pouvant être utilisées dans de nombreuses tâches. Au-delà des simples performances, il a aussi été montré expérimentalement qu'un filtre $3 \times 3 \times 3$ est la taille la plus adaptée.

Dans la continuité de ces travaux, Varol *et al.* [164] se sont intéressés au nombre optimal d'images successives d'une séquence à fournir à l'entrée d'une architecture à base de couches de convolutions 3D dans le cadre de la reconnaissance d'actions. Ces travaux se basent sur l'architecture *C3D* et évaluent 5 valeurs différentes : 20, 40, 60, 80 et 100 images. Les résultats de cette étude montrent expérimentalement qu'une entrée plus longue (100 images) permet d'avoir de meilleures performances.

Pour traiter la problématique de la taille du jeu de données nécessaire pour entraîner ce type d'architecture, Carreira *et al.* dans [19] proposent l'architecture nommée *I3D*, illustré dans la figure 2.6 en proposant d'ajouter une troisième dimension au module *Inception* (*cf.* section 1.3). L'architecture présentée est une architecture simple branche ou multi-branches. L'architecture simple branche prend en entrée une séquence d'images RVB. La version multi-branches complète l'entrée par une séquence de flots optiques. Les paramètres de la branche traitant la séquence d'images RVB sont pré-estimés sur le jeu de données *ImageNet* [138] en 2D. Les paramètres des filtres 3D sont initialisés en répliquant les paramètres 2D évitant ainsi une initialisation aléatoire. Pour l'ajustement final des paramètres, les auteurs utilisent le jeu de données *Kinetics-400* qu'ils introduisent. Les performances obtenues par cette nouvelle architecture et par la stratégie de pré-estimation des paramètres ont fait de cette architecture une référence pour la tâche de reconnaissance d'actions.

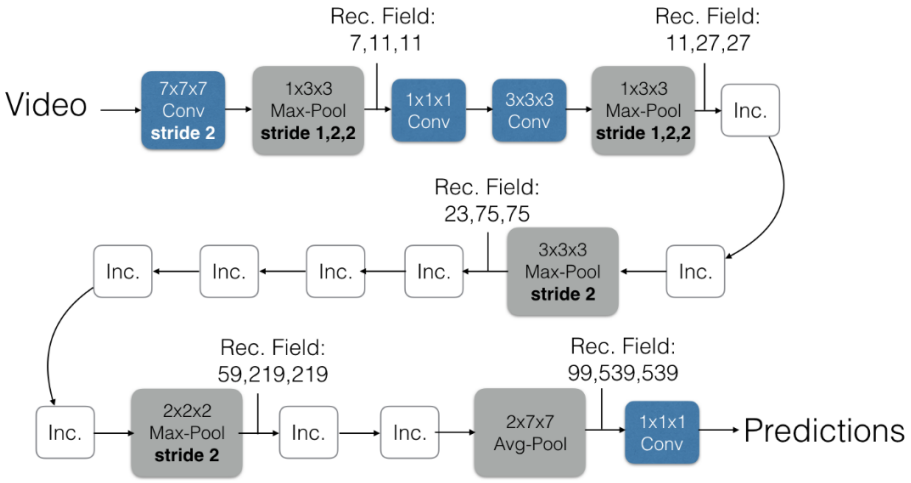


FIGURE 2.6 – Architecture *I3D* proposée par Carreira *et al.* dans [19].

Cette architecture est reprise dans nos travaux. Pour la présenter plus précisément, elle prend en entrée N images RVB ($N \geq 16$) successives échantillonnées à 25 images par seconde. Avant d'entrer dans le modèle *I3D*, les images sont redimensionnées en 224×224 et normalisées. Dans le modèle, avant d'atteindre le premier module *Inception* trois couches de convolutions et deux couches de *max-pooling* sont appliqués alternativement afin de réduire la dimension des données fournies en entrée. Ensuite neuf modules *Inception* en trois groupes séparés par deux couches de *max-pooling* : la première après les 2 premiers modules *Inception* et la seconde après les 5 modules *Inception* suivants. Enfin, pour prendre la décision après le dernier module *Inception* une couche d'*average pooling* et une couche de convolutions $1 \times 1 \times 1$ sont ajoutées. Cette tête de classification, inspirée des travaux de [97], est utilisé à la place d'une couche entièrement connectée afin de réduire le nombre de paramètres de cette dernière couche et ainsi le sur-ajustement.

Par ailleurs, comme pour les modules *Inception* l'extension à des filtres à 3 dimensions des blocs résiduels a également été proposée pour faire de la reconnaissance d'actions [65, 159, 66]. Cependant, les gains en performances sont apparus limités.

Afin de réduire le nombre de paramètres de ces architectures, des travaux [130, 160] se sont intéressés à la factorisation des couches de convolutions 3D (Figure 2.7 (a)) en deux couches de convolutions (Figure 2.7 (b)) : une couche de convolutions spatiales en 2D qui applique un filtre ($1 \times 3 \times 3$) et une couche de convolutions temporelles en 1D qui applique un filtre ($3 \times 1 \times 1$). Appliquées sur les jeux de données *Sports1M*, *Kinetics*, *UCF-101*, et *HMDB51*, ces architectures atteignent des résultats comparables ou supérieurs à l'état de l'art.

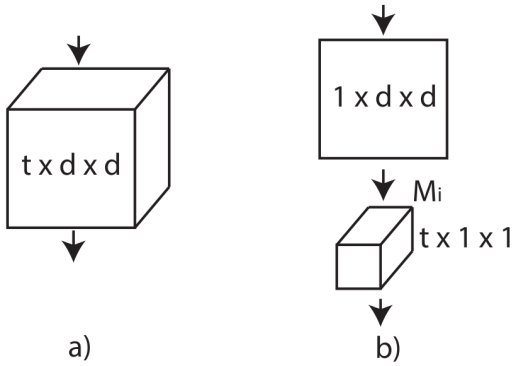


FIGURE 2.7 – Approche de factorisation de couches de convolutions 3D proposée par Tran *et al.* dans [160].

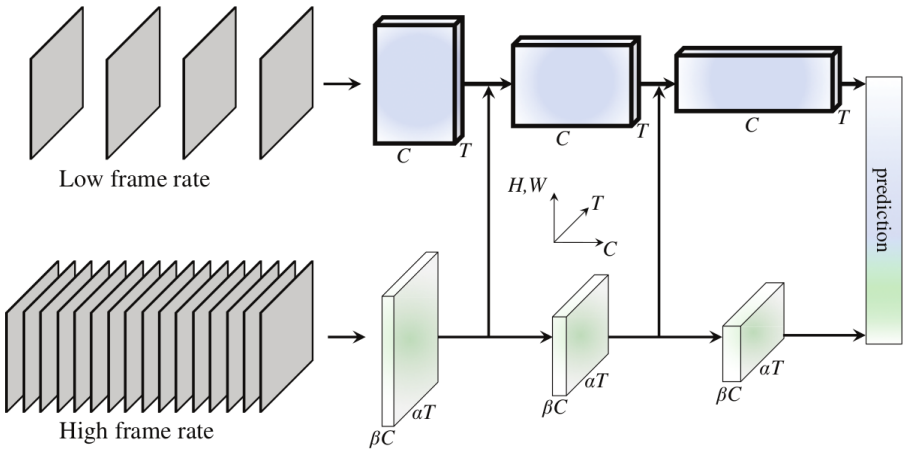


FIGURE 2.8 – Approche multi-branches proposée par Feichtenhofer *et al.* dans [47].

Par ailleurs, comme pour les architectures avec des couches de convolutions 2D, les architectures avec des couches de convolutions 3D peuvent être utilisées avec des couches récurrentes. L'objectif est de modéliser des motifs temporels de durées courtes avec des couches de convolutions 3D et de durées plus longues avec des couches récurrentes. Wang *et al.* dans [173] étudient la combinaison de l'architecture *I3D*, dont les paramètres sont pré-estimés sur le jeu de données *Kinetics-400*, à laquelle ils rajoutent une couche récurrente de type LSTM. Dans leur étude, les paramètres de l'architecture globale (*I3D*+LSTM) sont estimés. Les résultats montrent qu'il est intéressant de modéliser les deux types de temporalité.

Enfin, afin de limiter l'explosion du coût de calcul des architectures à base de convolutions 3D, Feichtenhofer *et al.* dans [47] proposent une architecture multi-branches pour traiter une séquence d'images de deux manières différentes, comme illustré dans la figure 2.8. Ces deux branches sont nommées "slow" et "fast" dans l'architecture *SlowFast Networks*. Elles sont conçues pour capturer des informations à différentes échelles temporelles et spatiales. La branche "slow" a une résolution temporelle réduite par sous-échantillonnage et une résolution spatiale élevée. Elle prend en compte les informations sémantiques de la scène. Cela est réalisé en utilisant une série de couches de convolutions 3D avec un noyau de taille $1 \times 5 \times 5$ pour agréger les informations spatiales. La branche "fast" a une résolution temporelle élevée et une résolution spatiale réduite. Elle prend en compte les mouvements dans la vidéo en utilisant une série de couches de convolutions 3D avec un noyau de taille $3 \times 3 \times 3$ pour capturer les informations temporelles. Les deux branches sont fusionnées en ajoutant les sorties des deux branches et en les normalisant par un facteur d'échelle adaptatif qui est appris pendant l'entraînement. De plus, l'architecture met en place des connexions latérales entre les deux branches permettant ainsi de prendre en compte les relations spatiales et temporelles à différentes échelles de mouvement. Cette approche fournit de très bonnes performances, notamment sur le jeu de données *Kinetics* tout en réduisant le coût de calcul.

2.2.4 Les stratégies d'apprentissage

Dans leur étude, Karpathy *et al.* dans [80] expérimentent les trois types de stratégies d'apprentissage (à partir de zéro, ajusté et fixé), présentées dans la section 1.1.3. Les paramètres de l'architecture sont pré-estimés sur le grand jeu de données *Sports1M* et ajustés sur le petit jeu de données *UCF-101*. La première conclusion de cette étude confirme qu'il n'est pas intéressant de partir d'une initialisation aléatoire des paramètres lorsque le jeu de données à disposition n'est pas grand. Ensuite, la deuxième conclusion est qu'il est

plus intéressant de fixer une partie de l'architecture que de chercher à ajuster les paramètres de toutes les couches sur un jeu de données plus petit. Enfin, la dernière conclusion montre qu'il est préférable de fixer les premières couches et d'ajuster que quelques couches avant la couche de décisions pour avoir une architecture plus performante sur un petit jeu de données. Au global, cette étude permet de conclure que les paramètres des premières couches apprises sur un grand jeu de données permettent d'extraire des caractéristiques pertinentes et généralisables pour être utilisées avec un plus petit jeu de données dans une autre tâche.

2.3 Reconnaissance sonore

Historiquement, le traitement du signal sonore s'est fortement développée pour la reconnaissance automatique de parole [131, 98, 11, 32, 26, 63]. Les études traitant le signal sonore se sont élargies ensuite à d'autres aspects comme la reconnaissance du locuteur, d'instruments de musique, de sentiments et d'une manière plus générale à des motifs sonores.

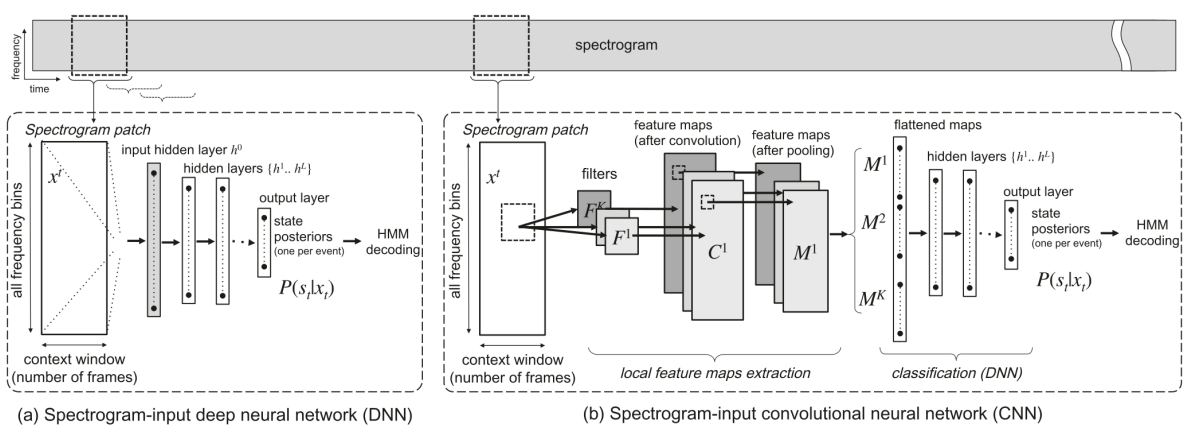
Les premiers travaux en reconnaissance d'événements sonores se sont appuyés sur les travaux de reconnaissance automatique de la parole. Ces techniques sont des approches d'apprentissage machine générative (GMM, HMM [104, 70], classifieur Bayésien) basée sur des représentations temps-fréquence comme le mel-spectrogramme ou des séquences de MFCC). Typiquement, un son se caractérise principalement par la distribution de ces fréquences dans le spectre. L'évolution au cours du temps de cette distribution caractérise un motif sonore. C'est pour cela que depuis de nombreuses années, les études sur la reconnaissance sonores s'appuient sur des séquences de coefficients spectraux (DSP, mel-spectrogramme, MFCC) pour caractériser des motifs sonores.

Par la suite, toujours basée sur la reconnaissance automatique de la parole, les architectures neuronales profondes associées à des représentations temps-fréquence ont fait leurs apparitions [73, 107]. Comme pour la communauté vision, les architectures neuronales profondes ont permis d'augmenter significativement le taux de reconnaissances et de détections. Des premiers travaux comme [84, 54, 46] se sont intéressés à faire de la reconnaissance d'événements sonores en utilisant des architectures à base de couches entièrement connectées. Ces premiers travaux montrent des résultats de classifications en hausse en comparaison à des méthodes à base de GMM+HMM.

Comme pour la reconnaissance d'activité humaine que nous avons présentée précédemment, la suite de cette section présente les principaux éléments des architectures de l'état de l'art.

2.3.1 Les approches convolutionelles 2D

Inspirés par la communauté image, certains travaux se sont intéressés à l'utilisation de couches de convolutions 2D pour diminuer la complexité des architectures [184, 127, 45, 126, 155] traitant de représentation à deux dimensions de type temps-fréquence (séquence de MFCC, mel-spectrogramme). Globalement, les architectures proposées sont composées de couches de convolutions 2D suivis de couches entièrement connectées. Les travaux de Espi *et al.*, dans [45], proposent en supplément d'ajouter un HMM (Figure 2.9) afin de modéliser l'évolution de temporelle des caractéristiques extraites. Ultérieurement, Takahashi et *et al.* dans [155] proposaient d'expérimenter les avancées de la communauté image en réduisant la taille des filtres (3×3) et augmentant la profondeur de l'architecture. Appliquées sur les jeux de données *RWCP*, *ESC-10*/*ESC-50*, *TUT Sound Events*, ces architectures de convolutions 2D traitant des données temps-fréquences sont devenues



(a) Spectrogram-input deep neural network (DNN)

(b) Spectrogram-input convolutional neural network (CNN)

FIGURE 2.9 – Approches proposées par Espi *et al.* dans [45].

significativement plus intéressantes que des architectures de couches entièrement connectées, ces dernières ne pouvant caractériser aussi finement des relations spectro-temporel à nombre de paramètres équivalent.

Par la suite, Hershey *et al.*, dans [72], ont évalué la pertinence des architectures provenant de la communauté image (*AlexNet*, *VGG*, *Inception*, *ResNet-50*) sur le signal audio. Ces architectures sont composées de couches de convolutions 2D incorporant les concepts des modules *ResNet* ou *Inception*. Les architectures sont entraînées sur le jeu de données *YouTube-100M* que Hershey *et al.* introduisent dans ces mêmes travaux. Les auteurs montrent que toutes les architectures de convolutions 2D obtiennent des performances de reconnaissance supérieures à des architectures de couches entièrement connectées, les architectures avec les modules *Inception* et *ResNet* étant les plus performantes.

Globalement, les architectures à base de couches de convolutions 2D capturent des structures communes sur la représentation audio comme pour leurs applications sur les images. La limite de ces travaux est que ces architectures à base de couches de convolutions 2D ne peuvent modéliser que des relations spectro-temporelles à court terme.

2.3.2 Les approches spectraux-temporelles

Parallèlement à l'intérêt porté aux couches de convolutions, certains ont porté leur intérêt sur les couches récurrentes de type RNN et LSTM [117, 175, 3] afin de prendre en compte les informations temporelles à long terme. Ces architectures cherchent à modéliser les temporalités des séquences des caractéristiques bas-niveaux (MFCC, mel-spectrogramme) en lieu et place des convolutions 2D vu précédemment (Figure 2.10). Appliquées sur les jeux de données de la communauté (*TUT Sound Events*, *TRECVID-MED*), ces architectures montrent que la modélisation explicite de la temporalité des séquences permet également d'obtenir des performances de reconnaissances supérieures à ceux obtenus avec des couches entièrement connectées.

Toutefois, comme Cakir *et al.* a pu le montrer dans [16], les architectures à base de couches de convolutions 2D présentent des résultats légèrement supérieurs à ceux obtenus par des architectures à base de couches récurrentes. Une explication avancée est que les couches récurrentes ne saisissent pas correctement l'information fréquentielle contenue dans les représentations temps-fréquence.

Afin de tirer parti des avantages des couches convolutionnelles 2D et récurrentes, des travaux [4, 16, 95] ont été rapidement proposés pour combiner ces deux formes d'architectures *Convolutional Recurrent Neural Network* (CRNN). Les architectures proposées sont donc composées dans un premier temps de couches convolutionnelles 2D [4, 16] ou 1D (filtres définis le long de l'axe temporel) [95] appliquées sur les représentations bas niveau

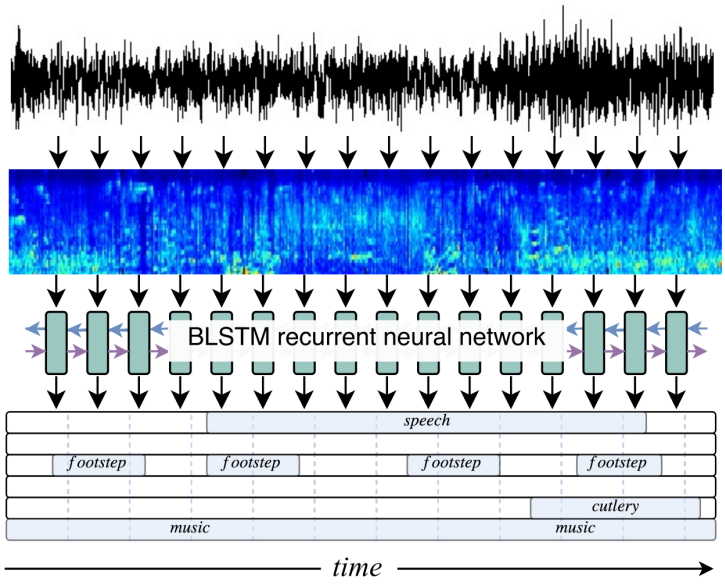


FIGURE 2.10 – Approches DNN *vs.* CNN proposées par Parascandolo *et al.* dans [117].

de type temps-fréquence du signal audio. Dans un second temps, les couches récurrentes sont appliquées sur les caractéristiques de plus haut niveaux extraites pas les couches convolutionnelles. Ainsi, grâce à la capacité de modéliser les relations spectro-temporelles à court terme (CNN) et leur évolution à plus long terme (LSTM ou GRU), ce type d'architecture a permis d'améliorer les taux de reconnaissance de motifs sonores (travaux présentés sur les bases de données *CHiME-Home*, *TUT Sound Events*, *TUT Rare Sound Events*). Aujourd'hui, ce type d'architecture fait référence sur de nombreuses tâches de reconnaissance sur le signal audio.

2.3.3 Vers une considération directe du signal temporel

Enfin, des approches plus récentes proposent d'appliquer des couches de convolutions 1D directement sur le signal sonore. C'est le cas des travaux de [34, 1] qui se sont intéressés à des tâches de reconnaissance d'environnement sonore (*UrbanSound8k*). L'intérêt majeur est de laisser l'architecture extraire des caractéristiques spécifique et pertinente à la tâche lors de l'apprentissage. Leur architecture est une succession de couches de convolutions 1D suivies de couches entièrement connectées. Ces études montrent que les filtres des premières couches de convolutions apprennent l'équivalent de filtres passe bande dont les fréquences centrales évoluent. Les résultats de classification obtenus sont proches des architectures de type *CRNN* mais avec moins de paramètres à estimer.

Une autre approche utilisant des convolutions 1D est apparue dans les travaux de [163] et [8] dédiés à des modèles génératifs du signal audio. Le principe de ce type de convolution 1D que l'on nomme aujourd'hui TCN (*Temporal Convolutional Networks*, est d'empiler des couches de convolution à une dimension en doublant le taux de dilatation à chaque nouvelle couche (Figure 2.11). La structure complète de *Wavenet* considère en plus une activation par "portes" et une connexion résiduelle à chaque couche. Ainsi, les travaux de [62] montrent que dans le cas de détection et de localisation de motifs sonores, ces approches permettent de faire la différence avec les approches par *CRNN* avec bien moins de paramètres. Ces résultats sont obtenus grâce notamment aux fonctions de dilatations successives qui permettent aux premières couches de modéliser des informations à court terme, et aux couches supérieures de modéliser des informations à plus long terme.

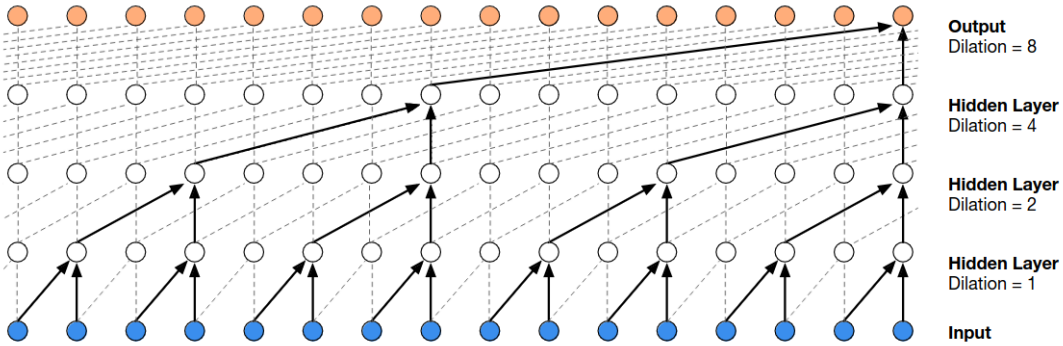


FIGURE 2.11 – Approche *Temporal Convolutional Networks* (*TCN*) proposée dans [163].

2.4 La combinaison de la vision et de l'écoute

En parallèle de ces travaux focalisés sur le traitement du signal audio ou vidéo, d'autres travaux proposent l'utilisation conjointe de ces derniers. Ces approches se retrouvent dans divers domaines comme la localisation de sources sonores [5, 116], la séparation de sources audio-vidéo [116], le rehaussement du signal parole [139, 106] ou encore la reconnaissance d'émotions [179]. Certains auteurs proposent même d'utiliser cette combinaison de signaux à des fins d'apprentissage de caractéristiques audio et vidéo [5, 6, 116].

Au vu du large éventail d'applications possibles combinant l'audio et la vidéo, nous aborderons cette section principalement avec des travaux dédiés à notre thématique de recherche de reconnaissance d'action. Avant cela, nous présenterons dans un premier temps des travaux portant sur l'apprentissage de caractéristiques audio et vidéo, puisqu'une partie nos travaux exploitera un de ces modèles.

2.4.1 Apprentissage conjoint de l'audio et la vidéo

Comme nous l'avons vu au travers des sections et chapitres précédents, l'estimation de modèles pour la reconnaissance d'évènements a couramment recours à l'utilisation d'extracteurs de paramètres (estimés préalablement via d'autres bases de données plus grandes). Ces extracteurs alimentent ensuite l'architecture dédiée à une tâche précise qui est estimée via une base de données dédiées et annotées. Pour apprendre ces extracteurs de paramètres, des travaux proposent d'exploiter les cohérences et corrélations entre signaux audio et vidéo dans des architectures multi-branches (une branche pour l'audio et une branche pour la vidéo). L'objectif est d'utiliser ces liens entre audio et vidéo pour apprendre des extracteurs avec peu ou indirectement pas d'annotations propres aux signaux (c'est-à-dire, pas d'utilisation d'annotations relatives au contenu précis des données utilisées).

Dans ce type d'apprentissage, nous trouvons les travaux de Aytar *et al.* dans [6] qui proposent l'architecture *SoundNet*. Orientée pour l'estimation d'un extracteur de paramètres audio, l'architecture *SoundNet* est estimée sans supervision selon une approche maître-élève via un signal vidéo synchrone : une branche maître (vidéo) traite une séquence d'images RVB et émet une prédiction qui est utilisée pour l'optimisation de l'architecture audio (branche élève, figure 2.12). Cette procédure d'apprentissage n'utilise aucune annotation (audio ou vidéo) mais repose sur un modèle vidéo (*AlexNet* ou *VGG*) dont les paramètres ont été précédemment appris.

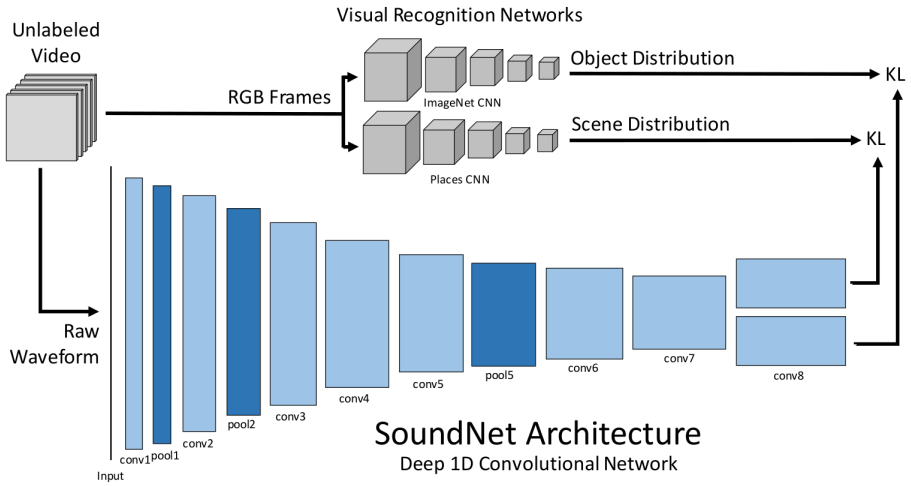


FIGURE 2.12 – Approche maître-élève proposée par Aytar *et al.* dans [6].

D'autres travaux comme ceux de Arandjelovic et Zisserman dans [5] ainsi que Cramer *et al.* dans [33] se sont intéressés à l'apprentissage conjoint d'extracteurs audio et vidéo sans annotation précise sur le contenu sonore ou vidéo. Ces travaux ont abouti à l'architecture *OpenL3* entraînée sur une tâche de correspondance audio-visuelle (Figure 2.13). Le modèle audio-visuel est composé d'une architecture audio et d'une architecture vidéo qui reçoivent en parallèle des signaux cohérents - lorsque les signaux audio et les séquences d'images sont issus de la même séquence audio/vidéo - et incohérents - lorsque le signal audio et les séquences d'images sont issus de séquences audio/vidéos différentes. L'objectif du modèle est de déterminer si le signal audio et la séquence d'images proviennent de la même source. L'optimisation du modèle se réalise en exploitant une annotation définissant la cohérence ou non des signaux. *OpenL3* a été entraînée sur un grand jeu de données *AudioSet* permettant d'apprendre des représentations nombreuses et variées. Les représentations sonores ont ensuite été utilisées pour apprendre des modèles sur des tâches de reconnaissance avec de plus petits jeux de données comme *UrbanSound8K*, *ESC-50* et *DCASE*.

L'architecture sonore d'*OpenL3* étant reprise dans nos travaux présentés dans le chapitre 3, nous en précisons ci-après les caractéristiques. *OpenL3* considère en entrée des segments de signal sonore de durée minimale de 1 seconde et d'une fréquence échantillonnage de 48kHz. Lorsque le signal présenté en entrée fait plus d'une seconde, celui-ci est découpé en N segments de 1s avec un chevauchement de 0,1s. Un mel-spectrogramme

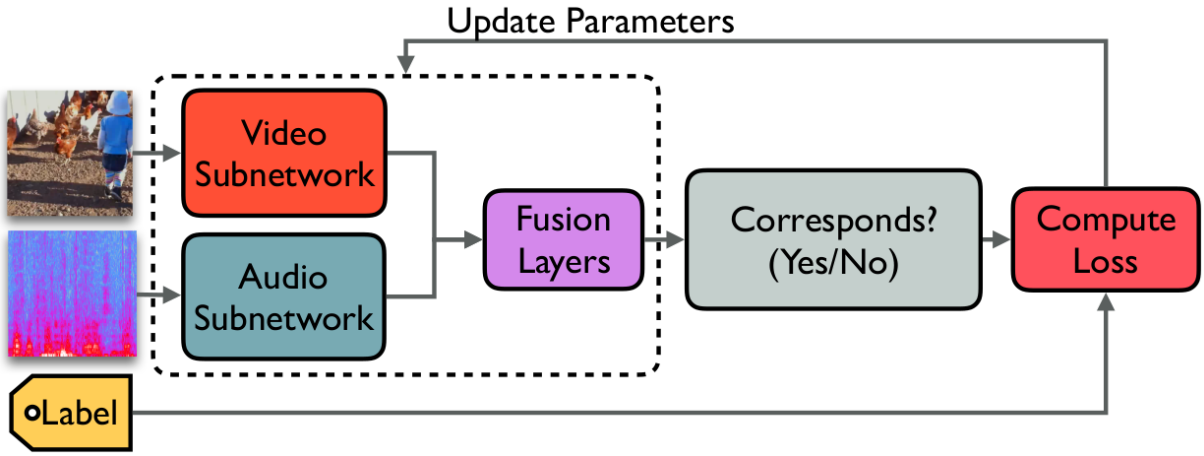


FIGURE 2.13 – Approche proposée par Arandjelovic *et al.* dans [5].

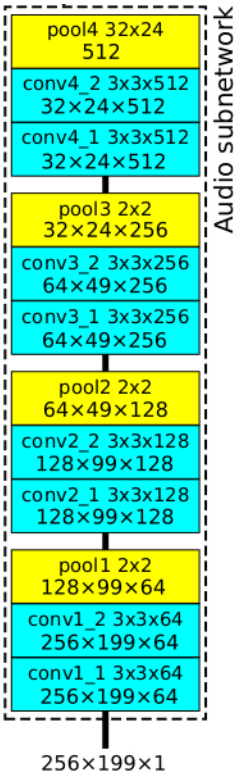


FIGURE 2.14 – Architecture *OpenL3* proposée par Cramer *et al.* dans [33].

est ensuite calculé pour chacun des N segments : chaque spectrogramme étant composé de 128 ou 256 coefficients fréquentiels, en fonction de la configuration, calculés sur une fenêtre temporelle de 40ms tous les 5ms. Chaque mel-spectrogramme est ensuite fourni au modèle *OpenL3* tel qu'il est présenté à la figure 2.14. Ce modèle est composé de 4 blocs, chacun composé de deux couches de convolutions et d'une couche de regroupement. Au final, la sortie de ce modèle est une projection de l'entrée qui permet une réduction de la dimension à un vecteur de 512 pour chacun des N segments.

2.4.2 Reconnaissance d'actions par audio et vidéo

Les premiers à s'intéresser à la combinaison de signaux audio et vidéo avec des architectures neuronales profondes pour le cadre de la reconnaissance d'action sont Wang *et al.*, dans [169]. Dans ces travaux, les auteurs proposent de combiner des spectrogrammes du signal audio, avec une séquence d'images RVB et une séquence de flots optiques. Comme indiqué à la figure 2.15, un pré-traitement est réalisé en parallèle sur chaque branche pour extraire de ces trois flux des caractéristiques de plus haut niveaux par l'intermédiaire d'architectures de type *VGG*. Ces caractéristiques sont ensuite combinées par concaténation, soit directement après des couches de convolutions de chaque entrée comme indiqué à la figure 2.15 (EF), soit après une série de couches entièrement connectées ajoutées sur chaque branche (LF). Les auteurs proposent ensuite de traiter les résultats de la combinaison par des couches entièrement connectées ou par un SVM. Les deux options sont mises en place par les auteurs pour chacun des niveaux de combinaison. La procédure d'apprentissage se déroule en deux temps sur une sous-partie du jeu de données *UCF-101* : les paramètres des extracteurs de caractéristiques sont d'abord estimés séparément et ensuite utilisés pour estimer conjointement les paramètres de l'architecture de combinaison et de décision. Globalement, les résultats montrent des résultats de reconnaissance plus élevés avec les différentes configurations multi-modales qu'avec des architectures uni-modales. Plus précisément, les auteurs observent que la combinaison EF est plus efficace lorsque

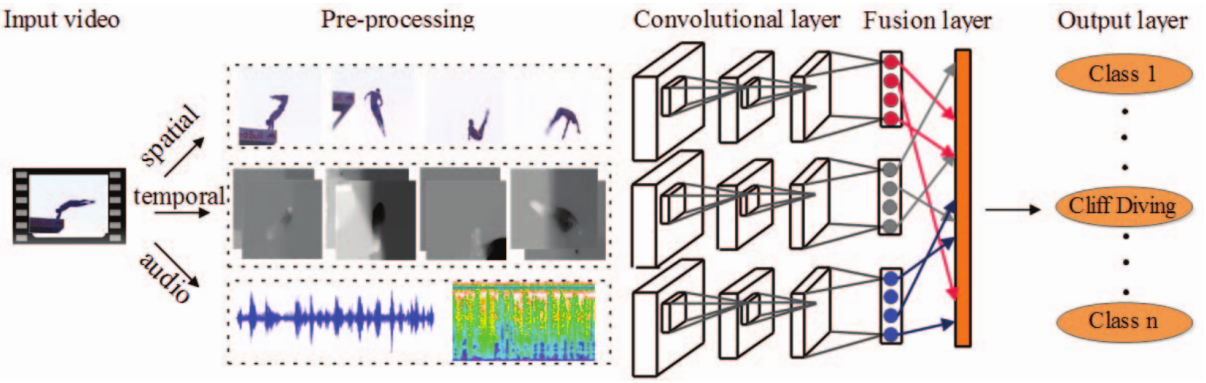


FIGURE 2.15 – Architecture proposée par Wang et al. dans [172].

ces combinaisons sont traitées par des couches entièrement connectées. La combinaison LF est quant à elle plus intéressante lorsque ces combinaisons sont traitées par un SVM.

Dans la continuité des travaux de Wang *et al.*, Kazakos *et al.* dans [82], reprennent l'architecture TSN [172] (présenté dans la section 2.2.2) et l'étendent avec l'ajout du mode audio. Les auteurs évaluent différents niveaux de combinaison (moyen ou tardif) ainsi que différentes stratégies de combinaison (concaténation ou mécanisme à portes). Les paramètres de ces architectures sont estimés sur le jeu de données *EPIC-Kitchens* qui permet de reconnaître des actions dans un environnement cuisine. Chaque action est annotée soit par un verbe (décrivant l'action) soit par des noms d'objets relatifs aux actions. En considérant une évaluation globale de reconnaissance (verbes+noms), les auteurs observent que l'architecture multi-modales est plus performante que les architectures uni-modales. Notamment, ils remarquent que la combinaison à un niveau moyen est plus performante que la combinaison à un niveau tardif. Enfin, en ce qui concerne la mise en œuvre de la combinaison, ils observent que celle par concaténation fournit de meilleurs résultats de reconnaissance que celle par mécanisme à portes.

En 2018, Owens et Efros dans [116], se sont intéressés à la combinaison du signal audio et d'une séquence d'images RVB pour différentes tâches dont celle de la reconnaissance d'action. Dans leurs travaux, ils ont évalué une architecture composée de couches de convolutions 3D pour traiter la séquence d'image RVB et de couches de convolutions 1D pour traiter l'onde sonore. Les sorties de ces couches sont combinées par concaténation et traitées ensuite par des couches de convolutions 3D. Les paramètres de leur architecture sont dans un premier temps estimés sur une tâche de reconnaissance de données audio et vidéo cohérentes ou incohérentes [1], comme pour les travaux qui introduisent l'architecture *OpenL3* précédemment présentée, sur une sélection aléatoire d'observation du jeu de données *AudioSet*. Les paramètres de l'architecture pré-estimés sont ensuite utilisés sur deux tâches : la reconnaissance d'actions et la séparation des sources audio. Pour la tâche de reconnaissance d'action, les paramètres de l'architecture sont ajustés sur le jeu de données *UCF-101* [148]. Concernant la tâche de séparation des sources audio, les paramètres de l'architecture sont ajustés sur le jeu de données *VoxCeleb* [109]. L'analyse qu'ils présentent montre que la pré-estimation des paramètres permet d'augmenter les performances tant pour la tâche de reconnaissance d'actions et que pour la tâche de séparation de sources audio.

Une autre approche pour exploiter le signal audio et vidéo est abordée par Tian *et al.* dans [157] qui proposent d'utiliser le signal audio pour "guider" le signal vidéo. L'architecture considérée extrait dans un premier temps des caractéristiques de la séquence d'images RVB à partir de l'architecture *VGG19* [144] et des caractéristiques du signal

1. L'objectif de cette tâche est de savoir si l'audio provient du même exemple que la vidéo. Dans la communauté anglophone, cette tâche est communément appelée *correspondence learning*.

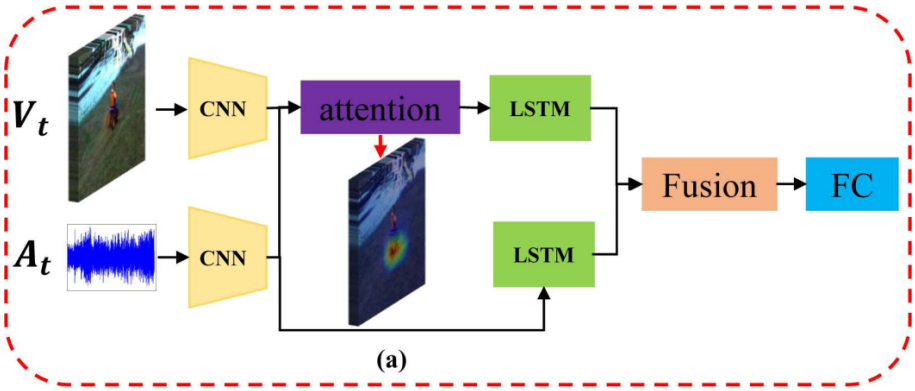


FIGURE 2.16 – Architecture proposée par Tian *et al.* dans [157].

audio à partir de l'architecture *VGGish* [72]. Une couche d'attention est ensuite appliquée sur la branche vidéo, contrôlée par la branche audio. L'objectif est de mettre en avant sur la branche vidéo les caractéristiques des évènements perçus sur le signal audio. Une couche récurrente de type LSTM est appliquée ensuite sur chaque branche pour modéliser la temporalité des signaux. Enfin, les sorties des couches récurrentes sont combinées par concaténation afin de prendre une décision finale. Les paramètres de cette architecture (hors extraction de caractéristiques) sont estimés sur le jeu de données *AVE* construit à cet effet (présenté en Annexe A.2). Les performances obtenues montrent que l'ajout de la couche d'attention permet de légèrement améliorer les performances par rapport à l'architecture n'implémentant pas la couche d'attention.

Dans la continuité de ces travaux, Brousmiche *et al.*, en 2020 dans [14], ont cherché à modéliser des interactions intra- et inter-branches (audio et vidéo) sur une tâche de reconnaissance d'événement. Les interactions entre les branches sont modélisées par attention croisée avec des couches à tête d'attention multiples (MHA). De plus, ils ont également évalué la modélisation temporelle des signaux avec des couches récurrentes de type LSTM selon plusieurs stratégies : une première utilise un LSTM unique pour les deux branches audio et vidéo, une deuxième utilise un LSTM pour chaque branche et enfin une troisième stratégie utilise un LSTM modifié, proposé par [133], qui consiste à considérer indépendamment les branches audio et vidéo à l'entrée du LSTM mais d'en partager la mémoire à long et à court terme. Les résultats sur le jeu de données *AVE* montrent que la couche d'attention croisée qui modélise les interactions entre les branches permet d'augmenter le taux de reconnaissance vis-à-vis d'une simple couche d'attention. Ensuite, leurs résultats montrent qu'utiliser un seul LSTM pour les deux branches (première stratégie) n'est pas une bonne stratégie car les évènements ne sont pas forcément perçus de manière équivalente et avec une même durée sur les deux signaux. En ce qui concerne les deux autres stratégies, l'utilisation de mémoires partagées du LSTM proposé par les auteurs (troisième stratégie) permet d'améliorer légèrement les performances par rapport à de simple LSTM placée en parallèle pour chaque branche.

2.5 Reconnaissance de violences

Dans le cadre de la reconnaissance d'action, nous nous intéressons maintenant au champ de recherche plus spécifique de la reconnaissance de violences, thématique au centre de l'application visée par le sujet de thèse.

En suivant la définition de l'Organisation Mondiale de la Santé (OMS), une violence est *l'utilisation intentionnelle de la force physique, de menaces à l'encontre des autres ou de soi-même, contre un groupe ou une communauté qui entraîne ou risque fortement d'entraîner des traumatismes psychologiques, des problèmes de développement physique*

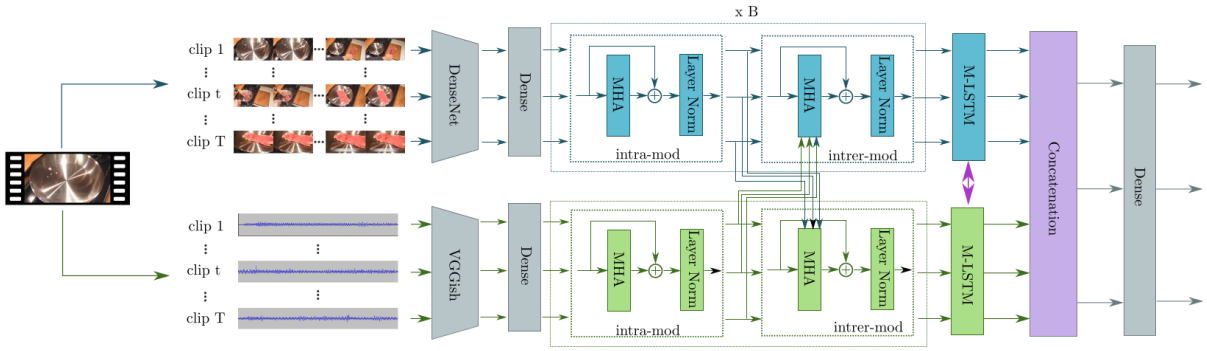


FIGURE 2.17 – Modélisation des interactions inter- et intra- branche proposée par Brousmiche *et al.* dans [14].

ou un décès[2]. Ce champ de recherche est donc très large dans sa définition. Les types d'actions régulièrement considérés dans la communauté s'appuyant sur cette définition sont : une bagarre, un feu, une poursuite en voiture, un coup de feu, une explosion, la présence de sang, la présence d'un cri, un accident ou encore une fusillade.

Des architectures capables de modéliser certains de ces types de violence ont été proposées, par exemple pour assurer la protection des enfants envers les scènes de violences dans les films, ou encore pour assurer la sécurité et la sûreté de certaines infrastructures recevant du public.

Dans cette section, nous faisons un point sur l'état de l'art des solutions proposées pour répondre à la détection/classification de ces actes de violence. Nous présenterons dans un premier temps des travaux uni-modaux basés sur le signal vidéo ou le signal audio, puis sur l'utilisation conjointe de ces deux modes. Enfin, nous terminerons avec la reconnaissance de violences dans l'environnement très particulier qu'est celui du transport collectif.

2.5.1 Études basées sur la vision

Comme pour la reconnaissance d'action, dans un premier temps, les travaux portant sur la reconnaissance de violences ont été menés avec des approches basées sur l'extraction de caractéristiques de haut-niveau avec lesquelles des modèles étaient estimés.

Les approches les plus anciennes cherchaient des mouvements particuliers/erratiques de personnes par des techniques de soustraction de l'arrière-plan [36]. Dans [28], le choix a été fait de calculer et d'analyser l'intensité des mouvements au travers d'une détection de la peau, l'hypothèse étant faite qu'une haute intensité étant souvent caractéristique de violence. Les caractéristiques sémantiques extraites étaient ensuite modélisées dans un moteur de règles afin de prendre une décision sur la présence ou non de violence.

Les limites de ces premiers travaux sont tout d'abord dues au cadre un peu trop "idéal" de leurs études : caméra et arrière-plan fixe, même champ de vue, absence d'occultation et une image se focalisant essentiellement sur l'action violente [36]. Dans ces travaux, le manque de performance résidait dans le fait que les caractéristiques ne permettaient pas finalement de discriminer efficacement tous les actes de violence ni même de les différencier des actes "normaux".

L'introduction de la modélisation par apprentissage et des premiers jeux de données a permis à la communauté d'aborder la problématique sous un nouvel angle. Ces nouvelles approches s'appuyaient sur des représentations de caractéristiques de haut niveau qui servaient ensuite à estimer les paramètres d'un modèle tel que des machines à vecteurs

2. Rapport mondial sur la violence et la santé - Synthèse : https://apps.who.int/iris/bitstream/handle/10665/67410/a77101_fre.pdf

supports (SVM) pour détecter et classer des violences. Ces nouvelles approches ont pu être développées grâce au partage des premiers jeux de données de taille importante tels que (*Hockey Fights* (*HF*) & *Movies Fights* (*MF*) [114] et *Crowd Violence* [67]). Ces modèles et ces jeux de données sont devenus une base de comparaison solide pour travaux qui ont suivi et notamment celui de Ionescu *et al.* [76] dans lequel les auteurs proposent de modéliser les caractéristiques de haut niveau avec une architecture neuronale profonde à base de couches entièrement connectées.

Akti *et al.*, dans [189], s'intéressent à la reconnaissance d'actions violentes en combinant une extraction de caractéristiques avec une modélisation temporelle. Pour cela, ils proposent une architecture à base de couches de convolutions 2D "*Fight-CNN*" appliquée à une nouvelle base de données qu'ils nomment *Surveillance Camera Fight* (*SCF*). À cette première architecture, les auteurs ajoutent des couches récurrentes de type LSTM avec un mécanisme d'attention comme illustré dans la figure 2.18 : les couches de convolutions 2D sont utilisées pour extraire des caractéristiques spatiales sur chaque image et les couches récurrentes sont utilisées pour modéliser leurs propriétés temporelles. Les auteurs comparent leurs architectures avec celles de la communauté et notamment *VGG16* [144] et *Xception* [25]. Les paramètres de ces architectures sont estimés sur le jeu de données *Surveillance Camera Fight* mais également sur les jeux de données de référence *Hockey Fights* (*HF*) & *Movies Fights* (*MF*). Il en ressort que la combinaison *Xception* + Bi-LSTM + Attention obtient de meilleurs résultats de reconnaissance sur les deux jeux de données de référence ainsi que sur celui proposé par Akti *et al.*. Inversement, l'architecture proposée (*FightCNN + Bi-LSTM + attention*) se comporte mieux avec leur jeu de données *SCF* plutôt qu'avec celles de références. Cette différence s'explique d'une part car les extracteurs génériques de la communauté ne sont pas adaptés à un jeu de données disposant d'une si grande variabilité. D'autre part, les jeux de données de références sont "trop petits" pour estimer convenablement les paramètres de l'architecture proposée par les auteurs. Enfin, ces travaux montrent également que l'ajout de la couche d'attention à un impact significatif sur le taux de reconnaissance des violences.

Les travaux de Cheng *et al.*, dans [22], s'intéressent à la reconnaissance d'actions violentes avec une architecture multi-branches. À cette occasion, les auteurs introduisent un nouveau jeu de données (*RWF-2000*) comportant un petit peu plus de variabilité que le jeu de données *Surveillance Camera Fight* et des scénarios plus complexes. L'architecture qu'ils proposent (Figure 2.19) est une architecture à base de couches de convolutions 3D pour extraire des caractéristiques à partir d'une séquence d'images RVB et d'une séquence de flots optiques. Les caractéristiques extraites dans chaque branche sont ensuite

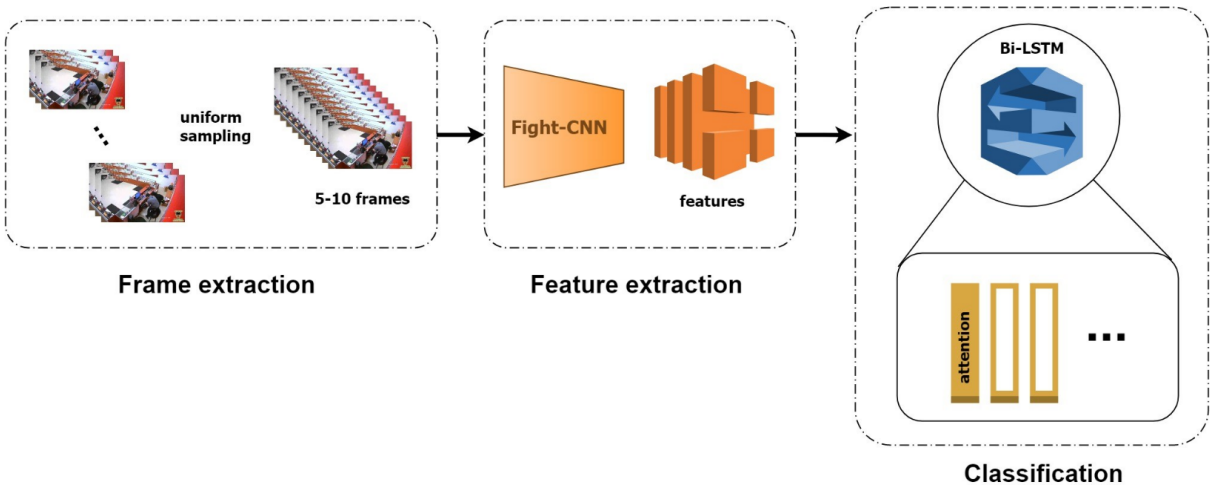


FIGURE 2.18 – Approche proposée par Akti *et al.* dans [189].

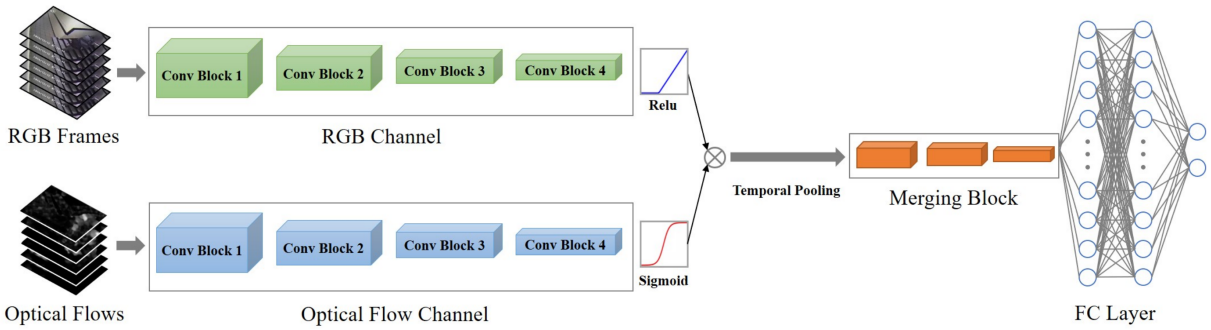


FIGURE 2.19 – Approche proposée par Cheng *et al.* dans [22].

combinées par multiplication et traitées par des couches entièrement connectées pour prendre la décision. Ils comparent leur approche avec des architectures de références telles que : *ConvLSTM* [152], *C3D* [158], *I3D* [19]. Les paramètres des différentes architectures sont estimés sur le jeu de données *RWF-2000* ainsi que sur d'autres de référence (*Hockey Fights* (*HF*) & *Movies Fights* (*MF*) et *Crowd Violence*). En premier lieu, l'architecture qu'ils proposent permet d'atteindre des résultats équivalents aux architectures de référence évaluées sur les jeux de données de la communauté. Ensuite, sur le jeu de données *RWF-2000*, l'architecture proposée permet de dépasser les performances des architectures de référence.

Le jeu de données introduit par Cheng *et al.* est repris par la suite dans les travaux de Su *et al.* [151] et Garcia-Cobo *et al.* [52], sur des travaux spécifiques à la reconnaissance de violence, ou les travaux de Hachiuma *et al.*[64], sur des travaux qui ne sont pas spécifiques à la reconnaissance de violence, avec une approche combinant l'analyse d'une séquence d'images et la détection de squelettes. Ces approches permettent aux modèles de focaliser leurs attentions sur les zones de la séquence d'images où un ou plusieurs corps sont en mouvement. Les résultats montrent que cette modélisation des actions par l'estimation de squelettes permet d'améliorer les résultats de reconnaissance.Cependant, ces performances se basent sur la capacité du détecteur de squelettes à fournir une information fiable pour le modèle de reconnaissance de violence. Lorsque ce détecteur ne détecte pas de squelette, car le corps est partiellement visible par exemple, les performances de reconnaissance de violence se retrouvent impactées.

2.5.2 Études basées sur l'écoute

En complément à l'analyse d'un flux vidéo, la reconnaissance de motifs sonores liés à la violence présente un intérêt non négligeable dans le sens où certaines violences peuvent ne pas être toujours bien définies visuellement mais présentent des caractéristiques sonores particulières et discriminantes comme les coups de feu ou les cris [56, 58, 162, 55]. Ces deux modes se complètent avantageusement, notamment en présence d'occultations vidéos ou lorsque le champ de vision est réduit.

En 2005, des modèles basés sur l'analyse du signal audio ont été proposés pour reconnaître des violences : Clavel *et al.* dans [30] et Giannakopoulos *et al.* dans [56] ont proposés des modèles estimés avec des approches d'apprentissage machine comme les GMM et SVM en se basant sur des caractéristiques de haut-niveau. Les travaux de Clavel *et al.* s'intéressaient spécifiquement à la reconnaissance de coups de feu alors que les travaux de Giannakopoulos *et al.* s'intéressaient à la modélisation de plusieurs classes de violence telles que les coups de feu, les cris, etc. Dans ces travaux, les paramètres sont extraits sur des jeux de données composés d'extraits de films contenant des violences. Les résultats de ces premiers travaux étant encourageants, les mêmes techniques ont été étendues pour discriminer différents types de violences (cris, coups de feu, explosions, bris

de glace) comme dans les travaux [162, 120, 123, 121, 49].

Plus récemment, des architectures neuronales profondes ont été appliquées au signal audio dans le cadre de la reconnaissance de violences [183, 31]. Zaheer *et al.*, dans [183], se sont intéressés à la reconnaissance de cris avec des architectures de type *Deep Boltzmann Machine* appliquées à des représentations bas-niveau de type MFCC. Cette architecture a été entraînée sur un jeu de données contenant des sons "ah" de peur et des sons "ah" de surprise, enregistrés dans différents environnements par différentes personnes. Les performances obtenues sont encourageantes et montrent que les architectures neuronales profondes permettent d'obtenir de meilleurs résultats par rapport aux approches par modèle GMM et SVM.

Colangelo *et al.*, dans [31], se sont intéressés à la reconnaissance de violences à partir de mel-spectrogrammes avec des architectures neuronales à base de couches récurrentes. Les résultats obtenus sur la base de données *MIVIA AED* montrent que l'approche avec des couches récurrentes est plus performante et plus résistante à un faible RSB que des approches plus anciennes basées sur une extraction de caractéristiques de haut niveau modélisées par un SVM.

2.5.3 Études basées sur l'utilisation conjointe de la vision et de l'écoute

Comme pour la reconnaissance d'actions, il est possible d'utiliser conjointement les signaux audio et vidéo pour améliorer la performance et la résilience des systèmes de reconnaissance de violences. Cette combinaison permet de tirer profit de la complémentarité des deux modes de perception.

Dans ce contexte, nous pouvons citer les travaux précurseurs de Nam *et al.* [111]. Ces travaux s'appuient sur la définition et l'extraction de certaines signatures sonores et visuelles telles que la variation temporelle du flot optique d'une séquence d'images, le changement brutal de la luminosité moyenne entre des images successives, l'histogramme des couleurs contenu dans les images, la variation temporelle de l'énergie du signal sonore et finalement sur une modélisation à partir de modèles multi-gaussiens (GMM). Les travaux ont été effectués sur une sélection de cinq films. Ils concluent que les résultats sont encourageants et qu'il serait intéressant d'étendre le nombre de caractéristiques sémantiques en entrée du modèle pour encore les améliorer. Cependant, ils observent que l'estimation du degré et de l'échelle de ces signatures est une tâche subjective et difficile à définir.

Dans [57], Giannakopoulos *et al.* proposent de déterminer la classe "violence" ou "non violence" en fonction d'un "méta-classifieur" appris pour identifier 4 sous-classes "non violence" et 3 sous-classes "violence". Le meta-classifieur est appris en adoptant la stratégie un contre tous et un réseau bayésien. Les distributions sous-jacentes à chaque sous-classe sont modélisées en appliquant l'algorithme des *k plus proches voisins* à un ensemble de caractéristiques extraites de chaque signal. 12 caractéristiques sont extraites sur des segments audio de durée 20ms et 100ms dont des statistiques du spectrogramme, des MFCC, le taux de passage par zéro, etc. À partir des séquences d'images sont extraits le mouvement moyen, la variance des vecteurs de mouvement mais sont aussi déterminés la présence et le suivi certains objets tels que les personnes ou des visages. Leur expérimentation est menée sur une sélection de 50 vidéos tirée de 10 films. Les performances obtenues montrent que le signal audio est plus pertinent que le signal vidéo pour reconnaître des violences, et que la combinaison des deux signaux permet d'accroître le taux global de reconnaissance de violence.

Penet *et al.*, en 2012, dans [122], ont effectués des travaux sur le jeu de données mis à disposition lors de l'atelier MediaEval 2011 Affect Task. Ce jeu de données est composé de 12 films pour l'apprentissage et de 3 films pour l'évaluation pour une durée totale de 30h de vidéo. Ce travail constitue une suite des travaux de Giannakopoulos. Les auteurs proposent un cadre de réseau bayésien pour l'intégration temporelle et la fusion d'informations multimodales. Dans un premier temps, ils montrent expérimentalement que l'introduction de la temporalité soit au travers du contexte, soit par lissage temporel permettait d'améliorer les résultats. Ils montrent également qu'une combinaison précoce avec des caractéristiques de natures différentes conduit à des résultats non satisfaisants tandis que la combinaison tardive semble plus prometteuse. Enfin, l'algorithme d'apprentissage permet d'extraire la structure entre les caractéristiques d'entrée, de détecter les variables les plus significatives et de fournir une structure temporelle cohérente.

Dans [118, 119], respectivement en 2020 et 2021, Peixoto *et al.*, se sont intéressés à la reconnaissance de violences avec des architectures neuronales profondes et une stratégie de classification d'un contre tous. Dans ce travail, les auteurs étudient comment une architecture neuronale appliquée à un signal audio et à un signal vidéo peut conduire à une bonne représentation d'une violence. Tout comme les deux travaux précédents, ils découpent la classe "violence" en k sous-classes (ici $k = 7$: sang, armes blanches, explosions, combats, feu, armes à feu et coup de feu) et pour chacune d'entre elles, il entraîne et évalue une architecture neuronale prenant en entrée des caractéristiques extraites du signal audio (MFCCs, Chroma Short-Time Fourier Transform, mel-Spectrogram et Spectral Contrast) ou du signal vidéo (images brutes, flot optique et accélération optique). Pour le signal vidéo, les architectures étudiées sont Inception v4, *C3D* et *CNN-LSTM*. Pour le flux audio, l'algorithme du Random Forest et une architecture neuronale totalement connectée sont étudiés. La méthodologie d'évaluation utilisée a permis de conclure que quelque soit l'entrée utilisée, l'*Inception V4* conduisait aux meilleurs résultats de classification dans les 7 sous-classes. Toutefois, les résultats variaient d'une sous-classe à l'autre en fonction de l'entrée utilisée. Du point de vue audio, un réseau neuronal prenant en entrée des statistiques calculées à partir de la distribution des caractéristiques précédemment citées, apporte les meilleurs résultats de classification sur les 7 sous-classes. La classification "violence" et "non violence" est obtenue en fusionnant par concaténation et par sous-classe le vecteur de sortie des 7 *Inception V4* entraînés sur la vidéo et les réseaux neuronaux entraînés sur l'audio. Finalement, après une étape de normalisation, un réseau de neurones totalement connectés est utilisé pour opérer la classification "violence" et "non violence".

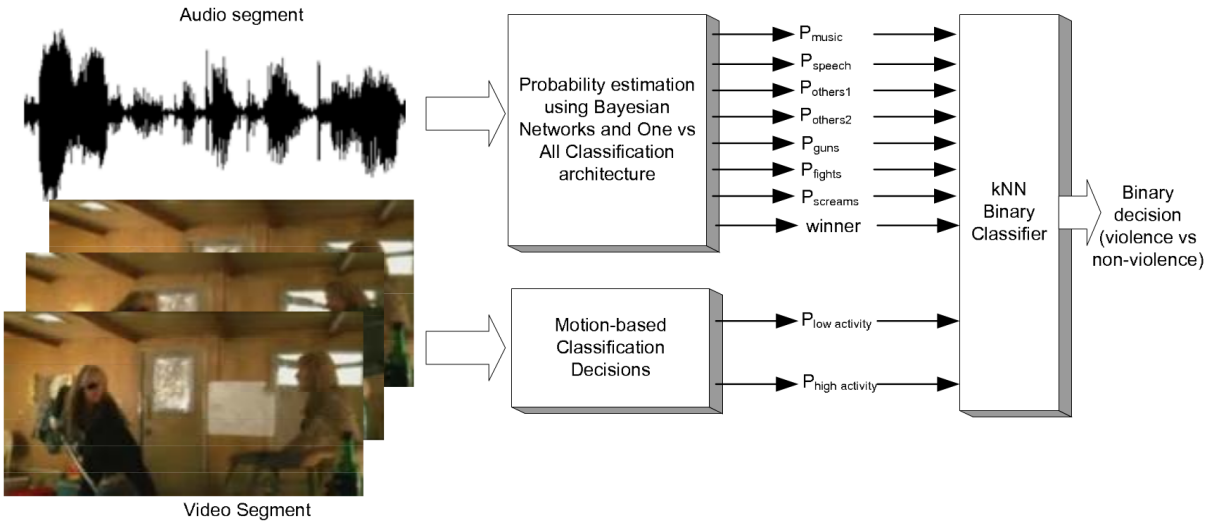


FIGURE 2.20 – Approche proposée par Giannakopoulos *et al.* dans [57].

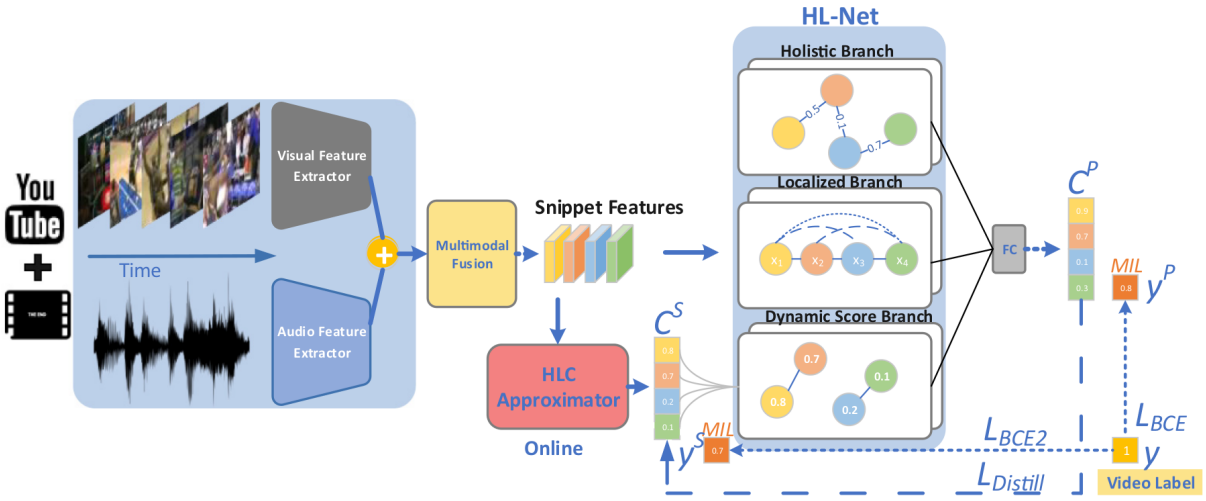


FIGURE 2.21 – Approche proposée par Wu *et al.* dans [180].

Plus récemment, Wu *et al.*, dans [180] prolongent ces recherches en entraînant des architectures neuronales sur base de vidéos faiblement annotée qu'ils ont produites : (*XD-Violence*) construit à partir de films et de vidéos partagées sur YouTube. Ils proposent deux architectures. La première (*Holistic and Local Network* : *HL-NET)* qui classifie une séquence audio/vidéo en inférant toutes ses données (c'est-à-dire du début à la fin de la séquence). La seconde (*Holistic and Localized Cue* : *HLC*) en est une approximation et permet une inférence en ligne (c'est-à-dire à un instant t à partir des données audio et vidéo seulement disponibles). La première architecture, *HL-NET* opère en deux étapes. Lors de la première étape, des extracteurs de caractéristiques pré-entraînés sont utilisés sur chaque type de données pour définir un espace de caractéristiques : *I3D* ou *C3D* pour les données vidéo et *VGGish* pour les données audio. Dans la seconde étape, la concaténation des vecteurs audio et vidéo constitue l'entrée du *HL-NET*. Le *HL-NET* est composé de 3 *Graph Neural Networks* (*GCN*) capables de capturer respectivement les relations spatio-temporelles à long-terme et les relations spatiales de proximité des caractéristiques (Holistic branch, localized branch and dynamic score branch). Il est important de noter que la branche holistique fournit une probabilité de violence à partir de la séquence entière, tandis que la dynamic score branch estime une probabilité à partir des données disponibles à un instant donné. Cette dernière branche étant causale, elle permet la mise en place du HLC et une inférence en ligne. L'évaluation de cette architecture montre que la combinaison des modes est une approche toujours plus efficace que les approches uni-modales. Elle conclut que l'exploitation simultanée des dépendances locale et à long-terme (intégration de toutes les branches) permet d'améliorer le taux de reconnaissance de la violence. Enfin, elle indique qu'une évaluation en ligne en supprimant la branche holistique réduit la précision moyenne que de 5%.

2.5.4 Reconnaissance d'actions violentes dans un environnement transport

Comme présenté en introduction de ce manuscrit, le nombre de travaux étudiant la reconnaissance d'action violente dans un environnement transport ferroviaire sont limités.

Cependant, des études ont pu déjà être menées soit dans le cadre de projet européen comme le projet BOSS [90] ou de projets nationaux comme les projets SURTRAIN et DéGIV [125, 188]. Ces travaux traitaient cette problématique avec des signaux audio et des signaux vidéo mais de manière indépendante et en dehors de la problématique neuronale. Dans ces études, les données nécessaires ont été acquises indépendamment,

sans annotation commune et sans synchronisation précise de tous les signaux.

D'un point de vue traitement du signal sonore, ces travaux mettaient en œuvre des systèmes automatiques exploitant les concepts des SVM, des GMM ou encore de la divergence de Kullback-Leibler (l'ouvrage [9] permet une vision précise de ces méthodes dans des cas généraux). Les motifs sonores principalement recherchés étaient ceux du cri et du bruit de bombes de peinture dans un contexte de dégradation.

Le système de détection proposé dans DéGIV était fondé sur une approche originale qui analysait localement les mouvements erratiques pouvant correspondre à des interactions violentes entre passagers. Comme peu d'événements violents constituaient la base d'images d'entraînement, le système exploite un modèle de normalité avec un SVM à une classe.

Pour combiner les deux décisions estimées à partir des deux flux indépendamment, une étape de fusion sur la base de la théorie des fonctions de croyance a été proposée, notamment pour tenir compte de l'asynchronisme des événements modaux.

Plus récemment, avec des approches neuronales, les travaux de Laffitte *et al.* [88, 87, 89] ont été proposés dans ce contexte d'application. Ces travaux s'intéressent à la reconnaissance de cris sur le signal audio dans le métro. Dans leur étude, ils évaluent la reconnaissance de trois classes (cris, parole et bruit de fonds) avec des architectures à base de couches de convolutions, de couches récurrentes ou de couches entièrement connectées. Les caractéristiques d'entrées des réseaux sont soit des séquences de coefficients MFCC soit des séquences de coefficients mel. L'étude montre que l'analyse temporelle avec une architecture récurrente est plus adaptée pour reconnaître des sons structurés comme le sont ceux de la parole.

Enfin, bien plus récemment, en parallèle de nos travaux, Jaafar *et al.*, dans [77], se sont intéressés à la reconnaissance du niveau de violences (3 classes : sans violence, violence moyenne et violence forte) avec les signaux audio et vidéo dans l'environnement d'un train. Les architectures qu'ils proposent combinent le signal audio, le signal vidéo, la transcription textuelle depuis l'audio et des informations "méta". Ces "méta" informations annotées manuellement indiquent quel signal a servi dans la prise de décision de l'annotateur entre le signal audio et/ou le signal vidéo mais également si l'historique de l'analyse a permis d'influencer cette annotation. Les combinaisons expérimentées sont des combinaisons des décisions d'architectures uni-modales ou de caractéristiques provenant de ces dernières (Figure 2.22). Les combinaisons ont été réalisées soit par concaténation soit par opérations mathématiques (Figure 2.23). Les résultats, sur la base de données de Yang *et al.* [182], montrent que la combinaison par concaténation est la combinaison qui donne des taux de reconnaissance équilibrés entre les classes. Enfin, les auteurs constatent que les "méta" informations fournissent une information décisive en permettant d'augmenter les taux de reconnaissance pour chacune des classes comparativement à des architectures ne disposant pas de ces informations dans la prise de décision. Malheureusement, dans un cadre opérationnel ces informations "méta" ne sont pas disponibles.

Conclusions

Dans ce chapitre, nous avons examiné les travaux antérieurs liés à notre domaine de recherche. Nous avons d'abord étudié l'évolution des jeux de données utilisés au fil du temps, constatant le manque de grands ensembles de données synchronisant le signal audio et vidéo, notamment pour la reconnaissance de la violence dans les environnements ferroviaires. Ensuite, nous avons abordé les principaux travaux de reconnaissance d'actions et de reconnaissance sonore, ainsi que les différentes approches combinant vision et écoute. Les conclusions de ces travaux démontrent l'avantage de combiner le signal audio et vidéo. Enfin, la dernière partie s'est concentrée spécifiquement sur les travaux concernant la reconnaissance des violences.

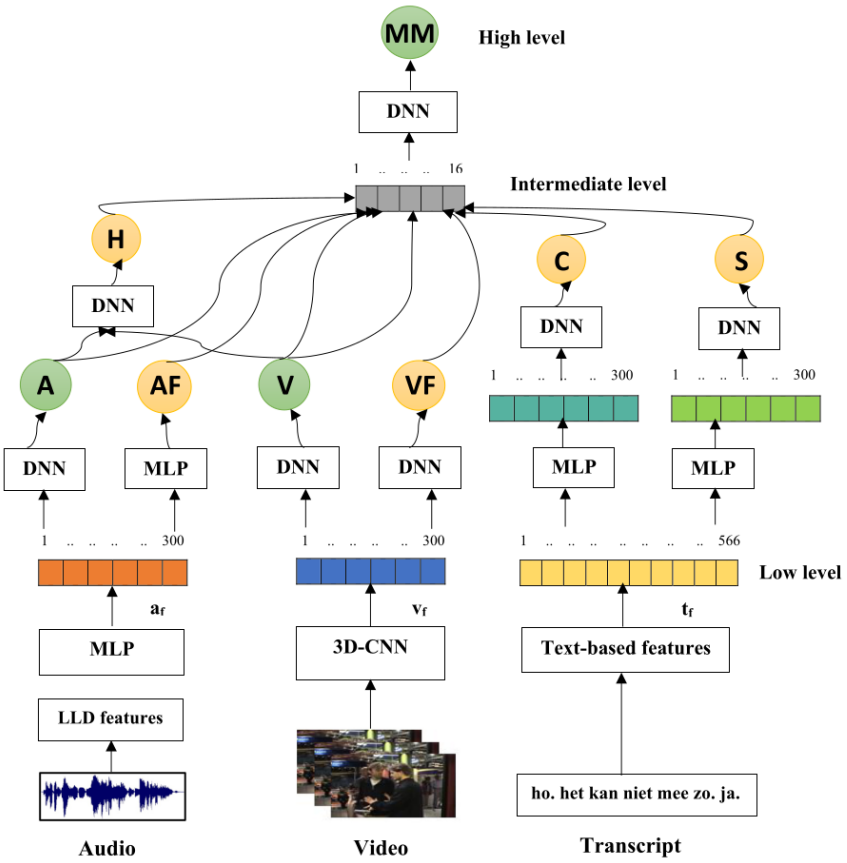


FIGURE 2.22 – Première approche proposée par Jaafar *et al.* dans [77].

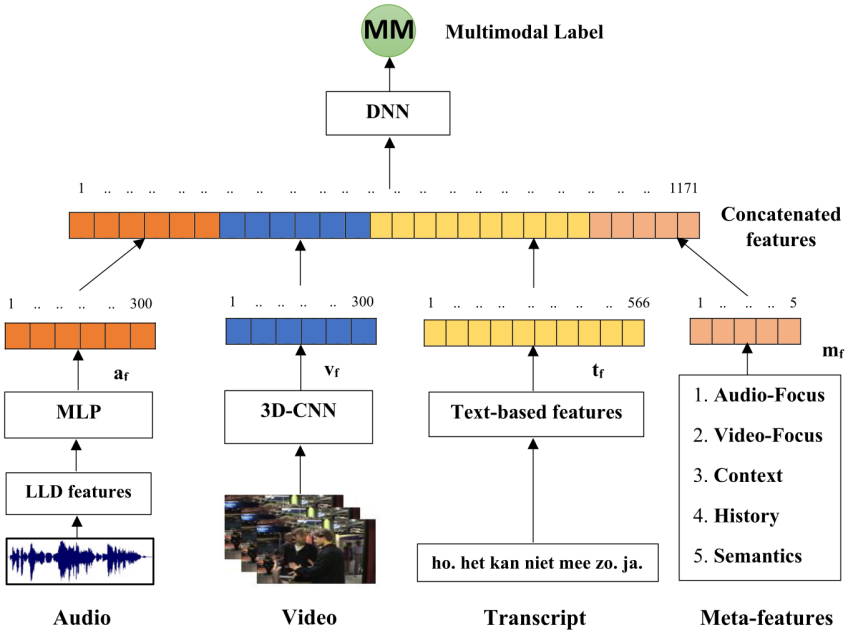


FIGURE 2.23 – Seconde approche proposée par Jaafar *et al.* dans [77].

Chapitre 3

Un jeu de données Transport et des architectures neuronales audio et vidéo

Nous avons vu dans les chapitres précédents que la reconnaissance d'action humaine et plus particulièrement celle de la violence est beaucoup plus souvent traitée par l'analyse de signaux vidéo que celle de l'analyse de signaux sonores. De plus, nous avons souligné que le traitement conjoint des signaux audio et vidéo permet d'obtenir de meilleurs résultats qu'en les traitant de manière uni-modale. Pour finir, nous avons mis en évidence que la reconnaissance de violence dans un contexte transport n'a été étudiée jusqu'à récemment que d'un point de vue uni-modal. Ce chapitre est consacré à la présentation de la base de données et des architectures neuronales proposées pour la reconnaissance d'action violente en environnement transport par le traitement conjoint des signaux audio et vidéo. Dans la première section, nous présentons une nouvelle base de signaux audio et vidéo que nous avons produite pour répondre à ce besoin en décrivant la mise en œuvre du système d'acquisition dans un train réel, la méthodologie adoptée pour les annotations des données et l'analyse de leur distribution. La seconde section est dédiée à la description des architectures neuronales étudiées et notamment des différentes stratégies de fusion qui ont été adoptées.

3.1 Jeu de données *R2N*

Comme vu dans le chapitre précédent, aucun jeu de données ne possède toutes les caractéristiques pour traiter la problématique que nous abordons à savoir une acquisition réalisée en environnement ferroviaire, de plusieurs flux audio et vidéo cohérents et synchronisés, disposant d'une résolution d'image vidéo élevée et en quantité suffisante pour faire de l'apprentissage dit "profond". Nous avons donc fait le choix d'enregistrer un jeu de données contrôlées, scénarisées, dans l'enceinte d'une rame SNCF répondant aux besoins de notre étude. Comme spécifié en section 1.2.1, le caractère scénarisé des enregistrements, nous permet de contrôler le contenu et la qualité de nos données et notamment le type des violences, leur durée, leur intensité, leur nombre, etc. Nous avons également pu contrôler l'emplacement de nos capteurs dans la rame SNCF. Ceux-ci ont été placés de telle sorte que notre jeu de données puisse être représentatif des contraintes liées à l'environnement : champ de vue, occultation, bruit de fond, défilement, etc. Évidemment, la configuration spatiale des capteurs est restée proche de celle définie par le système de surveillance embarqué d'origine. Nous étudions la problématique de reconnaissance des actes de violence sous l'angle de l'apprentissage supervisé des modèles. Dans le contexte des réseaux profonds, cela signifie que la quantité des données doit être suffisante pour

assurer la phase d'apprentissage, d'évaluation et de test, à la fois pour les classes "Violence" et "Non Violence". Au vu cette dernière remarque, nous aurions pu envisager de recueillir des données enregistrées directement en service opérationnel, mais d'un point de vue pragmatique, il aurait été difficile de récolter un nombre suffisant d'instances "Violence" à partir de telles données d'identifier et d'extraire des scènes de violences sans aucune information *a priori* sur le contenu des données. De plus, au-delà de la qualité et des difficultés de post-traitement et d'annotations des enregistrements, le nombre effectif de violences n'aurait peut-être pas été suffisant pour s'assurer de mener à bien nos travaux. Enfin, sans démarches lourdes, longues et incertaines, des données enregistrées lors de services commerciaux ne peuvent être conservées plus de 30 jours légalement.

Les sous-sections suivantes ont pour but de présenter plus précisément le contexte, le contenu et la construction de cette base de données dédiée à la reconnaissance de violences dans un environnement ferroviaire.

3.1.1 Tout d'abord, une mise en place...

Nous avons mis en œuvre toutes les étapes requises à l'acquisition d'un tel jeu de données (avec le concours de collègues SNCF quand cela fut nécessaire), à savoir : le choix de l'ensemble du système de captation, la sélection d'une rame, l'établissement des scénarios de violences et la recherche des comédiens professionnels pour jouer ces scénarios. Nous avons également géré l'organisation de la journée d'acquisition en s'acquittant des procédures de sécurité, en réservant le matériel roulant, un agent de conduite, un cadre traction, des sillons pour faire circuler la rame, etc.

Brièvement, la journée d'enregistrement s'est décomposée en deux phases : une première dédiée à l'installation de notre matériel de captation dans la rame et une seconde dédiée au "tournage" des scénarios sous notre supervision. Une fois les enregistrements récupérés, nous avons traité les vidéos en les découpant les scénarios en scènes de violence et en les annotant, comme nous le verrons plus en détails dans les sections suivantes.

3.1.2 Présentation de la rame

La rame sélectionnée pour réaliser l'acquisition des données est une *Regio2N*, aussi nommée "Porteur Hyper Dense" (PHD) construit par Bombardier, aujourd'hui Alstom. Nous avons sélectionné ce matériel car il est récent et intègre les aménagements des matériels qui seront acquis sur les prochains marchés d'acquisition de la SNCF. Présenté en figure 3.1 et figure 3.2, ce matériel initialement conçu pour le service de Transport Express Régional (TER) ou inter-ville, a été adapté au besoin péri-urbain. Il s'agit d'un matériel à l'architecture moderne de type "boa", permettant de se déplacer d'une extrémité à l'autre sans sortir de la rame. Ce matériel à l'architecture innovante est composé de plusieurs types de caisses :

— **Une caisse d'extrémité simple niveau** (nommées EXT_1N dans la figure 3.1)

— **Une caisse d'extrémité double niveau** (nommées EXT_2N dans la figure 3.1)

— **Des caisses voyageurs double niveau** (nommées V_2N dans la figure 3.1)

— **Des caisses plate-formes** (nommées PT dans la figure 3.1)

La *Regio2N* est modulable en nombre de caisses de 6, 7, 8 ou 10 caisses et modulable en aménagement intérieur des places assises avec des versions 2+2 ou 2+3 sièges par rangée comme illustré dans la figure 3.2. Ce matériel peut embarquer de 660 à 1300 personnes dont 350 à 770 personnes assises.

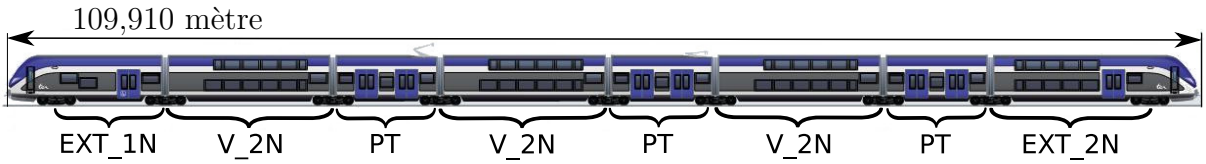


FIGURE 3.1 – Diagramme de la configuration d'un *Regio2N* mise à disposition dans le cadre de nos travaux (Doc. Bombardier)[1]

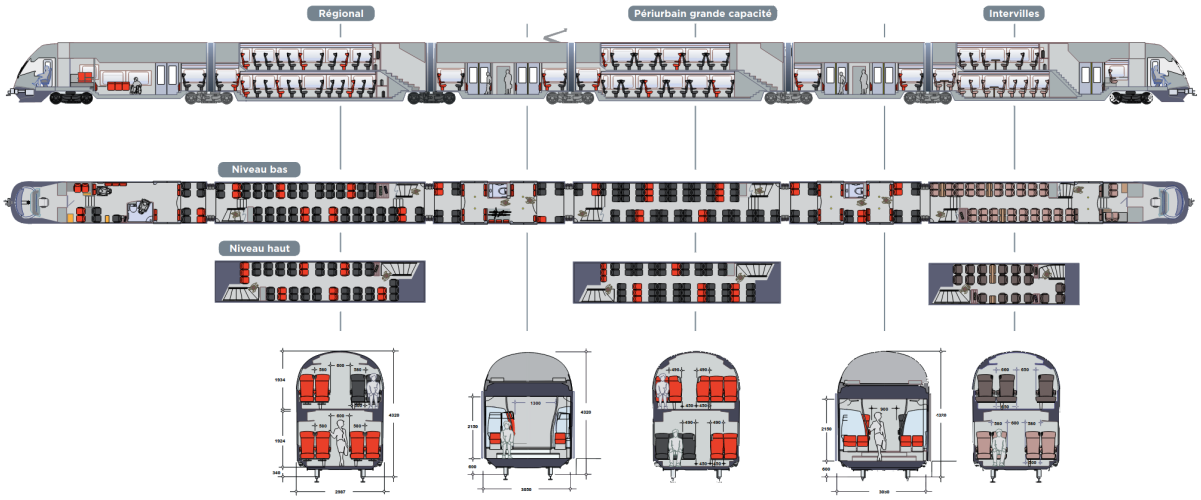


FIGURE 3.2 – Diagramme des différents aménagements pour le *Regio2N* (Doc. Bombardier)[1]

La rame que nous avons eue à disposition, exploitée par *Transilien*, est une rame 8 caisses avec un aménagement intérieur des places assises 2+3 dans sa configuration Île-de-France, avec une capacité totale de 1030 personnes dont 605 assises.

3.1.3 Instrumentation

Pour réaliser l'enregistrement vidéo et audio, la rame a été équipée avec matériel d'acquisition suivant :

— 8x Caméras AXIS P3935-LR

— 1x Station d'enregistrement AXIS S1116 MT

— 1x Switch RJ45 PoE+ AXIS T8508

Le modèle de caméra sélectionné est nativement équipé d'un microphone qui permet d'acquérir simultanément et de façon synchronisée l'audio et la vidéo en couleur (RVB). Elles ont été configurées pour acquérir la vidéo avec une résolution de 1920 × 1080 à une fréquence de 25 i/s (*image par seconde*) compressée en H264 - MPEG-4 AVC. Le signal audio a été capté avec une résolution de 32bits à une fréquence d'échantillonnage de 44,1kHz compressé en MPEG AAC avec un débit de 192kbit/s.

Présentées en figure 3.3, les caméras sont placées par deux dans les quatre salles sélectionnées pour l'expérimentation : une salle voyageurs basse, une salle voyageurs haute, une salle d'extrémité simple niveau et une plate-forme. Ces salles ont été sélectionnées afin de pouvoir acquérir les trois types d'aménagements de salles de ce type de rames.

1. http://www.horizonemployeur.fr/wordpress/wp-content/uploads/2018/06/bombardier-transportation-Regio-2N-datasheet-fr.pdf

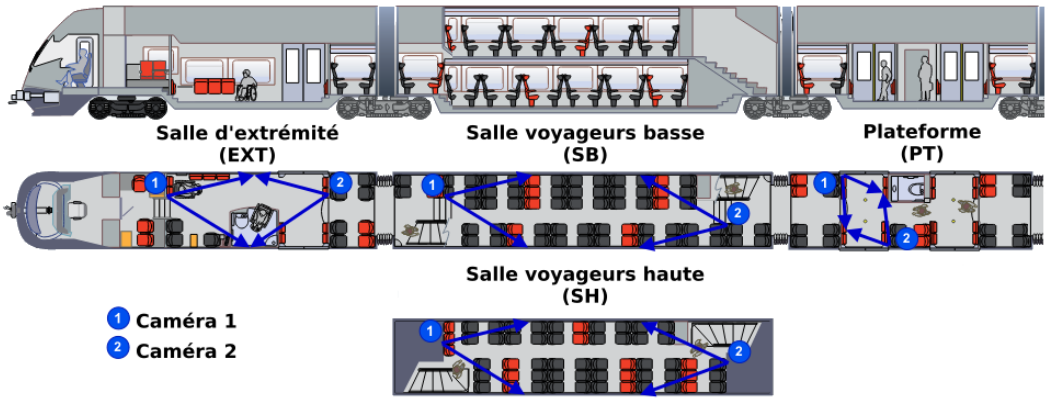


FIGURE 3.3 – Orientation des caméras dans chaque salle (L'aménagement utilisé en support dans cette illustration n'est pas exactement l'aménagement de la rame que nous eut à disposition).

3.1.3.1 Salle basse (SB)

La salle basse est la salle "basse" d'une voiture voyageurs double niveau. Longue d'environ 15 mètres, elle est large d'environ 3 mètres avec un plafond plat à 1,92 mètre. Les sièges de cette salle sont disposés par 2 d'un côté et par 3 de l'autre côté de l'allée centrale. Composée de 12 rangées, cette salle peut contenir une densité de 60 passagers assis pouvant se déplacer dans la longueur via l'allée centrale. Disposant d'un éclairage artificiel tout le long de l'allée centrale et de larges fenêtres de part et d'autre de la rame, la salle basse est donc un environnement fort lumineux pouvant être variable en fonction de la luminosité extérieure. D'un point de vue sonore, cette salle est naturellement calme par son éloignement des moteurs et des portes. Deux caméras ont été installées à chaque extrémité de la salle, dans la longueur de la voiture, comme dans le système de vidéo-protection natif de la rame, *cf.* figure 3.3. Les champs de vue des deux caméras (Figure 3.4) permettent d'observer l'ensemble de la salle, avec une perception plus « critique » au fur et à mesure que l'on s'éloigne de chaque caméra. Pour une question de simplicité, nous dénommerons dans la suite cette salle "SB".

3.1.3.2 Salle haute (SH)

La salle haute est la salle "haute" d'une voiture voyageurs double niveau. Elle reprend beaucoup de caractéristiques de la salle basse que nous venons de présenter : mêmes dimensions, même configuration d'éclairage, de fenêtres et même configuration sonore. Les caméras sont également positionnées à chaque extrémité de la salle, dans la longueur de la voiture, comme dans le système de vidéo-protection natif de la rame (*cf.* figure 3.3).

(a) Caméra 1. (b) Caméra 2.

FIGURE 3.4 – Champ de vue des caméras de la salle basse (SB).

(a) Caméra 1. (b) Caméra 2.

FIGURE 3.5 – Champ de vue des caméras de la salle haute (SH).

La salle haute se distingue de la salle basse par la forme voûtée de son plafond (à 1.93 m de hauteur au centre de l'allée) et par la présence de sorties d'escaliers à droite du champ de vue des caméras (*cf.* figure 3.5). Avec une disposition des sièges légèrement différente, la densité de passagers est équivalente à la salle basse avec 58 places assises disponibles. Dans la suite, nous dénommerons cette salle "SH".

3.1.3.3 Salle d'extrémité (EXT)

La salle d'extrémité est la salle d'une voiture d'extrémité simple niveau. Longue d'environ 12 mètres, elle est large d'environ 3 mètres avec un plafond plat à 2,15 mètres. Cette salle est composée d'une plate-forme avec des portes, d'un espace pour les personnes à mobilité réduite et d'un espace passager sans issue. L'espace pour les personnes à mobilité réduite est un espace libre avec des strapontins. Dans l'espace passager sans issue se trouve 2 carrés de 4 sièges avec une allée centrale. La densité de passagers assis est ainsi moindre que celle des salles précédentes avec 15 places assises plus la place à mobilité réduite. De plus, quelques personnes peuvent se tenir debout à l'intersection de l'espace passager sans issue et de l'espace pour les personnes à mobilité réduite ainsi qu'à l'intersection de l'espace pour les personnes à mobilité réduite et de la plateforme. Il s'agit d'une salle plutôt bruyante à cause des portes et de la proximité avec la motorisation. La configuration d'éclairage est différente des salles précédentes mais fournit une luminosité équivalente. Deux caméras ont été installées à chaque extrémité de la salle, dans la longueur de la voiture, comme dans le système de vidéo-protection natif de la rame (*cf.* figure 3.3). Comme le montrent les champs de vue des caméras en figure 3.6, cette configuration est différente des salles SB et SH de par la structure générale de la salle, de la densité de passagers possible et de l'environnement sonore. Dans la suite, nous dénommerons cette salle "EXT".

(a) Caméra 1. (b) Caméra 2.

FIGURE 3.6 – Champ de vue des caméras de la salle d'extrémité (EXT).

(a) Caméra 1. (b) Caméra 2.

FIGURE 3.7 – Champ de vue des caméras de la plate-forme (PT).

3.1.3.4 Plate-forme (PT)

La plate-forme est la salle d'une voiture plate-forme. Cette voiture est longue d'environ 8 mètres et large d'environ 3 mètres avec un plafond plat à 2,15 mètres. Cette voiture est composée de trois parties : 2 plate-formes avec des portes (d'environ 2 mètres) et un petit espace passager (d'environ 2 mètres) séparant les plate-formes avec 4 sièges en dos-à-dos. Au milieu de chaque plate-forme, 2 barres verticales de maintien sont disposées permettant d'accueillir dans cet espace des passagers non assis. La luminosité est équivalente aux salles précédentes, par contre cette salle est relativement bruyante car elle est proche des moteurs et que c'est un lieu de passage pour rejoindre les autres espaces passagers. Installées comme dans le système de vidéo-protection natif de la rame, les deux caméras sont positionnées au niveau des portes, dans la largeur de la salle (*cf.* figure 3.3). Il implique que le champ de vue des caméras est fort différent des salles précédentes, tant en termes de profondeur qu'en largeur de vue. De plus, la densité de passagers debout peut substantiellement provoquer des obstructions de vue des caméras (Figure 3.7). Dans la suite, nous dénommerons cette salle "PT".

3.1.4 Scénarisation

L'enregistrement de ce jeu de données s'est déroulé le 02 septembre 2021 sur un trajet sans arrêt entre Paris et Le Mans. Le trajet aller a été effectué de Paris à Le Mans de 11h24 à 13h35 (2h11) et le retour entre Le Mans et Paris a été effectué de 14h57 à 17h10 (2h13).

Pour des raisons expérimentales, de sécurité et afin de pouvoir recueillir facilement les droits à l'image de toutes les personnes à bord de cette rame, aucun passager classique n'était présent dans la rame. Les passagers de ce train sont donc simulés par :

— 15 comédiens professionnels et 2 coordinateurs de la société de prestation OZECLA.

— 18 agents SNCF.

Les agents SNCF simulent exclusivement les figurants alors que les comédiens professionnels simulent les scènes avec et sans violences.

Afin de guider les comédiens professionnels dans le jeu des scènes de violences, une trame de scénarios avec des contenus désirés a été construite (Tableau 3.2) ; cette trame était un guide plutôt qu'un script à suivre à la lettre car il a été recommandé aux comédiens de jouer les scènes de violences le plus naturellement possible). Plus précisément, un "scénario" (correspondant à une ligne dans le tableau 3.2) est défini par une combinaison de plusieurs caractéristiques :

— une thématique d'agression parmi vol, racisme, harcèlement, etc.

— un nombre de passagers entre 0, 5, 10 et 20.

— une salle parmi SB, SH, PT et EXT

— un état de la rame : à l'arrêt ou en dynamique

Chaque scénario est joué 3 fois à la suite avec un même couple de comédiens ("la victime" et "l'agresseur") afin de prendre en compte la distance des actions aux capteurs. Ces trois jeux correspondent à trois zones différentes dans la salle où le scénario est joué (excepté dans la salle PT où il n'y a que 2 zones) : une zone à mi-distance entre les deux caméras, une zone proche de la caméra 1 et une autre proche de la caméra 2.

Cette procédure d'enregistrement dans trois zones a été réalisée au total pour 2 à 3 couples de comédiens différents pour chaque scénario.

Des scénarios dits "complémentaires" ont aussi été joués et correspondent à des agressions de groupes et des scènes de "liesse". Ces scénarios supplémentaires ne suivent pas exactement la procédure précédente car l'effet de groupe ne représente plus précisément une zone et un couple de comédiens précis.

Après un post-traitement des enregistrements, nous obtenons au total 131 scènes dont 117 contiennent de la violence et 14 ne contiennent aucune violence (périodes de mises en place entre chaque scène). Si on considère indépendamment les caméras de chaque salle, cela revient à considérer à la fin 262 scènes : 234 scènes contenant des violences et 28 scènes n'en contenant aucune. À titre d'information, le tableau 3.1 donne la répartition du nombre de scènes par salle en fonction de la présence ou non de violence.

Les scènes de violence ont une durée variable de 32 secondes à 9 minutes et se décomposent généralement en 3 phases (*cf.* figure 3.8) : une courte phase sans violence en début et fin de scène dont la durée est comprise dans l'intervalle $[0s, 120s]$ (matérialisée en vert dans la figure), une phase de "montée de la violence" ou de "retour au calme" dont la durée est comprise dans l'intervalle $[5s, 120s]$ (matérialisée en orange dans la figure) et une phase pendant laquelle des actions de violence apparaissent dont la durée est comprise dans l'intervalle $[20s, 120s]$ (matérialisée en rouge dans la figure).

Dans la section suivante de ce chapitre, nous présentons la méthode mise en place pour annoter le contenu des 234 scènes de violence et quelques chiffres relatifs à la distribution des annotations réalisées.

	EXT	SB	SH	PT	Total
Avec violence	48	92	62	32	234
Sans violence	6	12	4	6	28
Total	54	104	66	38	262

TABLE 3.1 – Répartition des scènes avec et sans violence en fonction des salles.

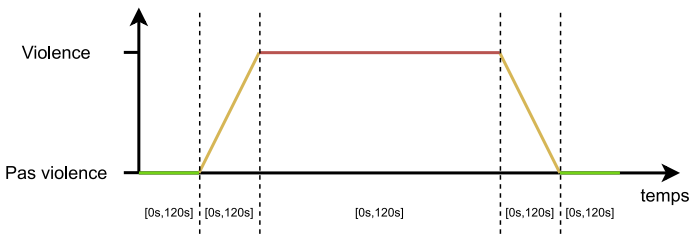


FIGURE 3.8 – Diagramme du niveau de violence en fonction du temps pour le jeu d'une scène violente.

		Thématique	Nb. passagers	Salle	Arrêt/Dynamique
Matin	Scénario 1	Vol	5	SB	Arrêt
	Scénario 2	Harcèlement	5		
	Scénario 3	Personne qui fume	20		
	Scénario 4	Demander de l'argent	20		
	Scénario 5	Demander de l'argent	10	SH	
	Scénario 6	Racisme	10		
	Scénario 7	Harcèlement	0		
	Scénario 8	Harcèlement	20	EXT	
	Scénario 9	Demander de l'argent	20		
	Scénario 10	Harcèlement	10	PT	
	Scénario 11	Harcèlement	10		
Matin	Scénario 12	Vol	0	SB	Dynamique
	Scénario 13	Personne qui fume	0		
	Scénario 14	Racisme	5		
	Scénario 15	Harcèlement	10		
	Scénario Complémentaire 1	Agression de groupe	20		
	Scénario Complémentaire 2	Agression de groupe	20		
	Scénario 16	Demander de l'argent	5	SH	
	Scénario 17	Racisme	10		
	Scénario 18	Vol	10		
	Scénario 19	Vol	5		
	Scénario 20	Harcèlement	0	EXT	
	Scénario 21	Vol	5		
	Scénario Complémentaire 3	Agression de groupe	0		
	Scénario Complémentaire 4	Agression de groupe	0		
	Scénario 22	Personne qui fume	0	PT	
	Scénario 23	Demander de l'argent	0		
	Scénario 24	Racisme	5		
Après-midi	Scénario 25	Demander de l'argent	10	SB	Dynamique
	Scénario 26	Vol	10		
	Scénario 27	Harcèlement	10		
	Scénario 28	Vol	10		
	Scénario 29	Demander de l'argent	20		
	Scénario 30	Racisme	20		
	Scénario Complémentaire 5	Liesse	33		
	Scénario 31	Personne qui fume	10	SH	
	Scénario 32	Racisme	10		
	Scénario 33	Harcèlement	10		
	Scénario 34	Personne qui fume	10	EXT	
	Scénario 35	Racisme	10		
	Scénario 36	Demander de l'argent	10		
	Scénario 37	Harcèlement	10	PT	
	Scénario 38	Racisme	10		
	Scénario 39	Demander de l'argent	10		

TABLE 3.2 – Trame de scénario des scènes de violences.

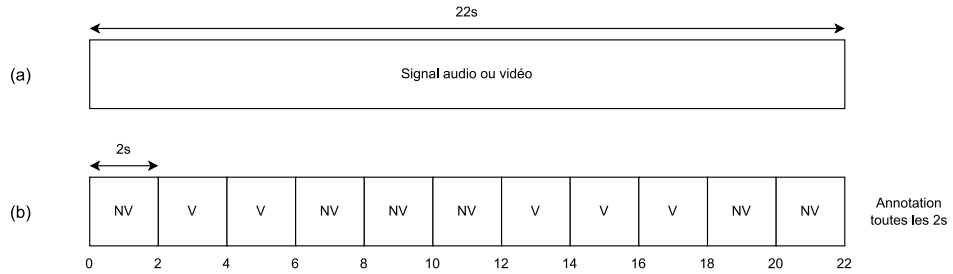


FIGURE 3.9 – Schéma relatif à l'annotation en intervalles de 2s des scènes contenant des violences (V) et des scènes ne contenant pas de violences (NV).

3.1.5 Annotation

Nous avons choisi d'effectuer une annotation consistant à indiquer la nature du contenu de chaque segment d'une durée donnée issu d'une séquence de notre base *R2N*. Nous n'avons donc pas choisi d'indexer précisément le début et la fin d'un évènement. Dans un premier temps, les flux audio et vidéo de chaque séquence ont été découpés en segments d'une durée constante. Dans un second temps, nous avons précisé la classe "violence" ou "Non Violence" à laquelle appartient chaque segment. La durée de chaque segment a été fixée à 2 secondes car nous avons observé qu'un instant de violence était rarement inférieur à cette durée (Figure 3.9).

Une annotation naturelle d'une séquence de la base aurait été d'affecter le label à chaque segment en visualisant et en écoutant simultanément son contenu. Nous avons fait le choix tout autre d'annoter en écoutant le flux audio puis d'annoter à nouveau en visualisant le flux vidéo. Nous obtenons en fin de processus deux annotations indépendantes, objectives et sans biais au regard des modes de perception utilisés, en admettant que les violences puissent être perçues différemment, voir ne pas être perçues, en fonction des capteurs utilisés. Ainsi, en considérant le signal audio, la présence de violence se manifeste par la présence de cris ou d'échanges verbaux agressifs ; et du point de vue du signal vidéo, la présence de violence est identifiée à travers la présence de bousculades, d'échanges de coups, etc.

De ces deux annotations indépendantes, nous produisons une annotation audio-visuelle que nous nommerons "globale". Cette dernière a été déduite en suivant l'équation 3.1 : un segment de 2s est annoté sans violence ($y_{globale} = 0$) si et seulement s'il n'y a pas de violence sur le signal audio et sur le signal vidéo. Si une violence est perceptible sur l'un des deux signaux, le segment de 2s est annoté comme contenant une violence.

$$y_{globale} = \begin{cases} 0, & \text{si } y_{audio} = 0 \text{ et } y_{video} = 0 \\ 1, & \text{sinon} \end{cases} \tag{3.1}$$

Dans l'objectif de mieux appréhender les résultats dans la suite de notre étude, nous avons complété l'annotation du signal vidéo par une méta-annotation subjective :

— **Le degré de violence** sur une échelle de 0 à 5. Cette annotation permet d'apprécier le degré de violence de la scène au travers notamment de la perception du mouvement : une violence avec peu de mouvements (bousculade) sera qualifiée de degré faible (0) alors qu'une violence avec beaucoup de mouvements (acharnement) sera qualifiée de degré fort (5).

— **Le degré d'occultation** sur une échelle de 0 à 5. Cette annotation permet d'apprécier la difficulté à percevoir les éléments d'une scène de violence. Les occultations se produisent à cause de l'aménagement, des passagers ou du champ de vision réduit de la caméra. Une scène de violence proche sans occultation qualifie un degré

(a) Occultation degré 0 (b) Occultation degré 1 (c) Occultation degré 2

(d) Occultation degré 3 (e) Occultation degré 4 (f) Occultation degré 5

FIGURE 3.10 – Illustrations des degrés subjectifs d'occultations de scènes de violence. Le cercle rouge indique dans l'image la présence de la scène violente.

		EXT	SB	SH	PT	Total
Scènes avec violence	Nbre de segments avec violence	854	2228	1869	886	5 837
	Nbre de segments sans violence	2011	4 440	3 047	1493	10 991
Scènes sans violence		19 647	16 034	18 734	21 583	75 998
Total de segments sans violence		21 658	20 474	21 781	23 076	86 959

TABLE 3.3 – Répartition des segments de 2s de l'ensemble des 8 caméras selon l'annotation "globale" en fonction des classes Avec ou Sans violence et des salles.

d'occultation faible (0) alors qu'une scène lointaine avec de l'occultation qualifie un degré d'occultation fort (5). La figure 3.10 donne une illustration de l'échelle du degré de l'occultation.

Le degré de violence à partir du signal audio n'a pas été considéré car cette annotation serait plus délicate à mettre en œuvre. En effet, sur la base uniquement de l'énergie sonore perçue du signal, il est difficile d'associer un degré de violence sans avoir connaissance de la distance au microphone à laquelle se déroule la scène. Ce travail serait possible si nous avions accès à l'image dans le même temps. Or, comme indiqué précédemment, nous nous sommes placés dans le cadre de deux annotations indépendantes.

3.1.6 Analyse de la base de données

Nous proposons dans cette section d'analyser les résultats de la procédure d'annotation décrite ci-avant en considérant différentes répartitions des segments de 2s avec ou sans violence, résultant de l'annotation "globale".

Le tableau 3.3 présente le nombre de segments de 2s contenant et ne contenant pas de violence en fonction des salles. On constate un déséquilibre entre les segments avec violence et sans violence ; ceci est principalement dû au fait qu'une scène de violence est composée de périodes non-violentes avant et après les violences effectives, et parfois certaines scènes peuvent comporter des périodes de violence non-perceptibles. Par ailleurs, en prenant en compte l'ensemble des 8 caméras (2 par salles), pour une salle avec une violence jouée, il y a 3 salles sans scène de violence ; le déséquilibre entre les segments avec violence et sans violence des intervalles de 2s est ainsi majoré.

	EXT	SB	SH	PT	Total
Audio & Vidéo	304	588	508	225	1625
Audio	462	1578	1309	654	4003
Vidéo	88	62	52	7	209

TABLE 3.4 – Répartition par salles des segments de 2s contenant des violences en fonction du signal sur lequel a été perçue la violence.

	EXT	SB	SH	PT	Total
Caméra 1	413	1098	896	432	2839
Caméra 2	441	1130	973	454	2998

TABLE 3.5 – Répartition par salles des segments de 2s contenant des violences perçues en fonction de la caméra.

Parmi les segments contenant des violences, l'évènement n'est pas toujours perceptible sur les deux modes de perception. Le tableau 3.4 présente la répartition des segments de 2s contenant des violences en fonction du signal sur lequel a été perçue la violence. Tout d'abord, on peut observer que peu de violences sont perceptibles seulement sur le signal vidéo car des cris vont souvent accompagner les scènes de violences. Ensuite, on peut remarquer qu'il y a plus de violence perceptible sur le signal audio que simultanément sur le signal audio et le signal vidéo, ceci étant principalement dû au fait que le signal audio n'est pas sensible aux occultations totales ou aux scènes hors-champ.

La répartition des segments de 2s contenant des violences (sur l'annotation globale) en fonction de la salle peut aussi être détaillée en fonction de la caméra 1 et 2 (Tableau 3.5), en fonction de la zone dans laquelle est jouée la violence (Tableau 3.6) ou encore en fonction du degré d'occultation ou du degré de violence (Tableau 3.7 et 3.8).

En analysant le nombre de violences perçues par les caméras 1 et 2 en champs de vue croisés (Tableau 3.5), on peut remarquer qu'une même scène de violence peut ne pas être perçue identiquement sur les caméras : en effet, des occultations peuvent être présentes que sur un seul point de vue. Ceci implique que le nombre de violences dans une même salle peut être différent d'une caméra à l'autre.

En observant le nombre de violences jouées par Zones, dans le tableau 3.6, on peut analyser qu'il y a plus de violences captées dans les Zone 1 et 2 que dans la Zone 3. Cette répartition peut s'expliquer par le fait que les violences en Zone 3 peuvent être moins perceptibles sur le mode audio et le mode vidéo, celles-ci étant plus éloignées du capteur, elles peuvent être sujettes à plus d'occultation.

Enfin, en s'appuyant sur la méta-annotation du signal vidéo, on peut remarquer dans le tableau 3.7, une répartition des degrés d'occultations centrée sur une occultation de degré 3. Cette répartition s'explique par le fait que l'environnement ferroviaire est un environnement présentant beaucoup d'occultations dû à l'arrangement de la rame ou aux passagers.

	EXT	SB	SH	PT	Total
Zone 1	277	776	441	430	1924
Zone 2	340	766	510	456	2072
Zone 3	237	686	326	-	1249

TABLE 3.6 – Répartition par salles des segments de 2s contenant des violences perçues en fonction de la zone

	EXT	SB	SH	PT	Total
Occultation 0	42	31	34	42	149
Occultation 1	66	61	50	18	195
Occultation 2	34	146	112	36	328
Occultation 3	118	252	207	79	656
Occultation 4	83	151	152	56	442
Occultation 5	49	9	4	1	68

TABLE 3.7 – Répartition des segments de 2s contenant des violences en fonction du degré d'occultation et des salles.

	EXT	SB	SH	PT	Total
Violence 1	6	49	65	14	135
Violence 2	50	166	128	41	385
Violence 3	104	179	132	61	476
Violence 4	103	158	94	63	418
Violence 5	129	98	140	53	420

TABLE 3.8 – Répartition des segments de 2s contenant des violences en fonction du degré de violence et des salles.

Dans le tableau 3.8, on peut observer une répartition relativement uniforme entre les violences de degrés 2 à 5 et peu de segments de 2s contenant une violence de degré 1. Ce nombre plus faible de segments s'explique par la difficulté à discriminer entre une scène sans violence et une scène contenant une violence de degré 1.

Finalement, la mise en œuvre de l'acquisition des données nous a permis de construire un jeu de données d'une taille proche des jeux de données avec des scènes de violences de la communauté en termes de durée totale des scènes de violences avec 3 heures et 14 minutes. Il a été construit afin de présenter une certaine variabilité, tant dans les divers espaces d'une rame ferroviaires, que dans la représentation de la violence à travers des jeux et des acteurs différents. Ainsi, ce jeu de données répondant aux besoins de notre étude nous permet donc d'estimer et d'évaluer objectivement les paramètres des architectures de la tâche de reconnaissance de violence dans un environnement transport embarqué.

3.2 Nos architectures

Les différentes architectures neuronales profondes que nous proposons ont pour objectif de reconnaître la présence d'une violence dans une séquence d'observations à travers un signal (audio ou vidéo) ou à travers plusieurs signaux issus de capteurs différents (audio et vidéo). Dans ce denier cas, nous définirons des architectures multi-modales pour lesquelles nous aborderons la combinaison des modes audio et vidéo sous différents angles : le premier consistera à établir des modèles en fonction du niveau auquel aura lieu la combinaison dans l'architecture (combinaison bas niveau, haut niveaux...). Le deuxième angle proposera différentes stratégies de mise en œuvre, c'est-à-dire la solution technique permettant de réaliser cette combinaison (par moyenne, par concaténation, etc.).

Les architectures proposées suivent une approche "générique" composée de 3 à 4 blocs : un premier bloc d'extraction de caractéristiques, un deuxième bloc de modélisation des séquences de caractéristiques, un troisième bloc dédié à la combinaison pour les architectures multi-modales et un dernier bloc de décision. Dans un premier temps, nous présenterons l'architecture uni-modale audio, les architectures uni-modales vidéo, puis suivront les architectures multi-modales, avec les différents niveaux de combinaisons (combinaison des

décisions, combinaison tardive et combinaison moyenne) suivis des stratégies de combinaisons (combinaison par concaténation, combinaison par mécanisme à porte et combinaison par attention croisée).

3.2.1 Architecture audio

L'architecture de référence uni-modale audio (Figure 3.11) est constituée de l'extracteur de caractéristiques *OpenL3* [33, 5] que nous avons décrit dans la section 2.4.1 Apprentissage conjoint de l'audio et la vidéo. On rappel que les paramètres de l'extracteur *OpenL3* ont été appris pour deux jeux de données différents, *AudioSet-Music* et *AudioSet-Environmental*. Nous avons retenu celui dont les paramètres ont été estimés sur *AudioSet-Music* car l'étude [33] montre que le sous-ensemble *Music* est plus performant que le sous-ensemble *Environmental*. L'extraction de caractéristiques (ϕ_a) est réalisée par une architecture à base de convolutions 2D appliqué sur un mel-spectrogramme (a). En sortie d'*OpenL3*, nous avons traité cette séquence de caractéristiques par l'ajout d'une couche récurrente de type LSTM. À la suite, sont ajoutées trois couches entièrement connectées (FC) activées par la fonction d'activation *ReLU*. Le nombre d'unités de ces couches sera défini afin de réduire progressivement le nombre d'unités et tendre vers le nombre d'unités de la couche finale de l'architecture. Enfin, pour prendre la décision (p_{audio}), la couche finale ($\mathbf{z}_a$) possède deux unités activées par une $softmax$ (Équation 3.2). Cette architecture sera nommée *Audio* dans la suite de cette thèse.

$$softmax(\mathbf{z}_a) = \frac{e^{\mathbf{z}_a}}{\sum_{j=1}^{2} e^{\mathbf{z}_a(j)}} \tag{3.2}$$

$$p_{audio} = softmax(\mathbf{z}_a) \tag{3.3}$$

3.2.2 Architecture vidéo

Pour l'architecture uni-modale vidéo, nous avons considéré la même architecture générale que celle du modèle uni-modale audio mais où l'extracteur de caractéristiques vidéo est l'architecture *I3D* [19] que nous avons présentés dans la section 2.2.3 (Figure 3.12).

Les paramètres de cet extracteur sont estimés sur le jeu de données *Kinetics-400* [81]. La séquence de caractéristiques extraite (ϕ_v) est traitée avec l'ajout d'une couche récurrente de type LSTM. Comme pour l'architecture audio, suivent trois couches entièrement connectées (FC) dont le nombre d'unités est réduit pour tendre vers le nombre d'unités de la couche finale. Leur fonction d'activation est la fonction *ReLU*. La décision "Violence" *vs.* "Non Violence" est prise avec (p_{video}) en sortie de la $softmax$ appliquée à $\mathbf{z}_v$ (Équation 3.4). Cette architecture sera nommée *Vidéo* dans la suite de cette thèse.

$$p_{video} = softmax(\mathbf{z}_v) \tag{3.4}$$

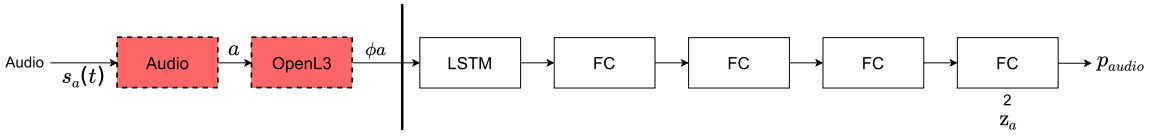


FIGURE 3.11 – Architecture uni-modale audio (*Audio*), avec $s_a(t)$ le signal sonore brute, a un mel-spectrogramme, ϕ_a une séquence de caractéristiques sonores. Le bloc *Audio* est un bloc de pré-traitement et d'estimation du mel-spectrogramme, le bloc *OpenL3* est un bloc d'extraction de caractéristiques.

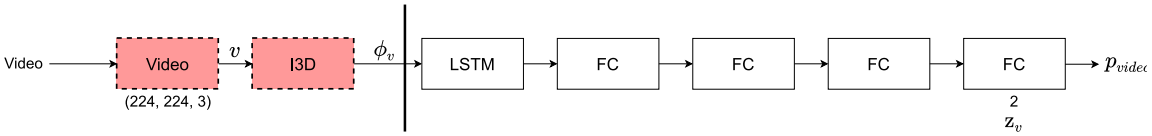


FIGURE 3.12 – Première architecture uni-modale vidéo (*Vidéo*), avec *Vidéo* un flux d'images brutes $1280 \times 720 \times 3$, v une séquence d'image $224 \times 224 \times 3$, ϕ_v une séquence de caractéristiques vidéo. Le bloc *Vidéo* est un bloc de pré-traitement, de segmentation et de mise à l'échelle des images, le bloc *I3D* est un bloc d'extraction de caractéristiques.

L'extracteur *I3D* considère en entrée une séquence d'images de résolution (224×224). Il est donc nécessaire de réduire la résolution de nos vidéos enregistrées avec une résolution (1280×720). Ceci est réalisé après un remplissage appliqué au-dessus et en dessous des images afin de réaliser cette réduction sur des images carrées (1280×1280). La figure 3.13 présente un aperçu du résultat et du rapport d'échelle entre les données brutes (Figure 3.13 (a)) et les données pré-traitées (Figure 3.13 (b)). On peut observer facilement que les scènes ont une définition différente en fonction de leur distance au capteur et que cette définition est particulièrement faible lorsque les scènes se produisent loin du capteur. Malheureusement, cette variation particulière de définition n'a pas été considérée dans le jeu de données *Kinetics-400* lors de l'estimation des paramètres de l'architecture *I3D*. Par conséquent, le modèle risque de ne pas être en mesure d'extraire correctement les caractéristiques des scènes jouées loin du capteur.

Pour éviter ce biais et rester cohérent avec les données utilisées pour l'estimation des paramètres de l'*I3D*, nous choisissons de transformer nos données et proposons une nouvelle architecture vidéo pour s'adapter à cette variation de définition. Cette seconde architecture uni-modale vidéo (Figure 3.15) considère en entrée trois découpages de l'image originale en parallèle. Ces découpages sont illustrés dans la figure 3.14 pour la caméra 1 de la salle voyageur basse. Le premier découpage, *Crop1*, (Figure 3.14 (a)) permettra de traiter les scènes proches du capteur, le second découpage, *Crop2*, (Figure 3.14 (c)) a pour objectif de mieux traiter les scènes a mi-distance du capteur et enfin le dernier découpage, *Crop3*, (Figure 3.14 (e)) pour objectif de mieux traiter les scènes les plus éloignées du capteur. Les mêmes pré-traitements que la première architecture uni-modale vidéo sont

FIGURE 3.13 – Résultat de la réduction de la résolution des images vidéo de 1280×720 à 224×224. Exemple de la caméra 1 de la salle basse. (a) : Image originale de taille 1280×720 en sortie du capteur. (b) : Image de 224×224 après pré-traitements à l'entrée de l'extracteur de caractéristiques *I3D*.

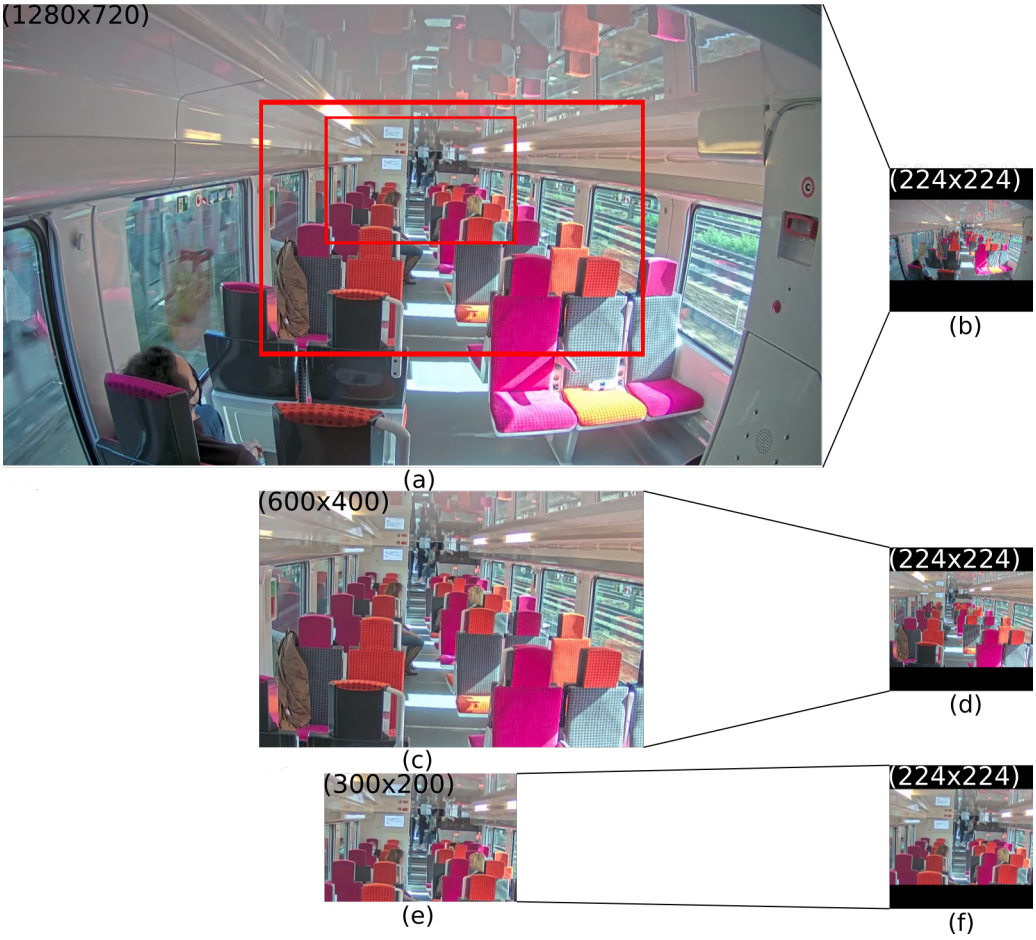


FIGURE 3.14 – Visualisation pour la caméra 1 de la salle basse d'une image originale (a) et des images après les différents découpages (c, e). (b, d, f) sont les images associées après pré-traitement présentées à l'entrée des extracteurs de caractéristiques *I3D* de notre architecture *VidéoCrop*.

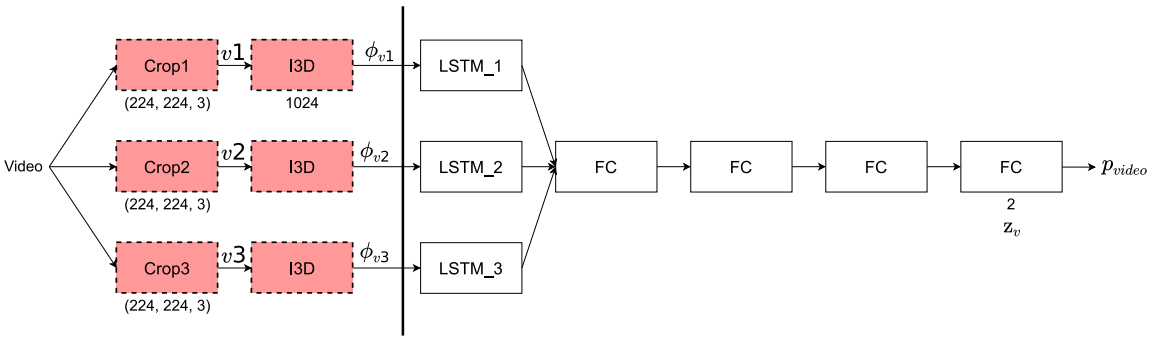


FIGURE 3.15 – Seconde architecture uni-modale vidéo (*VidéoCrop*).

appliqués sur chacun des découpages. Les caractéristiques ϕ_{v1}, ϕ_{v2} et ϕ_{v3} sont extraites en parallèle sur les séquences d'images pré-traitées $v1$, $v2$ et $v3$ avec l'architecture *I3D*. Ces séquences de caractéristiques sont chacune traitées avec l'ajout de trois couches récurrentes de type LSTM. Elles sont ensuite combinées par concaténation pour être traitées par trois couches consécutives entièrement connectées (FC) activées par la fonction d'activation *ReLU*. Enfin, comme précédemment, la décision "Violence" vs. "Non Violence" est prise à partir (p_{video}), la sortie d'une $softmax$ appliquée à ($\mathbf{z}_v$) (Équation 3.4). Cette architecture sera nommée *VidéoCrop* dans la suite de cette thèse.

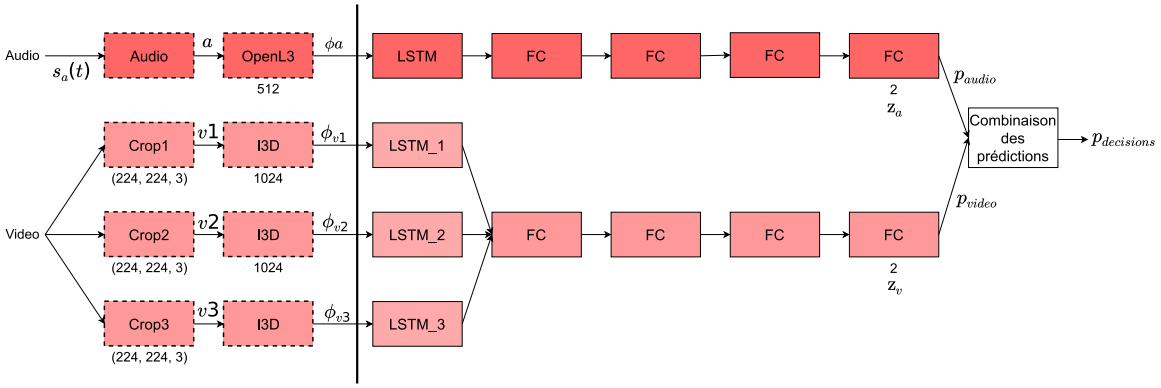


FIGURE 3.16 – Architecture combinant les décisions (*Décisions*).

3.2.3 Architectures combinant l'audio et la vidéo

Comme précisé en introduction de ce chapitre, nous avons étudié différents types architectures afin de traiter conjointement le signal audio et le signal vidéo. Nous présenterons dans un premier temps des architectures considérant différents niveaux de combinaisons :

— le niveau moyen, c'est-à-dire juste après la partie de modélisation des séquences de caractéristiques,

— le niveau tardif, c'est-à-dire sur l'antépénultième couche entièrement connectée du bloc de décisions,

— le niveau décision, c'est-à-dire la combinaison des décisions des architectures unimodales.

Dans un second temps, nous présenterons des architectures réalisant cette combinaison selon 3 stratégies de combinaisons différentes :

— la combinaison par concaténation,

— la combinaison par mécanismes à portes,

— la combinaison par attention croisée.

3.2.3.1 Les niveaux de combinaisons

3.2.3.1.1 Combinaison des décisions

Ce premier niveau de combinaison (Figure 3.16) est le plus élémentaire et définit une architecture simple et "naïve" qui consiste à combiner les décisions des architectures uni-modales présentées dans les sections 3.2.1 et 3.2.2 suivant l'équation 3.5. Avec cette combinaison, les paramètres des architectures uni-modales *Audio* et *VidéoCrop* seront estimés indépendamment lors de l'apprentissage et seules leurs décisions seront combinées. Cette architecture sera nommée *Décisions* dans la suite de cette thèse.

$$p_{decisions} = \begin{cases} 0, & \text{if } p_{audio} = 0 \text{ and } p_{video} = 0 \\ 1, & \text{sinon} \end{cases} \tag{3.5}$$

3.2.3.1.2 Combinaison à un niveau tardif

Le deuxième niveau de combinaison mis en place est une combinaison tardive (Figure 3.17). L'objectif de cette combinaison est de prendre une décision conjointe en interprétant mieux les différentes cohérences et complémentarités des deux branches audio et

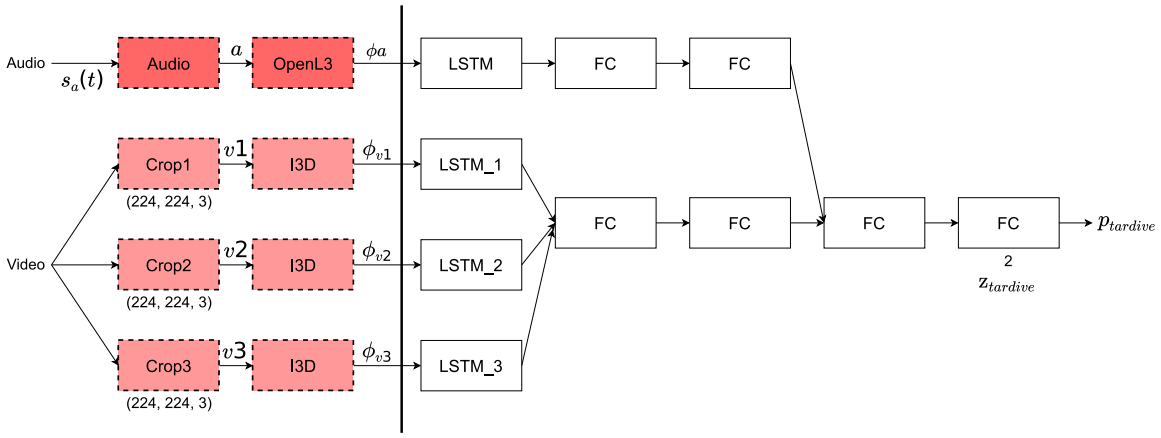


FIGURE 3.17 – Architecture de combinaison à un niveau tardif (*Tardive*).

vidéo. Cette stratégie consiste à combiner les projections de l'antépénultième couche entièrement connectée des architectures uni-modales *Audio* et *VidéoCrop* avec 2 couches entièrement connectée (FC) consécutives activées par la fonction d'activation *ReLU*. La décision ($p_{tardive}$), est obtenue par le résultat d'une fonction $softmax$ appliquée après les 2 couches FC consécutives ($\mathbf{z_{tardive}}$) (Équation 3.6). Pour ce niveau de combinaison, la phase d'apprentissage consistera à estimer conjointement l'ensemble de l'architecture à partir des deux signaux audio et vidéo (excepté les extracteurs audio et vidéo). Cette architecture sera nommée *Tardive* dans la suite de cette thèse.

$$p_{tardive} = softmax(\mathbf{z_{tardive}}) \tag{3.6}$$

3.2.3.1.3 Combinaison à un niveau moyen

Le troisième et dernier niveau de combinaison réalisé est une combinaison à un niveau moyen (Figure 3.18). L'objectif de cette combinaison est de construire tôt une projection commune des branches, pouvant modéliser les cohérences et complémentarités entre audio et vidéo. Cette stratégie consiste à combiner les projections de caractéristiques à la sortie des couches récurrentes de type LSTM des branches audio et vidéo. La combinaison de ces projections est ensuite traitée par trois couches entièrement connectées (FC) activées par la fonction d'activation *ReLU*. Enfin, pour prendre la décision ($p_{moyenne}$), la couche finale ($\mathbf{z_{moyenne}}$) possède deux unités activées par une $softmax$ (Équation 3.7). Pour ce niveau de combinaison, comme précédemment, la phase d'apprentissage consistera à estimer l'ensemble des paramètres de l'architecture, excepté les extracteurs de caractéristiques. Cette architecture sera nommée *Moyenne* dans la suite de cette thèse.

$$p_{moyenne} = softmax(\mathbf{z_{moyenne}}) \tag{3.7}$$

3.2.3.2 Les stratégies de combinaisons

Pour les équations associées aux 3 stratégies de combinaisons présentées ci-après, les projections de la branche audio seront désignées par **a**, les projections des branches vidéo associées au découpage 1, 2 et 3 seront respectivement désignées par **v1**, **v2** et **v3**.

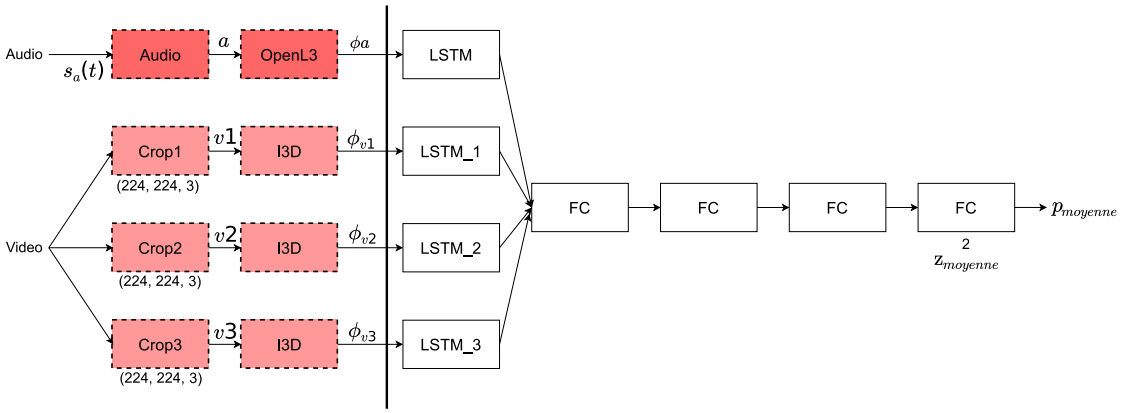


FIGURE 3.18 – Architecture de combinaison à un niveau moyen.

3.2.3.2.1 Combinaison par concaténation

La première stratégie de combinaison réalisée est la concaténation. Cette stratégie laisse aux couches entièrement connectées qui suivent le soin de sélectionner les caractéristiques les plus pertinentes avant la modélisation des cohérences des deux modes.

$$\mathbf{z} = concat([\mathbf{a}, \mathbf{v1}, \mathbf{v2}, \mathbf{v3}]) \tag{3.8}$$

3.2.3.2.2 Combinaison par mécanismes à portes

La deuxième stratégie de combinaison mise en œuvre est la combinaison par mécanisme à portes. Cette stratégie de combinaison est intéressante car elle permet de réaliser une sélection des caractéristiques pertinentes dans les projections de chacune des branches. Cette stratégie est décrite dans l'équation 3.9 et est illustrée en figure 3.19. Pour chacune des branches (la projection audio et les trois projections vidéos), une projection cachée est calculée (h_a, h_{v1}, h_{v2} et h_{v3}) avec une fonction *tanh*. En parallèle, une projection cachée commune est calculée sur la concaténation des entrées, activée par une fonction *sigmoid*. La combinaison est alors obtenue en faisant la somme des projections cachées h_a, h_{v1}, h_{v2} et h_{v3}. Cette projection est multipliée par la projection cachée commune.

$$\begin{aligned}
\mathbf{h_a} &= tanh(\mathbf{W_a} \cdot \mathbf{a}) \\
\mathbf{h_{v1}} &= tanh(\mathbf{W_{v1}} \cdot \mathbf{v1}) \\
\mathbf{h_{v2}} &= tanh(\mathbf{W_{v2}} \cdot \mathbf{v2}) \\
\mathbf{h_{v3}} &= tanh(\mathbf{W_{v3}} \cdot \mathbf{v3}) \\
\mathbf{h} &= \sigma(\mathbf{W_h} \cdot [\mathbf{a}, \mathbf{v1}, \mathbf{v2}, \mathbf{v3}]) \\
\mathbf{z} &= \mathbf{h} \times (\mathbf{h_a} + \mathbf{h_{v1}} + \mathbf{h_{v2}} + \mathbf{h_{v3}})
\end{aligned} \tag{3.9}$$

3.2.3.2.3 Combinaison par attention croisées

Enfin, la troisième et dernière stratégie de combinaison proposée est la combinaison par attention croisée. L'intérêt de cette stratégie de combinaison est d'identifier dans les différentes projections les caractéristiques les plus discriminantes, de les mettre en avant et de réduire l'importance de celles servant peu à la décision. Cette stratégie décrite dans l'équation 3.10 est illustrée dans la figure 3.20. Pour chacune des branches (la projection audio et les trois projections vidéos), une attention est calculée entre la projection audio et les projections vidéos ($\mathbf{h_{av1}}$, $\mathbf{h_{av2}}$, $\mathbf{h_{av3}}$, $\mathbf{h_{v1a}}$, $\mathbf{h_{v2a}}$ et $\mathbf{h_{v3a}}$). Ces nouvelles projections

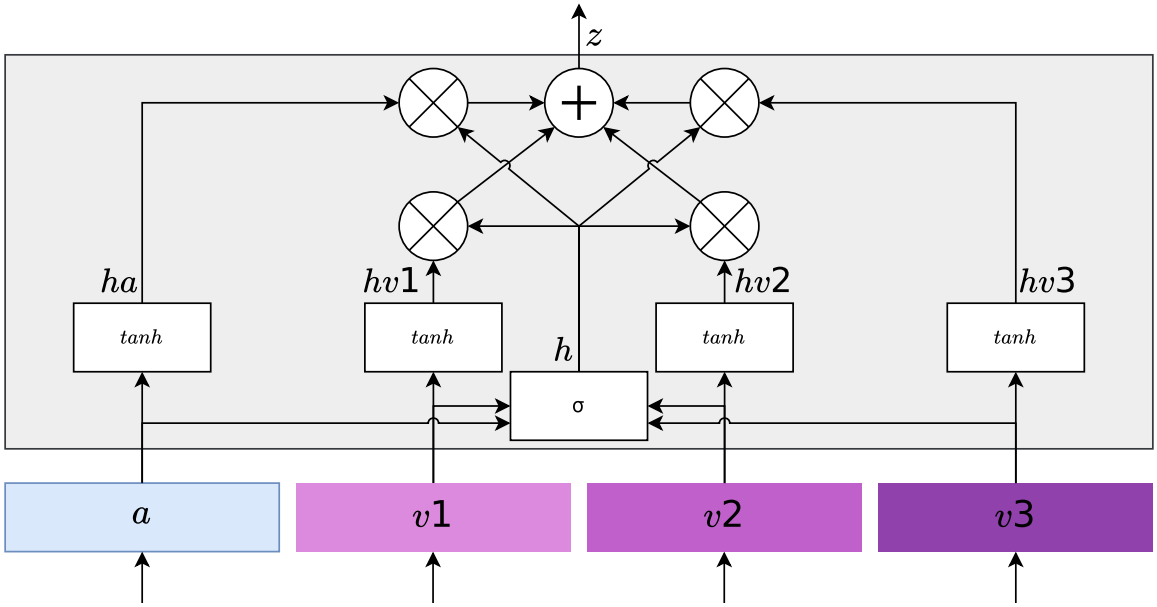


FIGURE 3.19 – Couche de combinaison par mécanisme à porte.

sont additionnées sur chaque entrée pour mettre en avant ou réduire les caractéristiques. Pour terminer, toutes les projections normalisées sont concaténées pour former **z**.

$$
\begin{aligned}
Attention(Q,K,V) &= softmax(\frac{QK^\top}{\sqrt{d_k}})V \\
\mathbf{h_{av1}} &= Norm(Attention(\mathbf{a},\mathbf{v1},\mathbf{v1}) + \mathbf{a}) \\
\mathbf{h_{av2}} &= Norm(Attention(\mathbf{a},\mathbf{v2},\mathbf{v2}) + \mathbf{a}) \\
\mathbf{h_{av3}} &= Norm(Attention(\mathbf{a},\mathbf{v3},\mathbf{v3}) + \mathbf{a}) \\
\mathbf{h_{v1a}} &= Norm(Attention(\mathbf{v1},\mathbf{a},\mathbf{a}) + \mathbf{v1}) \\
\mathbf{h_{v2a}} &= Norm(Attention(\mathbf{v2},\mathbf{a},\mathbf{a}) + \mathbf{v2}) \\
\mathbf{h_{v3a}} &= Norm(Attention(\mathbf{v3},\mathbf{a},\mathbf{a}) + \mathbf{v3}) \\
\mathbf{z} &= [\mathbf{h_{av1}}, \mathbf{h_{av2}}, \mathbf{h_{av3}}, \mathbf{h_{v1a}}, \mathbf{h_{v2a}}, \mathbf{h_{v3a}}]
\end{aligned}
\tag{3.10}
$$

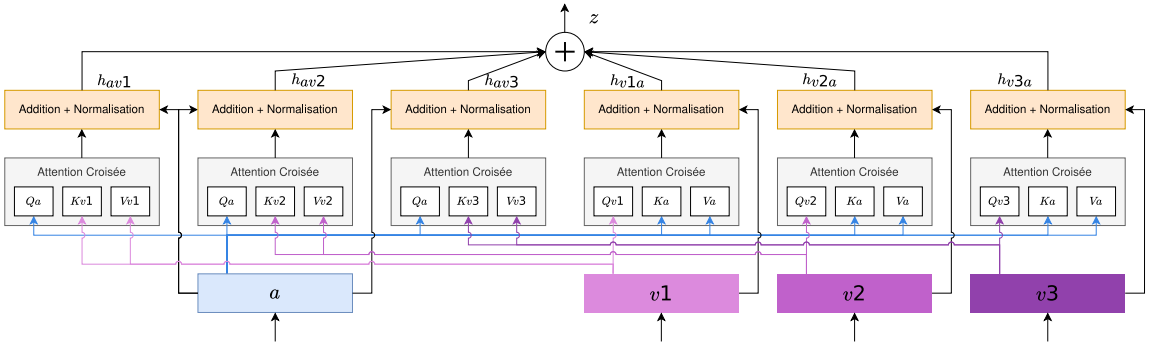


FIGURE 3.20 – Couche de combinaison par attention croisée.

Conclusions

Ce chapitre est une présentation de la construction du jeu de données que nous avons utilisé dans nos travaux et des différentes architectures uni- et multi-modales que nous avons évalués dans nos travaux. Dans un premier temps, nous avons l'acquisition de notre base de données, la technique retenue pour l'annotation des données ainsi que quelques statistiques basées sur les annotations produites. Puis, nous avons présenté les spécificités de nos différentes architectures telles que les extracteurs de caractéristiques retenus, le traitement de la profondeur de champ sur le mode vidéo ainsi que les niveaux de combinaison et les types de combinaison évalués.

Chapitre 4

Méthodologie

4.1 La Base de données *R2N*

4.1.1 Établissement de la base de données

Les architectures que nous avons présentées au chapitre précédent reçoivent en entrée, comme tout système de reconnaissance opérationnel, des flux continus à traiter et à analyser. Cependant, le traitement et les décisions associées nécessitent une certaine durée d'observation qui se traduit par une observation composée de N échantillons consécutifs. Dans notre étude, nous avons décidé d'inférer nos différents réseaux sur des segments de 5s, c'est-à-dire une durée différente de celle utilisée lors de l'annotation. Dans ce cas d'usage, les architectures proposées seront donc entraînées à reconnaître une violence sur des segments de 5s, qu'elle soit partiellement ou complètement présente sur la durée du segment.

Vu que nous cherchons à reconnaître la présence de violence sur la séquence d'entrée quel que soit le signal la percevant (audio ou vidéo ou les deux), nous étendons l'annotation de référence dite *Globale* (Section 3.1.5) à chaque segment de 5s. La figure 4.1 présente la procédure utilisée pour construire notre base de données avec une annotation sur un segment de 5s. La séquence est découpée en segments de 5s avec un chevauchement de 3s (Figure 4.1 (c)) et l'annotation des segments de 5s est déduite en suivant l'équation 4.1 : un segment de 5s est annoté "Violence" ($y = 1$) si au moins un segment de 2s le composant est lui même annoté comme violence.

$$y = \begin{cases} 0, & \text{si } \sum_{i=0}^{3} y^i_{globale} = 0 \\ 1, & \text{sinon} \end{cases} \tag{4.1}$$

où $i = 3$ pour 3 annotations de 2s consécutives afin de pouvoir considérer une durée de 5s.

Sur l'ensemble des enregistrements effectués en parallèle sur les 8 caméras, les enregistrements permettant d'extraire les 232 scènes de violences sont :

— les enregistrements de la salle basse, la salle haute et de la salle d'extrémité (les enregistrements de la plateforme n'ont pas été retenus car ils présentent des champs de vue trop différents des autres salles) ;

— les enregistrement contenant des scénarios où des scènes de violence ayant été jouées (les caméras enregistrant les salles sans violence joués à un instant t ne sont pas retenues) ;

— les enregistrement ne contenant pas de scénarios complémentaires.

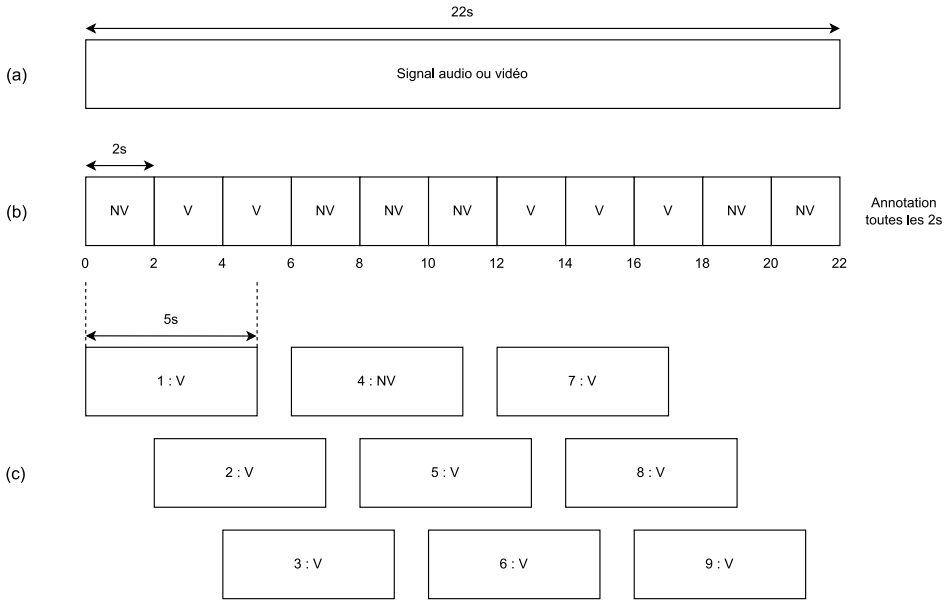


FIGURE 4.1 – Stratégie d'annotations des segments de 5s. (a) : Séquence d'une scène de violence. (b) : annotations "Violence" (V) et "Non Violence" (NV) sur des sections de 2s. (c) : Indexation des données finales sur une durée de 5s avec un chevauchement de 3s. Tout segment de 5s incluant au moins une annotation de violence de 2s est annoté comme violence sur son ensemble.

4.1.2 Répartition et équilibrage de la base de données

L'estimation et l'évaluation des paramètres des diverses architectures que nous proposons se feront classiquement à travers un découpage de notre jeu de données en trois ensembles : *Entraînement*, *Validation* et *Test*. Ces trois ensembles sont utilisés à différentes étapes de nos expérimentations :

- L'ensemble *Entraînement* : Ensemble consacré à l'estimation des paramètres de nos architectures, c'est-à-dire que c'est sur cet ensemble que l'optimisation est réalisée pendant la phase d'entraînement ;
- l'ensemble de *Validation* : à la fin de chaque *epoch* au cours de la phase d'optimisation, à savoir après une estimation des paramètres de nos architectures sur l'ensemble d'*Entraînement*, l'ensemble de *Validation* est utilisé pour suivre le bon déroulement de l'apprentissage et notamment s'assurer de la bonne convergence du critère à minimiser et de l'absence de sur-apprentissage. Ainsi nous stopperons ou non l'apprentissage en fonction de la valeur du critère évalué sur ces données, afin d'en retenir le meilleur jeu de paramètres ;
- l'ensemble de *Test* : les paramètres retenus seront ensuite utilisés pour tester les architectures sur l'ensemble de *Test*, constitué par définition de données n'ayant pas été utilisées lors de la phase d'apprentissage.

Notre jeu de données est réparti en suivant les proportions suivantes :

- 75% pour l'ensemble d'*Entraînement*,
- 12,5% pour l'ensemble de *Validation*,
- 12,5% pour l'ensemble de *Test*.

Il est important de noter que ces trois ensembles ont été générés sur la base des scénarios. Ce choix a été fait afin de s'assurer que des scènes provenant d'un même scénario ne peuvent être présentées à la fois dans l'ensemble d'*Entraînement*, de *Validation* et dans l'ensemble de *Test*. Rappelons que pour un même scénario, différentes scènes peuvent être

		Entraînement	Validation	Test
Non Violence		8633 (± 609)	1439 (± 484)	1477 (± 443)
Violence	sur Audio & Vidéo	1732 (± 136)	284 (± 99,1)	291 (± 112)
	sur Audio seul	2883 (± 2634)	487 (± 183)	527 (± 210)
	sur Vidéo seul	144 (± 19,0)	23,7 (± 13,9)	21,7 (± 13,9)
	sur l'ensemble des modes	4426 (± 324)	686,0 (± 231)	702 (± 234))

TABLE 4.1 – Moyennes et écart-types du nombre de segments de 5s annotés "Violence" et "Non Violence" après 100 répartitions et tirages aléatoires pour chacun des ensembles d'*Entraînement*, de *Validation* et de *Test* (avant équilibrage).

"similaires" si on considère la répétition du type de violence ou encore la salle dans laquelle l'action se déroule. Ce contrôle permet de nous mettre dans les meilleures conditions possibles afin d'éviter la sur-estimation des paramètres et de mieux "généraliser" la prise de décision de nos architectures.

Nous avons effectué 100 répartitions aléatoires des scénarios sur chacun des 3 ensembles (*Entraînement*, *Validation*, *Test*). Le tableau 4.1 présente les moyennes et les écart-types du nombre de segments de 5s annotés "Violence" et "Non Violence". Pour l'annotation "Violence" la moyenne et l'écart-type sont calculés par mode de perception. Nous pouvons constater dans un premier temps qu'au sein des ensembles générés, le nombre de segments de 5s est déséquilibré entre les deux classes "Violence et "Non Violence". Dans un second temps, en considérant la répartition de la violence en fonction de ses modes de perception, nous constatons qu'une grande partie des violences ne sont perçues que sur le mode audio et qu'une seconde partie l'est simultanément sur les deux modes. Il apparaît que la perception par la vidéo seule ne représente qu'une très petite partie des violences annotées.

Afin de faciliter l'optimisation et l'estimation des paramètres de nos architectures et d'augmenter la robustesse des modèles appris, nous avons eu comme premier objectif d'équilibrer les 3 ensembles pour les deux classes "Violence" et "Non Violence". Dans un deuxième objectif nous avons également voulu équilibrer la répartition de la classe "Violence" en fonction des modes de perception, afin d'éviter que le modèle soit plus représentatif d'un mode que d'un autre, augmentant ainsi sa capacité à apprendre les possibles cohérences et complémentarités entre audio et vidéo. Pour cela nous avons appliqué la stratégie d'équilibrage aléatoire dur (Random Hard Sampling Technique) [177]. Cette stratégie d'équilibrage consiste à sélectionner aléatoirement parmi les segments de 5s de la classe "Non Violence" (majoritaire) le nombre de segments de la classe "Violence" (minoritaire). Elle a été appliquée sur les annotations modales conditionnant l'annotation *Globales* que nous avons produites ("Non Violence", "Violence" sur audio & vidéo, "Violence" sur audio et "Violence" sur vidéo). Plus précisément, nous avons équilibré les classes "Non violence avec "Violence" sur audio & vidéo et les classes "Violence" sur audio avec "Violence" sur vidéo. Au final nous avons réduit le déséquilibre entre les classes "Non Violence et "Violence" (tous modes confondus) tout en maximisant le nombre de segments où la violence est perceptible seulement sur un des deux modes ("Violence" sur audio et "Violence" sur vidéo).

Au final, toujours sur la base des 100 tirages, nous présentons, dans le tableau 4.2, les moyennes et les écarts-types du nombre d'annotations "Violence" et "Non Violence" obtenues sur les segments de 5s après la stratégie d'équilibrage. Tout comme pour le tableau précédent, pour l'annotation "Violence" la moyenne et l'écart-type sont calculés par mode de perception. Conformément à la stratégie d'équilibrage que nous avons mise en place, le nombre total de segments de 5s annotés "Violence" sur le mode "Audio & Vidéo" (mode majoritaire) est égal au nombre de segments de 5s annotés "Non Violence". Ceci nous à permis d'équilibrer plus facilement et conjointement les violences perçues

		Entraînement	Validation	Test
Non Violence		1732 (± 136)	284 (± 99,1)	291 (± 112)
Violence	sur Audio & Vidéo	1732 (± 136)	284 (± 99,1)	291 (± 112)
	sur Audio	144 (± 19,0)	23,7 (± 13,9)	21,7 (± 13,9)
	sur Vidéo	144 (± 19,0)	23,7 (± 13,9)	21,7 (± 13,9)
	sur l'ensemble des modes	2019 (± 153)	331 (± 110)	335 (± 126)

TABLE 4.2 – Moyennes et écart-types du nombre des annotations "Violence" et "Non Violence" après répartition des segments de 5s réalisées sur 100 tirages aléatoires pour chacun des ensembles d'entraînement, de validation et de test (après équilibrage), en fonction de la classe "Non Violence" et de la classe "Violence" selon les modalités de perception.

uniquement sur le mode audio ou sur le mode vidéo. Le nombre de violences perçues que sur le mode audio a donc été ajusté et diminué :

— vis-à-vis du mode audio & vidéo, pour permettre au modèle de se spécialiser davantage sur le mode audio & vidéo et donc mieux apprendre les relations entre les caractéristiques audio et vidéo ;

— vis-à-vis du mode vidéo, pour éviter que le mode audio ne soit trop fortement représenter au détriment du mode vidéo.

4.2 Évaluation

Nous avons fait le choix d'évaluer nos travaux selon deux axes. Le premier est quantitatif afin de mesurer et de comparer les résultats entre les architectures. Le second, qualitatif, est ciblé sur l'architecture ayant obtenu les meilleurs résultats lors de l'évaluation quantitative. L'évaluation qualitative a pour objectif de mieux identifier les facteurs impactant le plus la reconnaissance de violence.

4.2.1 Évaluation quantitative

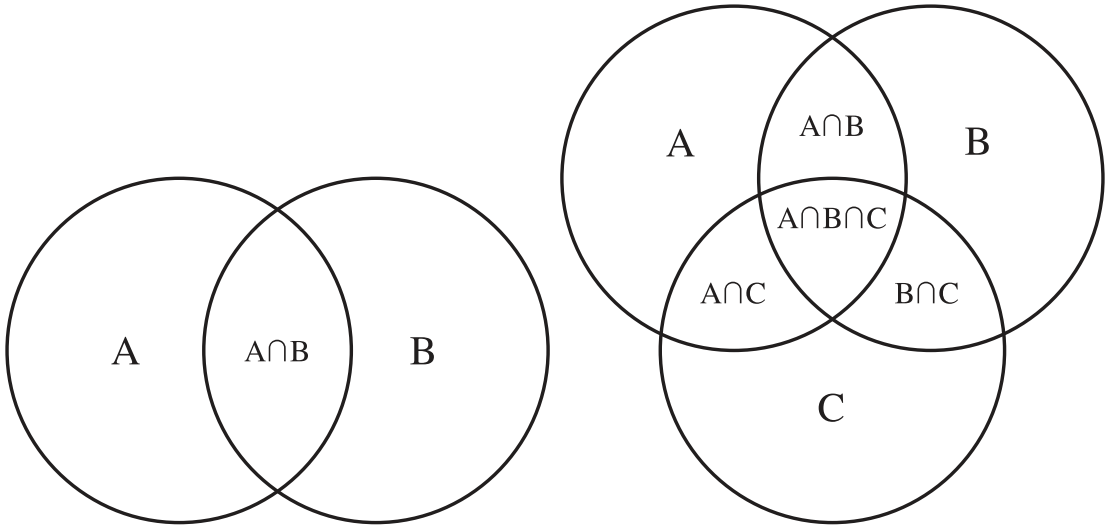
Les métriques sélectionnées afin de comparer quantitativement les performances des modèles sont la matrice de confusion, l'exactitude, la précision, le rappel et le diagramme de Venn.

• La **matrice de confusion**. Comme déjà présentée en section 1.5, la matrice de confusion est un moyen classique d'évaluation des systèmes de classification. Dans notre contexte, les éléments positifs sont les segments annotés "Violence", et les éléments négatifs sont les segments "Non Violence".

• L'**exactitude**, la **précision**, le **rappel** ont déjà été présentés en section 1.5 et nous permettrons de mieux apprécier les performances en fonction des taux de bonnes détections, de non détections, de fausses alarmes. Ils seront exprimés en pourcentage.

• Les **Diagrammes de Venn**, introduits en 1880 par John Venn [166, 167] sont une amélioration des diagrammes d'Euler. Ces graphiques, conçus dans la théorie des ensembles, permettent d'apprécier les relations logiques entre des ensembles. Encore d'actualité [71, 137, 178], nous n'avons pas croisé cette représentation dans la littérature dédiée à la reconnaissance automatique, que ce soit uni ou multi modale. Nous proposons dans ce qui suit de faire brièvement une description de ces diagrammes afin de mieux appréhender leur interprétation quand nous les exploiterons dans l'analyse de nos résultats.

Comme indiqué ci-avant, un diagramme de Venn est un schéma utilisé pour représenter et décrire les relations existantes entre des ensembles. Ces ensembles sont représentés par



(a) Diagramme de Venn d'ordre 2. (b) Diagramme de Venn d'ordre 3.

FIGURE 4.2 – Exemples de diagrammes de Venn [178]. (a) : Diagramme d'ordre 2, produisant un total de $R = 4$ régions, A, B, $A \cap B$ et de l'ensemble vide représenté par aucune des régions occupées. Les régions A, B sont composées de membres ne faisant partie que d'un seul ensemble et d'aucun autre. La région $A \cap B$, est sont composées de membres faisant partie des deux ensembles. (b) : Diagramme d'ordre 3, produisant un total de $R = 8$ régions. Les régions A, B et C sont composées de membres qui ne font partie que d'un seul ensemble et d'aucun autre. Les trois régions $A \cap B$, $A \cap C$ et $B \cap C$ sont composées de membres faisant partie de deux ensembles mais pas du troisième. La région $A \cap B \cap C$ est composée de membres faisant partie simultanément des trois ensembles.

des courbes simples fermées dans le plan (communément des cercles) qui se croisent mutuellement et dont il résulte un nombre de régions distinctes. Ces régions représentent alors les éléments communs entre ces ensembles. Un diagramme de Venn est qualifié d'ordre N quand celui-ci est une collection de N ensembles tel que (1) les courbes divisent le plan en $R = 2^N$ régions connectées et (2) chaque sous-ensemble de $\{1, 2, ..., N\}$ correspond à une région unique formée par l'intérieur des intersections des courbes. La figure 4.2 issue de [178], présente des diagrammes de Venn pour deux et trois ensembles.

Dans nos travaux les ensembles représenteront les différents modes (audio, vidéo, audio et vidéo) utilisés pour réaliser la reconnaissance de violence. Ils seront caractérisés par leurs erreurs de reconnaissance de violence (FP ou FN). L'objectif de cette évaluation est de mesurer le nombre d'erreurs communes entre les différentes architectures que nous évaluerons afin de mieux comprendre l'importance et l'influence des modes les uns par rapport aux autres. La figure 4.3 présentes des diagrammes de Venn, générés par la librairie *PyVenn*[1], tels que nous les représenterons et où des couleurs permettront de mieux distinguer les différentes interactions entres les différents modes.

4.3 Caractéristiques et paramètres

4.3.1 Extraction des caractéristique audio

Comme présenté dans la section 3.2.1, l'architecture *Audio* se base sur l'extracteur de caractéristiques *OpenL3* dont les paramètres sont estimés sur la base de données *AudioSet-Music*. Parmi les diverses configurations proposées [5, 33], nous avons choisi le modèle

1. https://github.com/tctianchi/pyvenn

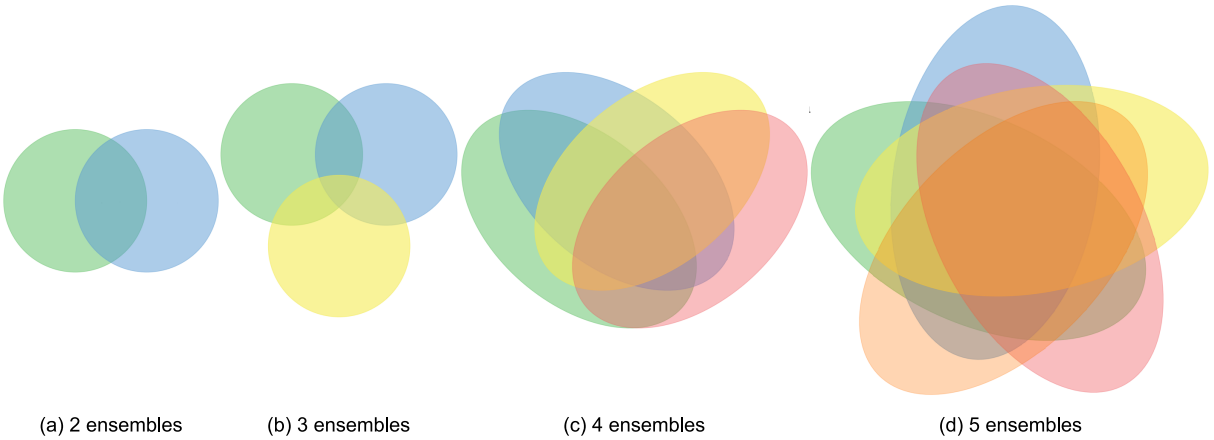


FIGURE 4.3 – Illustration des possibles représentations sous forme de diagramme de Venn.

prenant pour entrée un mel-spectrogramme de 1s composé de vecteurs de 128 cœfficients fréquentiels. Le traitement des segments de 5s de signal audio consiste alors à considérer un segment de 5,5 secondes dont les 0.5s premières secondes sont composées de remplissage de 0. Ce segment est ensuite envoyé dans le modèle *OpenL3* en le décomposant en 46 segments de 1s extraits toutes les 100ms. Avec cette configuration, l'extracteur projette 5 secondes de signal audio ($s_a(t)$) dans un espace composé de 46 vecteurs consécutifs de caractéristiques de dimensions 512 (ϕ_a).

4.3.2 Extraction des caractéristiques vidéo

La première architecture vidéo présentée en section 3.2.2 (Architecture *Vidéo*) considère l'extracteur de caractéristiques *I3D* avec des paramètres estimés sur la base de données *Kinetics-400*. Après un remplissage de 280 zéros sur le haut et le bas de l'image originale (1280×720) une réduction de la résolution de 224×224 est réalisée sur chaque image avant d'être envoyée à l'extracteur *I3D*. Pour traiter 5 secondes de signal vidéo, cet extracteur considère en entrée une séquence de 125 images RVB successives (acquises à 25 i/s) réduites qu'il projette dans un espace composé de 15 vecteurs successifs de caractéristiques de dimension 1024 (ϕ_v).

La seconde architecture vidéo (Architecture *VidéoCrop*) que nous proposons à la spécificité de considérer, comme l'architecture précédente, une suite d'images originales (1280×720) ainsi que celles extraites de deux zones des images originales : la première zone est de résolution 600×400 et la seconde zone est de résolution 300×200, chacune définie en fonction du contenu des zones à observer comme précisé en section 3.2.2. Ces 3 séquences de 5s sont composées chacune de 125 images et sont ensuite traitées en parallèle par un extracteur *I3D* dédié. Nous obtenons donc en sortie 3 "canaux" de 15 vecteurs successifs de caractéristiques de dimension 1024.

4.4 Paramètres d'optimisation

Tout au long de nos diverses expérimentations, les paramètres de nos architectures sont estimés en minimisant la fonction de perte *CrossEntropy* (Équation 4.2) selon une descente de gradient composée de lots de 128 segments de 5s et un nombre d'*epochs* égal à 500. Pour nos architectures, nous avons choisi la méthode d'optimisation *Adam* avec un taux d'apprentissage de 0.0001.

$$loss(\mathbf{p}, \mathbf{y}) = \sum_{i=1}^{C=2} y_i \log(p_i) \tag{4.2}$$

où **p** est le vecteur d'estimation des deux neurones de sortie de l'architecture et **y** est l'annotation de la portion de 5s associée pour une observation donnée.

Par ailleurs, une couche de normalisation est ajoutée avant les couches récurrentes de type LSTM, afin d'obtenir une distribution homogène et centrée des caractéristiques.

Enfin, des couches de *Dropout* sont ajoutées entre chaque couche avec un taux de 0.5 afin d'éviter tout risque de sur-estimation.

4.5 Se comparer à la communauté

Dans cette section, nous décrivons les évaluations que nous avons menées afin de comparer notre architecture avec certaines propositions de l'état de l'art. Dans un premier temps, il s'agit d'évaluer notre architecture neuronale vidéo de base, c'est-à-dire sans crop sur les bases *Hockey Fight*, *Surveillance Camera Fight*, et *RWF-2000*. Dans un second temps, il s'agit d'étudier les performances de trois architectures de la littérature sur notre base *R2N*.

4.5.1 Architecture *Vidéo* et les bases de données de la communauté

Cette première phase d'évaluation a pour objectif de valider la pertinence de l'architecture uni-modale *Vidéo* (n'utilisant que le mode vidéo) présentée en section 3.2.2 sur des jeux de données "vidéo" de la communauté, dédiés à la reconnaissance de violence. Rappelons que les jeux de données utilisés pour éprouver notre architecture sont :

— *Hockey Fight* [114] : ce jeu de données présenté en section A.4 contient des violences provenant de match de hockey ;

— *Surveillance Camera Fight* [189] : ce jeu de données également présenté en section A.4 est un jeu de données de vidéo provenant de YouTube contenant des violences agrégés ;

— *RWF-2000* [22] : ce jeu de données tout comme le précédent contient des vidéos contenant des violences agrégés de la plateforme de partage de vidéo YouTube. Ces vidéos peuvent être captées depuis une caméra fixe (une caméra de surveillance) ou une caméra mobile (une caméra de téléphone). Ce jeu de données est aussi plus précisément présenté dans la section A.4.

Les jeux de données retenus fournissent par défaut des ensembles d'*Entraînement*, de *Validation* et de *Test* afin de pouvoir comparer les résultats avec les autres travaux utilisant ces jeux de données.

Le nombre d'unités pour l'architecture *Vidéo* est fixé à 512 pour la première couche récurrente de type LSTM. Après ce LSTM, les trois couches entièrement connectées sont composées de respectivement 256, 64 et 16 neurones. Enfin, la sortie de cette dernière couche est envoyée à la couche de décision.

La procédure expérimentale consiste donc à estimer les paramètres et à évaluer l'architecture *Vidéo* sur chacun de ces jeux de données selon l'optimisation décrite en section 4.4.

La métrique d'évaluation retenue pour cette première expérimentation est le score d'exactitude. Les résultats obtenus seront comparés avec ceux atteints dans l'état de l'art (*State Of The Art* (SOTA)).

4.5.2 Évaluations de trois architectures de la communauté sur *R2N*

Cette seconde phase d'évaluation consiste à évaluer trois architectures de la communauté sur notre base de données *R2N* (section 3.1) :

— La première architecture éprouvée est une architecture ne considérant que des signaux vidéo. C'est l'une des architectures proposées par Cheng *et al.* développées pour la base de données *RWF-2000* [22]. Cette architecture prend en entrée une séquence d'images RVB (nommée ***RVB***). Les paramètres de cette architecture sont estimés en minimisant la fonction de perte *CrossEntropy*. Nous décidons de conserver les paramètres de l'optimisation associés à cette architecture qui est réalisée avec une descente de gradient stochastique (aka. "SGD" pour *Stochastic Gradient Descent*), calculée sur des lots de 32 segments de 5s. Le nombre d'*epochs* est fixé à 30 avec un taux d'apprentissage de 0.01.

— la deuxième architecture est aussi une autre version d'architecture proposée par Cheng *et al.* dans [22]. La différence est que cette version prend en entrée une séquence d'images RVB et une séquence de flots optiques (nommée ***RVB+FO***). La procédure et les paramètres d'entraînement sont identiques à la version précédemment décrite : l'optimisation est réalisée avec une descente de gradient stochastique, calculée sur des lots de 32 séquences, un nombre d'*epochs* de 30 et un taux d'apprentissage de 0.01.

— La troisième architecture retenue est celle proposée par Tian *et al.* développée pour la base de données *AVE* dédiée à la combinaison du signal audio et du signal vidéo (séquence d'images RVB) [157] (mais hors contexte de la notion d'actions violentes). Comme pour les architectures que nous proposons, l'architecture *AVE* se base sur des extracteurs de caractéristiques génériques : *VGG19* [144] pour les séquences d'images RVB et *VGGish* [72] pour le signal audio. Tout comme proposé dans [157], les paramètres de l'architecture sont estimés en minimisant la fonction de perte *CrossEntropy*, en utilisant la méthode d'optimisation *Adam* sur des lots de 32 segments d'une durée de 5s. Le nombre d'*epochs* est fixé à 500 avec un taux d'apprentissage de 0.0001.

Les performances de ces architectures sont comparées avec le score d'exactitude, de précision et de rappel. Une nouvelle fois, nous calculerons des scores moyens sur une 5 répartitions et tirages aléatoires d'ensembles différents.

En février 2023, dans [108], les auteurs ont proposé, tout comme nous [103], une architecture fondée sur l'extracteur *I3D*. Ils exploitent également un ensemble de crop dans les images afin de définir un processus de *Hard Attention* grâce à un apprentissage par renforcement. Cela permet de ne plus nécessiter l'annotation de localisation de la violence dans les données d'entraînement et donc de gagner en robustesse vis-à-vis de la position à laquelle se déroule la violence. Sur la base RWF-2000, les performances sont meilleures avec un taux de reconnaissance de 90.4%. Ce travail a été présenté trop récemment pour que nous ayons pu exploiter l'architecture proposée sur notre base *R2N*. Ce travail constitue une suite évidente qui sera menée dans les mois à venir. Cette même architecture sera également fusionnée à l'architecture audio que nous proposons et pourrait mener à des résultats prometteurs.

4.6 Évaluation des architectures uni-modales

Dans cette section, nous présentons l'évaluation les architectures uni-modales (*Audio*, *Vidéo* et *VidéoCrop*) présentées en section 3.2 menée sur notre base de données *R2N*.

L'objectif de cette première phase est de se rendre compte de la pertinence de chaque mode de perception sur la tâche de reconnaissance des violences.

Paramètres des architectures

— Architecture *Audio* : la première couche récurrente de type LSTM comporte 512 unités. Les couches entièrement connectées suivantes sont composées de respectivement 256, 64 et 16 unités. Enfin, la sortie de cette dernière couche est envoyée à la couche de décision.

— Architecture *Vidéo* : les paramètres sont identiques à ceux utilisés lors de l'évaluation de cette architecture sur les bases de donnée de l'état de l'art 4.5.1, à savoir : 512 unités pour la couche LSTM et respectivement 256, 64 et 16 unités pour les couches entièrement connectées qui précédent la couche de décision. Nous pourrons remarquer que cette configuration est également identique à celle de l'architecture *Audio*.

— Architecture *VidéoCrop* : les 3 branches de caractéristiques vidéo sont traitées en parallèle avec des couches récurrentes de type LSTM de 512 unités. Après une combinaison par concaténation de ces branches parallèles, les couches entièrement connectées suivantes sont composées respectivement une nouvelle fois de 256, 64 et 16 unités et suivies de la couche de décision.

Paramètres d'apprentissage

Malgré la taille de notre jeu de données, nous avons fait le choix d'utiliser comme paramètres des différents extracteurs les paramètres fournis et estimés sur de larges jeux de données afin d'éviter les problématiques de sur-estimation. Ainsi, les paramètres de ces blocs d'extraction sont fixés et ne sont ni ré-appris ni adaptés lors des phases d'apprentissage. Ces blocs correspondent aux blocs colorés en nuances de rouge dans les architectures présentées dans la section 3.2. Les paramètres des architectures sont estimés en suivant la procédure d'optimisation détaillée en section 4.4.

Procédure d'évaluation

L'ensemble des évaluations que nous présentons ci-dessous seront des résultats moyens estimés sur 100 tirages. Nous commercerons par évaluer l'architecture *Vidéo* à l'aide du score d'exactitude en fonction des zones afin de comprendre la difficulté de perception due à la distance de l'action et de justifier ainsi l'intérêt de l'architecture *VidéoCrop*. Ensuite, nous présenterons les scores d'exactitude, de précision et de rappel ainsi que les matrices de confusions pour les 3 architectures uni-modales. Enfin, nous utiliserons les diagrammes de Venn entre les architectures *Audio* et *VidéoCrop* afin de visualiser les erreurs spécifiques et communes entre ces deux architectures.

4.7 Évaluation des architectures multi-modales en fonction de la combinaison

4.7.1 Influence du niveau de la combinaison

Dans cette nouvelle phase, nous proposons d'évaluer les architectures multi-modales en fonction du niveau de combinaison des modes : combinaison des décisions (Architecture *Décisions*), combinaison tardive (Architecture *Tardive*) et combinaison moyenne

(Architecture *Moyenne*). L'objectif de cette évaluation est de mieux appréhender les capacités des architectures à modéliser des cohérences et des complémentarités entre les signaux audio et vidéo en fonction des niveaux de combinaison. Pour rappel, les algorithmes multi-modaux ne concernent que des algorithmes combinant des caractéristiques audio avec des caractéristiques vidéo présentés dans la section 3.2.3.

Paramètres des architectures

— Architecture *Décisions* : le premier niveau de combinaison que nous proposons est la combinaison au niveau des décisions. Les décisions sont reprises des architectures *Audio* et *VidéoCrop*. Les paramètres des deux branches correspondent donc exactement à ceux définis en section 4.6 (Évaluation des architectures uni-modales)

— Architecture *Tardive* : le deuxième niveau de combinaison que nous proposons est la combinaison à un niveau tardif. Dans cette architecture, les traitements des signaux audio et vidéo sont dans un premier temps disjoints avec des architectures s'inspirant des architectures *Audio* et *VidéoCrop*. La combinaison des projections de chaque branche est réalisée sur l'antépénultième couche de ces architectures. Cette combinaison est ensuite traitée par une couche entièrement connectée composée de 16 unités, suivie de la couche de décision.

— Architecture *Moyenne* : Le troisième et dernier niveau que nous proposons est la combinaison à un niveau moyen. Plus précisément, la combinaison est réalisée après les couches LSTM de chaque branche audio et vidéo. Après cette combinaison, les couches entièrement connectées sont identiques à celles que nous avons déjà rencontrées, à savoir qu'elles sont composées de 256, 64 et 16 unités ainsi que d'une couche de décision.

Paramètres d'apprentissage

Par définition, les paramètres de l'architecture *Décisions* sont des paramètres dont les branches audio et vidéo sont estimées indépendamment l'une de l'autre. Ils correspondent alors aux paramètres des architectures *Audio* et *VidéoCrop* estimés précédemment. Pour les deux architectures suivantes, *Tardive* et *Moyenne*, hormis les paramètres des extracteurs de caractéristiques qui sont fixés, les paramètres de ces deux architectures sont estimés en suivant l'optimisation décrite en section 4.4.

Procédure d'évaluation

Ces architectures seront évaluées avec les scores d'exactitude, de précision et de rappel ainsi qu'avec les matrices de confusions. De plus, nous utiliserons les diagrammes de Venn entre les architectures *Audio*, *VidéoCrop* et *Tardive* ainsi qu'entre les architectures *Audio*, *VidéoCrop* et *Moyenne* afin de visualiser les erreurs spécifiques et communes entre des architectures uni-modales et différents niveaux de combinaisons d'architectures multi-modales. Comme précédemment, l'ensemble de ces résultats sont des moyennes obtenues sur 100 différents tirages d'ensembles.

4.7.2 Influence de la technique utilisée pour la combinaison

Nous proposons d'évaluer le type de mise en œuvre de la combinaison des caractéristiques audio et vidéo : combinaison par concaténation (Architecture *Concaténation*), combinaison par mécanisme à porte (Architecture *Porte*) et combinaison par attention croisée (Architecture *Attention*). L'objectif de cette quatrième phase d'évaluation est de déterminer quelle stratégie permet une combinaison la plus optimale au sens des scores

de reconnaissance. Pour cela, nous avons focalisé cette évaluation pour un seul niveau de combinaison, celui a un niveau moyen comme défini en section 4.7.

Paramètres des architectures

— Architecture *Concaténation* : la première combinaison que nous proposons est la combinaison par concaténation. Dans cette architecture, les sorties des couches LSTM des branches audio et vidéo sont combinées par concaténation selon l'équation 1.2. Ensuite, cette combinaison est traitée par les couches entièrement connectées de 256, 64 et 16 unités ainsi que la couche de décision. Les scores de cette architecture reprendront donc exactement ceux obtenus avec l'architecture *Moyenne* décrite dans la section précédente.

— Architecture *Porte* : la deuxième combinaison que nous proposons est celle de la combinaison par mécanisme à porte. Comme la précédente architecture, cette combinaison reçoit en entrée les sorties des couches LSTM des branches audio et vidéo. Le nombre d'unités pour cette couche de combinaison est de 512 unités. Elle est suivie par des couches entièrement connectées de 256, 64 et 16 unités ainsi que la couche de décision. Pour rappel, l'objectif de cette combinaison est de sélectionner les caractéristiques les plus pertinentes sur chacune des branches audio et vidéo pour la prise de décision réalisée par les couches entièrement connectées suivantes.

— Architecture *Attention* : La troisième et dernière combinaison que nous proposons est la combinaison par attention croisée. Tout comme les deux précédentes architectures, la combinaison est réalisée avec les sorties des couches LSTM des branches audio et vidéo. La sortie de cette couche d'attention est de 3072 unités (5×512) représentant la concaténation des attentions entre la branche audio et la branche vidéo décrit dans l'équation 3.10 de la section 3.2.3.2.3. Pour rappel, l'objectif de cette combinaison est de fournir aux couches entièrement connectées suivantes des caractéristiques présentant des corrélations entre chacune des branches relevées par des fonctions d'attention. Cette couche de combinaison est ensuite traitée par trois couches entièrement connectées de 256, 64 et 16 unités ainsi que la couche de décision.

Paramètres d'apprentissage

Les paramètres utilisés pour l'apprentissage seront exactement les mêmes que ceux utilisés pour l'architecture *Moyenne*, c'est-à-dire celle définie en section 4.4.

Procédure d'évaluation

Comme pour l'évaluation des différents niveaux de combinaison, les différentes mises en œuvre de combinaison seront évaluées sur 100 tirages avec les scores d'exactitude, de précision et de rappel ainsi qu'avec des matrices de confusions. De plus, nous utiliserons les diagrammes de Venn entre les architectures *Audio*, *VidéoCrop* et *Concaténation*, entre les architectures *Audio*, *VidéoCrop* et *Porte* ainsi qu'entre les architectures *Audio*, *VidéoCrop* et *Attention* afin de visualiser les erreurs spécifiques et communes entre des architectures traitant des caractéristiques uni-modales et différents types de combinaisons de caractéristiques dans le cas d'architectures multi-modales.

4.8 Architectures multi-modale et apprentissage dépendant des modes

Enfin, nous proposons comme quatrième et dernière phase d'évaluation, de comparer les scores de reconnaissance d'une même architecture multi-modale entraînée selon des stratégies d'apprentissage différentes.

Les travaux de Xiao *et al.* dans [181] ont relevé que la convergence lors de la phase d'apprentissage pouvait être influencée par la nature même des modes et de leurs complexités respectives à représenter l'espace des observations. Plus précisément, un mode par son contenu peut influencer la convergence de l'algorithme au détriment d'un autre : un premier risque est une optimisation atteinte en ayant considéré un mode plus qu'un autre.

Dans notre contexte, la violence peut parfois n'être perçue que par le mode audio ou que par le mode vidéo : dans chacun de ces deux cas, la cohérence entre audio et vidéo n'existe plus et nécessite que notre architecture multi-modale soit capable de définir un espace de représentation où la violence peut être décrite par l'audio seule, la vidéo seule et l'audio et la vidéo conjointement. Malheureusement dans notre base de données, nous n'avons pas l'équilibre de ces différentes perceptions. L'équilibre que nous avons pu mettre en œuvre, présenté en section 4.1.2, a consisté, de ce point de vue, à équilibrer entre elles les perceptions audio seule et vidéo seule, sans pouvoir les équilibrer avec la perception audio et vidéo de la violence. Ayant peu de données avec un seul mode où la violence est perçue, nous proposons de considérer cette information dans notre phase d'apprentissage, contrairement à Jaafar *et al.* dans [77] qui considèrent cette information en entrée de leur architecture. L'objectif est que ces données ne viennent pas perturber l'optimisation de notre architecture, à défaut de ne pouvoir correctement les représenter :

— Architecture *Standard* : cette stratégie d'apprentissage est la plus classique, c'est-à-dire sans considération particulière vis-à-vis des caractéristiques et des modes utilisés lors de l'estimation des paramètres ;

— Architecture *Contrainte* : cette seconde stratégie d'apprentissage reprend également les paramètres définis en section 4.4 et intègre la particularité de désactiver l'apprentissage sur la branche audio ou sur la branche vidéo lorsque la violence n'est pas perçue sur l'un ces modes.

Nous avons réalisé cette dernière étude en ne considérant qu'une seule architecture multi-modale, correspondant à une combinaison à un niveau moyen par concaténation. Bien que la phase d'apprentissage utilise une information *a priori* sur la présence ou non de la violence sur l'un ou l'autre des deux modes, l'inférence des prédictions ne considérera pas cette information lors de la phase de test, celle-ci ne pouvant être disponible directement dans un cadre opérationnel.

Procédure d'évaluation

Réalisées sur 100 tirages, les différentes stratégies d'apprentissage seront évaluées selon les scores d'exactitude, de précision et de rappel ainsi qu'avec les matrices de confusions. De plus, nous utiliserons les diagrammes de Venn entre les architectures *Audio*, *VidéoCrop* et *Standard* (c'est-à-dire Architecture *Moyenne*) ainsi qu'entre les architectures *Audio*, *VidéoCrop* et *Contrainte* afin de comparer les erreurs spécifiques et communes entre les architectures vis-à-vis de ces deux stratégies d'apprentissage.

4.9 Analyse qualitative

Afin d'avoir une compréhension approfondie des résultats de reconnaissance de violence, nous proposons d'évaluer notre meilleure architecture qualitativement grâce à la méta-annotation que nous avons produite et présentée dans la section 4.1. Cette architecture sera donc définie à la suite des résultats quantitatifs. L'ensemble des diverses analyses qualitatives seront réalisées avec le score d'exactitude et les matrices de confusion (moyennes sur les 100 tirages) selon différents critères de perception que nous avons sélectionnés et que nous présentons ci-dessous.

Analyse en fonction de la perception des violences sur les signaux

L'objectif de cette évaluation est de confronter les résultats de reconnaissance de violence en fonction des modes de perception audio et vidéo. L'intérêt est de pouvoir apprécier l'impact éventuel des modes de perception qui ont permis d'établir notre annotation "globale" utilisée pour lors des phases d'apprentissage et d'évaluation.

Pour rappel, l'annotation globale est réalisée sur un segment de 5 secondes et ce segment est considéré comme "Violent" si la présence d'une violence a été perçue sur au moins un segment de 2 secondes composants ce segment de 5 secondes. Cette procédure d'annotation a été appliquée quel que soit le mode de perception.

Analyse en fonction de la distance aux capteurs

Cette évaluation nous permet d'analyser les résultats en fonction de la distance de la violence au capteur. En effet, comme illustré dans la figure 3.3 du Chapitre 3, les salles mesurant entre 8 et 10 mètres de long, une scène de violence proche des capteurs est plus facilement perçue qu'une scène de violence éloignée. Cela se manifeste au travers de la résolution plus ou moins élevée des éléments de la scène de violence en fonction de sa distance à la caméra. En ce qui concerne le signal sonore, cet aspect est plus délicat à appréhender car il existe une relation physique entre la distance d'une source sonore et l'amplitude de son signal : en effet, un son avec une faible ou une forte dynamique ne peut être associé à une distance sans utilisation de plusieurs microphones ou sans l'ajout d'*a priori*). Cette évaluation s'appuie sur les 3 zones de salles que nous avons définies en section 3.1.4 et que nous précisons ci-dessous :

— Zone 1 : scène de violence proche du capteur (de 0 à 3 mètres),
— Zone 2 : scène de violence à moyenne distance (de 3 à 6 mètres),
— Zone 3 : scène de violence loin du capteur (de 6 à 9 mètre).

Analyse en fonction du degré de violence perçue sur la vidéo

Cette analyse porte sur la description des résultats de reconnaissances en fonction du degré de violence annoté sur le signal vidéo. Pour rappel ce degré de violence a été défini sur une échelle de 0 à 5 en fonction de la perception de celle-ci sur le signal vidéo (sans audio) sur des segments de 2 secondes. Ainsi sur un segment de 5 secondes apparaissent 3 valeurs consécutives de degré de violences. Afin d'appréhender cette caractéristique sur une durée de 5 secondes, nous avons fait le choix de moyenner ces 3 valeurs pour obtenir une seule valeur sur les 5 secondes et d'établir une nouvelle échelle de 0 à 3 où :

— le degré 1 représente une violence avec un niveau moyen de violence entre 0 et 1,66 ;
— le degré 2 représente une violence avec un niveau moyen de violence entre 1,66 et 3,33 ;
— le degré 3 représente une violence avec un niveau moyen de violence entre 3,33 et 5.

Analyse en fonction du degré d'occultation sur la vidéo

Comme décrit auparavant, l'observation vidéo peut être altérée par nombres d'occultations en fonction de la densité de passagers, à laquelle s'ajoute l'éloignement de la caméra de la scène à analyser. Nous proposons donc d'analyser nos résultats de reconnaissance en fonction du degré d'occultation que nous avons émis lors de l'annotation du signal vidéo. Comme pour le degré de violence, ce degré d'occultation a été défini sur une échelle de 0 à 5 en fonction de la perception de celle-ci sur le signal vidéo sur des segments de 2 secondes. Afin de pouvoir considérer cette caractéristique sur 5 secondes, nous avons, comme pour le degré de violence, moyenné les 3 valeurs consécutives de degré d'occultation pour obtenir une seule valeur sur les 5 secondes. Tout comme pour les degrés d'occultations nous avons fait le choix d'établir une nouvelle échelle de 0 à 3 où :

— le degré 1 représente une violence avec un niveau moyen de violence entre 0 et 1,66 ;

— le degré 2 représente une violence avec un niveau moyen de violence entre 1,66 et 3,33 ;

— le degré 3 représente une violence avec un niveau moyen de violence entre 3,33 et 5.

Analyse en fonction de la durée de perception des violences en fonction des signaux audio et vidéo

Selon notre procédure d'annotation décrite en section 4.1.1, les segments de 5 secondes annotés comme "Violence" peuvent contenir une durée minimale de violence seulement sur un mode. De plus, cette durée de perception de violence peut différer selon les modes (Violence sur Audio & Vidéo, Violence sur Audio et Violence sur Vidéo). L'objectif de cette évaluation est donc d'apprécier l'impact de la durée de perception des violences sur les résultats.

Ainsi, suivant notre stratégie d'annotation, un segment de signaux de 5s peut contenir entre 1s à 5s de violence. Nous avons donc défini trois catégories de durée de violence perçue dans 5s de signaux et établi une échelle de durée de 1, à 3, où :

— la durée 1 représente la durée de violence la plus faible, soit entre 1s et 1,66s.

— la durée 2 représente une durée de violence entre 1,66s et 3,33s.

— la durée 3 représente la durée de violence la plus longue, soit entre 3,33s et 5s.

L'objectif est ici d'analyser l'impact de ces durées en fonction du mode de perception de la violence (audio seul, vidéo seul et simultanément sur audio et vidéo).

Conclusions

Ce chapitre méthodologique présente la mise en œuvre de nos travaux d'évaluation de nos architectures. Dans un premier temps, nous avons abordé les traitements que nous avons réalisés sur nos données pour obtenir un jeu de données exploitable, puis nous avons présenté les métriques utilisées pour l'évaluation et les paramètres d'apprentissage. Enfin, une présentation des différentes évaluations que nous avons réalisées pour sélectionner la meilleure architecture a été faite. L'architecture avec les meilleurs résultats sera utilisée pour approfondir la compréhension des résultats avec des caractéristiques spécifiques de notre jeu de données.

Chapitre 5

Application à la détection des violences dans un environnement ferroviaire

Ce chapitre est dédié à la présentation des résultats des diverses expérimentations que nous avons décrites au chapitre précédent. Nous commencerons par présenter en section 5.1 quelques statistiques relatives aux 100 répartitions et tirages aléatoires de notre base de données utilisée l'évaluation de nos modèles. Dans un second temps, nous présenterons en section 5.2 les résultats préliminaires nous donnant des informations relatives quant à nos travaux vis-à-vis de l'état de l'art. Enfin, en section 5.3, nous présenterons les résultats de nos diverses expérimentations dédiées à l'évaluation de nos architectures neuronales sur notre base de données *R2N*.

5.1 Analyse des données *R2N* sur 100 répartitions et tirages aléatoires

Ce chapitre présente les résultats sur les 100 répartitions et tirages aléatoires décrits en section 4.1.2qui nous fournissent 100 ensembles différents d'observations audio et vidéo de 5s. Nous rappelons que ces différents tirages ont été réalisés après une opération d'équilibrage faite à chaque nouvelle répartition des scénarios. L'objectif est de vérifier, sur 100 répartitions et tirages aléatoires, si la procédure d'extension de l'annotation de 2s à 5s, modifie ou non le contenu vis-à-vis des données annotées sur 2 secondes présentées en section 3.1.6.

Ainsi, les tableaux 5.1 et 5.2 présentent sur 100 tirages, les moyennes et les écart-types du nombre de segments de 5s annotés "Violence" dans les ensembles d'*Entraînement*, de *Validation* et de *Test*, ceci en fonction respectivement des degrés d'occultation et de violence, et de la durée de la perception. Le tableau 5.1 est obtenu via l'annotation des segments issus uniquement du signal vidéo sur 5 secondes. Le tableau 5.2 regroupe les informations issues des annotations *audio*, *vidéo*, et "globale", établies sur 5 secondes. Nous rappelons que ces différents degrés sur 5 secondes sont des moyennes des degrés annotés sur 2 secondes tel que cela est décrit en section 4.9.

Il apparaît clairement que les tableaux 5.1 et 5.2 présentent un déséquilibre du nombre de segments entre les différents degrés d'occultation, de violence et de durées de perception quel qu'en soit le mode. En comparant le tableau 5.1 avec les tableaux d'analyses 3.7 et 3.8 de la section 3.1.6 portant sur des segments de 2 secondes, nous constatons que les déséquilibres "naturels" présents originellement dans le jeu de données sont conservés en moyenne sur les 100 tirages avec des segments et une annotation portée de 2s à 5s.

	Entraînement	*Validation*	*Test*
Occultation degré 1	368 (± 48,6)	61,9 (± 38,7)	60,0 (± 34,8)
Occultation degré 2	1002 (± 110)	164 (± 79,4)	173 (± 91,3)
Occultation degré 3	504 (± 43,3)	81,3 (± 38,0)	79,7 (± 30,6)
Violence degré 1	159 (± 27,8)	27,8 (± 19,0)	26,5 (± 19,1)
Violence degré 2	917 (± 80,5)	152 (± 55,9)	154 (± 64,4)
Violence degré 3	799 (± 66,9)	128 (± 53,6)	132 (± 54,6)

TABLE 5.1 – Moyennes et écart-types du nombre de segments de 5s annotées "Violence" en fonction du degré de violences ou d'occultation établis sur 5 secondes (degré 1 : valeur comprise entre 0.00 et 1.66, degré 2 : valeur entre 1.66 et 3.32 et degré 3 : valeur entre 3.32 et 5.0). Ces résultats sont issus des 100 tirages aléatoires des ensembles d'*Entraînement*, de *Validation* et de *Test*.

Nous avons en moyenne l'occultation de degré 2 (degré moyen entre 1.66 et 3.32) nettement supérieur au degré 1 et 3. Dans le tableau 3.7 le degré d'occultation le plus représenté est de niveau 3, ce qui est cohérent avec notre intervalle de degrés 2 sur 5 secondes. Ensuite apparaissent les niveaux 2 et 4 : Le niveau 2 correspond encore à ce même intervalle de [1.66 - 3.32], alors que le niveau 4 tend à s'être dissout en partie dans cet intervalle par l'effet "mathématique" de la moyenne sur 5s.

En ce qui concerne les degrés de violence, le premier degré moyen de violence est beaucoup moins représenté que ceux du deuxième et troisième degrés en moyenne sur les 100 répartitions (Tableau 5.1). Ceci reste cohérent avec les données annotées sur 2 secondes (Tableau 3.8) : la violence la moins représentée est celle de niveau 1 correspondant parfaitement à l'intervalle [0 - 1.66] dans le tableau 5.1. Les valeurs moyennes de violences sur 5s pour les intervalles supérieurs sont bien supérieurs comme peuvent l'être les données observées sur 2 secondes avec une légère baisse pour les plus "fortes" violences (ce que l'on retrouve également dans le tableau 5.1).

Le tableau 5.2 présente la durée de perception moyenne de la violence sur les 100 répartitions. Au regard du signal audio, la perception de violence est perçue principalement sur une durée entre 3,32s et 5s, à la différence de la vidéo dont les différences entre les durées sont plus modérées, avec une supériorité pour les perceptions les plus longues et une minorité pour la perception de durée moyenne. Enfin, on peut constater que les durées moyennes de la violence perçue mutuellement sur l'audio et la vidéo sur 5 secondes, suit les mêmes durées moyennes que les violences perçues sur le mode sonore. Ainsi, sur 5 secondes, les plus longues durées en moyenne des violences définies par l'annotation globales sont induites principalement par les longues durées de présence de violence sur le signal sonore.

5.2 Résultats préliminaires sur l'état de l'art

5.2.1 Test de notre architecture vidéo sur 3 bases de données de l'état de l'art

Cette section présente dans un premier temps les résultats obtenus lors de l'application de notre architecture vidéo "de base" appliquée à trois bases de données de la littérature. Les résultats sont regroupés dans le tableau 5.3. Ce tableau présente les scores d'exactitude, de précision et de rappel obtenus. Nous rappelons également que ces résultats sont calculés sur les ensembles fournis par les auteurs du jeu de données. Ces résultats ne sont

		Entraînement	*Validation*	*Test*
Audio	Durée perception 1	202 (± 22,6)	32,3 (± 17,1)	29,1 (± 15,3)
	Durée perception 2	370 (± 30,9)	59,7 (± 23,2)	57,9 (± 22,6)
	Durée perception 3	1303 (± 112)	216 (± 81,2)	226 (± 95,2)
Vidéo	Durée perception 1	659 (± 62,5)	107 (± 40,8)	109 (± 51,5)
	Durée perception 2	446 (± 30,3)	73,4 (± 20,8)	75,3 (± 23,8)
	Durée perception 3	770 (± 73,6)	127 (± 58,1)	129 (± 59,6)
Globale	Durée perception 1	193 (± 20,8)	31,5 (± 14,3)	29,5 (± 14,0)
	Durée perception 2	304 (± 23,5)	49,4 (± 18,0)	46,9 (± 17,2)
	Durée perception 3	1521 (± 127)	250 (± 92,2)	258 (± 107)

TABLE 5.2 – Moyennes et écart-types du nombre de segments de 5s annotées "Violence" en fonction de la durée des violences estimée sur 5 secondes (durée 1 : durée comprise entre 0,00s et 1,66s, durée 2 entre 1,66s et 3,32s et durée 3 entre 3,32s et 5,0s). Ces résultats sont issus de 100 tirages aléatoires des ensembles d'*Entraînement*, de *Validation* et de *Test*.

donc pas des moyennes contrairement aux analyses que nous produirons sur le jeu de données que nous introduisons.

On peut observer que notre architecture *Vidéo* atteint des performances proches de l'état de l'art sur le jeu de données *Hockey Fight* et *RWF-2000* avec respectivement des scores d'exactitude de 93,0% et de 86,0% pour notre architecture contre 98,0% et 84,5% pour [22]. Pour le jeu de données *Surveillance Camera Fight*, notre architecture *Vidéo* permet de dépasser les performances de l'état de l'art avec un score d'exactitude de 90,0% pour un SOTA à 72,0% dans [189]. Nous avons fait le choix de fixer la partie *I3D* de l'architecture, réduisant ainsi le nombre de paramètres à apprendre à ceux des LSTM et des couches entièrement connectées. Ainsi, sur la base *Surveillance Camera Fight* de taille plus petite que *Hockey Fight* et *RWF-2000*, nous obtenons une propriété de généralisation plus grande et finalement de meilleures performances.

Ces premiers résultats montrent que l'architecture vidéo que nous utilisons est pertinente pour cette tâche de reconnaissance de violences par analyse vidéo, fournissant des scores d'exactitude quasi constants à travers des bases de données pouvant être plus ou moins variées et des scores pouvant être supérieurs à ceux du SOTA.

5.2.2 Test de 3 architectures de l'état de l'art sur notre base de données *R2N*

Le tableau 5.4 regroupe les scores moyens sur 5 répartitions de l'exactitude, de la précision et du rappel de trois architectures sélectionnées dans la communauté testées sur notre jeu de données (*R2N*) : RWF-2000 [22] avec et sans flux optique et AVE [157]. Nous rappelons que les deux premières architectures ne traitent que le signal vidéo et ont la particularité d'être dédiées à la détection de violence. La troisième, à l'origine non orientée

Jeu de données	SOTA	Architecture *Vidéo*		
	Exactitude	Exactitude	Rappel	Précision
Hockey Fight	98,0% [22]	93,0%	94,0%	92,1%
Surveillance Camera Fight	72,0% [189]	90,0%	93,3%	87,5%
RWF-2000	84,5% [22]	86,0%	85,0%	86,7%

TABLE 5.3 – Résultats sur l'ensemble de test des jeux de données *Hockey Fight*, *Surveillance Camera Fight*, *RWF-2000* de notre architecture *Vidéo*.

spécifiquement pour la reconnaissance de violence, considère le flux audio et vidéo pour réaliser une tâche de classification de scènes.

Tout d'abord, les résultats sur 5 répartitions nous indiquent que les architectures retenues ne montrent pas, sur notre base ferroviaire, des performances équivalentes à celles obtenues dans les articles correspondants : il semble difficile de transférer les méthodes actuelles d'une manière générale à des contextes fort variés, et plus précisément au contexte ferroviaire dont l'environnement présente des spécificités. Plus précisément, en considérant la première architecture *RWF-2000* qui utilise en entrée une séquence d'images couleur, nous observons un score d'exactitude moyen de 62,0% alors que sur le jeu de données auquel elle est associée, les performances atteignaient 84,5%. Pour la seconde architecture *RWF-2000* qui utilise une séquence d'images RVB et une séquence de flots optiques (*RVB+FO*), le score d'exactitude moyen atteint 63,2% alors que sur le jeu de données associé les performances atteignaient 87,2%. En considérant maintenant l'architecture *AVE* qui se base sur le signal audio et vidéo, nous remarquons que le score d'exactitude moyen (71,8%) est bien plus élevé que celui atteint pour les architectures précédentes qui n'exploitent que le signal vidéo. Ces performances nous confirment que la combinaison des signaux audio et vidéo est pertinente pour la tâche de reconnaissance de violences dans un environnement ferroviaire. Cette architecture montre une précision de 77,7% mais témoigne d'un score de rappel plus faible à 68,4%.

5.3 Résultats quantitatifs de nos architectures sur la base de données *R2N*

5.3.1 Résultats des architectures uni-modales audio et vidéo

Les résultats obtenus avec les architectures uni-modales sont présentés dans le tableau 5.5. Ce tableau présente les scores d'exactitude, de précision et de rappel de trois architectures uni-modales proposées en section 3.2.1 et 3.2.2 dont les paramètres ont été estimés suivant la procédure décrite en section 4.6.

Premièrement, en observant les performances de l'architecture uni-modale *Vidéo*, on observe que cette architecture obtient un score d'exactitude moyen sur 100 répartitions de 61,1% (± 5,18%), plus faible que ceux obtenus sur les jeux de données de la communauté (Tableau 5.3). Plus finement, les performances présentées au travers de la matrice de confusion (Figure 5.1a), montrent que l'architecture n'est pas précise, c'est-à-dire de nombreuses scènes sans violence sont classées comme des scènes contenant des violences : comme introduit en 3.2.2, ces mauvaises performances s'expliquent par la difficulté de l'extracteur de caractéristiques *I3D* à extraire correctement les mouvements quelle que soit la distance à laquelle ils se produisent. N'ayant pas été appris avec ce type de va-

Architecture de la communauté	Exactitude	Rappel	Précision
RWF-2000 *RVB*	62,0% (± 6,80%)	62,0% (± 17,9%)	64,1% (± 6,28%)
RWF-2000 *RVB+FO*	63,2% (± 8,24%)	68,8% (± 12,1%)	66,9% (± 9,76%)
AVE	71,8% (± 3,10%)	68,4% (± 10,8%)	77,7% (± 6,20%)

TABLE 5.4 – Résultats des architectures RWF-2000 [22] et AVE [157] sur notre base de données *R2N* pour 5 répartitions aléatoires. *RWF-2000* considère soit une séquence d'images (RVB), soit une séquence d'images et un flux optique ($RVB + FO$). AVE considère une séquence d'image et un signale audio

Architectures	Exactitude	Rappel	Précision
Vidéo	61,1% (± 5,18%)	66,9% (± 17,5%)	64,8% (± 7,91%)
Vidéo Crop	73,4% (± 5,43%)	71,8% (± 11,4%)	78,8% (± 8,28%)
Audio	83,1% (± 4,84%)	80,0% (± 9,98%)	88,0% (± 5,85%)

TABLE 5.5 – Résultats moyens sur les 100 répartitions pour les trois architectures uni-modales développées.

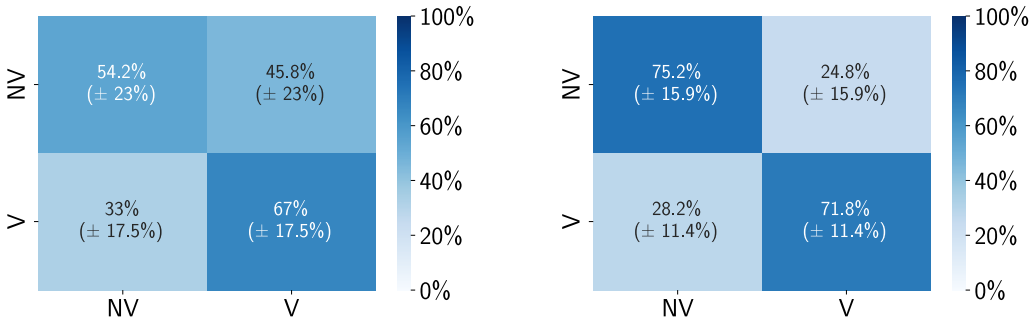
	Violence en Zone 1	Violence en Zone 2	Violence en Zone 3
Vidéo	64.0%	25.0%	5.0%

TABLE 5.6 – Illustrations de la décroissance du score d'exactitude de reconnaissance de violence de l'architecture uni-modale *Vidéo* en fonction des zones de jeu de la violence. Scores obtenus pour une répétition.

riabilité, il est difficile pour ce modèle d'extraire de "bonnes" caractéristiques quelle que soit la zone de jeu des violences. En conséquence, le modèle apprend mal les violences distantes (avec une faible définition), qui de plus viennent pénaliser l'apprentissage des mouvements observés au premier plan. Ces lacunes en cascades sont illustrées à travers le résultat d'une répétition où le score d'exactitude de reconnaissance de violence décroît en fonction de l'éloignement de la scène de violence (Tableau 5.6). La répartition des violences en zones pour les salles EXT, SB et SH présentées dans le tableau 3.6 [1] indiquent de plus que le nombre de scènes en zones 3, pouvant être utilisées dans les 100 répartitions, est plus faible que dans les deux autres zones (c'est-à-dire scènes pouvant être moins bien représentées lors des phases d'apprentissage). Néanmoins, en considérant les scènes en zone 1 et 2, les scores montrent que même avec une représentation de données équivalentes (avec une légère supériorité pour la zone 2), l'architecture peine sur la zone la plus éloignée.

Avec la prise en compte de la profondeur de champs, c'est-à-dire en fournissant des images permettant de mieux uniformiser la définition des mouvements dans l'image quelle que soit la zone, l'architecture uni-modale *VidéoCrop* permet d'améliorer les résultats de reconnaissance. L'amélioration permet d'atteindre un score d'exactitude moyen sur

1. Bien que ce tableau présente des répartitions de segments annotées sur 2 secondes, nous estimons que ces valeurs restent représentatives de notre base de données même dans le cas de traitement de segments de 5s.



(a) Sans crop. (b) Avec crop.

FIGURE 5.1 – Matrice de confusion des architectures uni-modale *Vidéo* (a) et *VidéoCrop* (b) sur les 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.

100 tirages de 73,4% (± 5,43%). Enfin, la matrice de confusion, nous confirme que les performances de cette architecture sont plus équilibrées et plus élevées entre les deux classes (Figure 5.1b).

Maintenant, au regard des performances de l'architecture uni-modale *Audio* dans le tableau 5.5, on peut constater que le traitement de ce signal est pertinent pour reconnaître des violences dans notre contexte d'étude puisqu'il obtient des scores moyens supérieurs à l'architecture uni-modale *VidéoCrop* : de 83,1% (± 4,84%). Ce gain de performance est similaire sur le score de précision et de rappel. L'analyse de la matrice de confusion de cette architecture (Figure 5.2), permet d'observer que les résultats de l'architecture *Audio* sont équilibrés et plus élevés pour les deux classes vis-à-vis de ceux obtenus avec les architectures traitant les signaux vidéo. Premièrement, on peut déduire de ces résultats que les paramètres de l'extracteur audio *OpenL3* semblent être adaptés à ce type d'environnement. Dans un second temps, ces performances peuvent s'expliquer par le fait que la discrimination sonore entre une scène sans violence *vs.* avec violence est moins sensible à l'environnement que celle obtenue par vidéo. En effet, ordinairement les passagers des salles font peu de bruit, cette situation pouvant être différente lors d'une scène de violence. Alors que d'un point de vu perception vidéo, une scène contenant des personnes se déplaçant rapidement peut être associée à tort à une scène de violence.

Pour aller plus loin, nous pouvons analyser les erreurs des architectures uni-modales *Audio* et *VidéoCrop* avec des diagrammes de Venn (Figure 5.3). Sur chacun des diagrammes, le cercle violet représente les erreurs commises par l'architecture *Audio* et le cercle jaune représente les erreurs commises par l'architecture *VidéoCrop*. Les erreurs communes entre les deux architectures sont représentées par la superposition des deux cercles. Au sein de chaque cercle, les chiffres indiqués représentent les erreurs moyennes sur 100 répartitions (auxquelles sont associés les écart-types). En analysant le nombre d'erreurs dans les cercles sans superposition, nous constatons que la majorité des erreurs sont des erreurs spécifiques à chaque architecture. Sur 100 répartitions, en moyenne seulement 15,0 erreurs de faux positifs et 30,4 erreurs de faux négatifs sont communes entre les architectures *Audio* et *VidéoCrop* (Figure 5.3a et 5.3b). Cette répartition des erreurs entre les architectures confirme l'intérêt de combiner les signaux dans un système de reconnaissance.

5.3.2 Niveau de combinaison

La deuxième phase d'évaluation de nos architectures consiste à évaluer nos architectures multi-modales dans un premier temps en fonction du niveau de la combinaison des branches audio et vidéo. Les résultats sont présentés dans le tableau 5.7. Ce tableau présente les scores d'exactitude, de précision et de rappel des trois architectures en fonctions

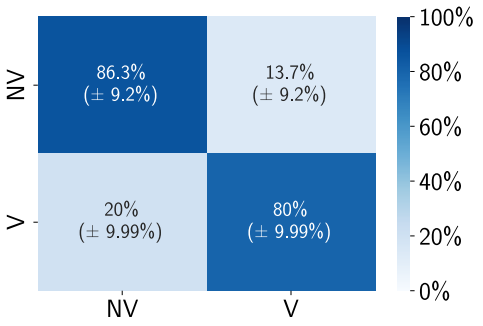
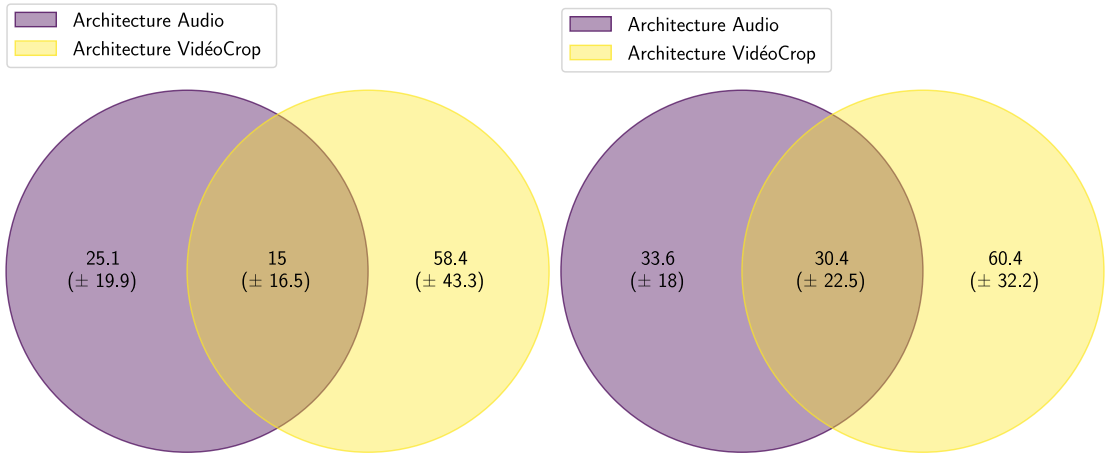


FIGURE 5.2 – Matrice de confusion de l'architecture uni-modale *Audio* sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.



(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

FIGURE 5.3 – Diagrammes de Venn des erreurs de type faux positif (a) et de type faux négatif (b) des architectures uni-modales *Audio* et *VidéoCrop* sur 100 répartitions.

du niveau de la combinaison : *Moyenne*, *Tardive* et *Décision*.

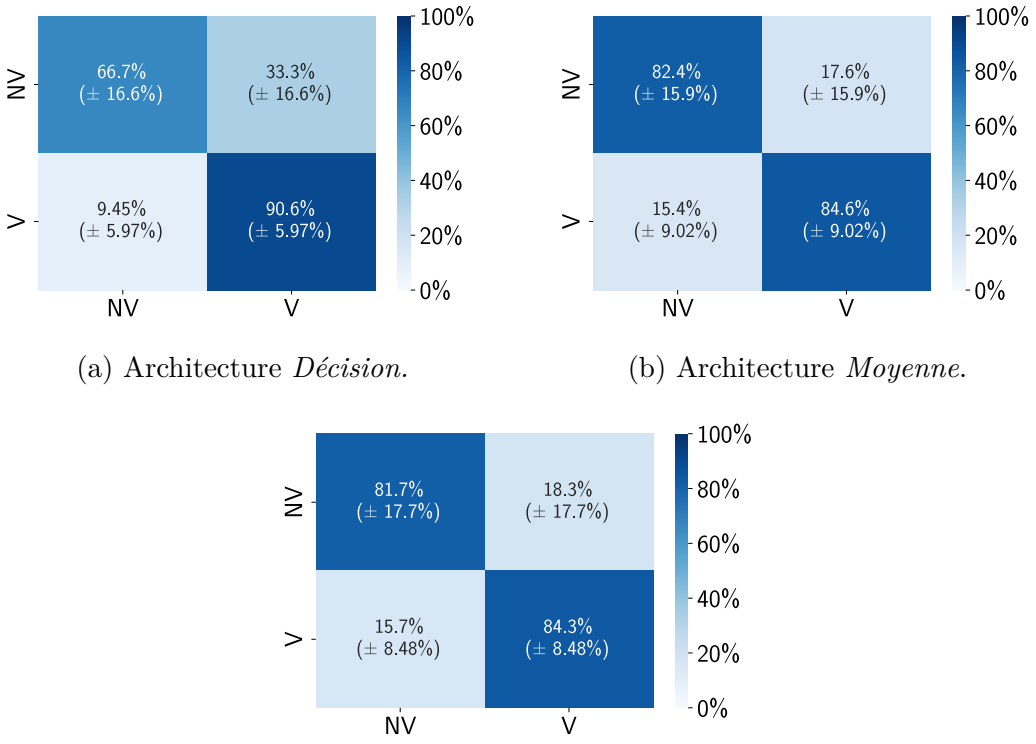
En comparant dans un premier temps l'architecture avec la combinaison la plus naïve, à savoir l'architecture *Décision*, avec les architectures uni-modales *Audio* et *VidéoCrop* précédemment vu (Tableau 5.5), on constate d'une manière générale que cette première architecture mutli-modale "simple" permet d'augmenter les scores vis-à-vis du modèle uni-modal *Vidéo*. Cependant, cette combinaison ne semble pas optimale puisque ses scores sont d'un point de vue général inférieurs à l'architecture uni-modale *Audio* (excepté le score de rappel). Plus précisément, l'observation de la matrice de confusion de cette architecture combinant naïvement les décisions (Figure 5.4a), montre que la reconnaissance des violences est plus élevée en termes de faux négatifs et de bonnes détections (FN & VP) que les architectures uni-modales *Audio* et *VidéoCrop* (avec un score de rappel de 90,6%). Par ailleurs, ce type d'architecture est beaucoup moins performant sur la reconnaissance des exemples sans violence au regard des taux de confusion de "Non Violence" en "Violence" et de celui de la bonne détection des "Non Violence" (FP & VN) avec un score de précision de 77,0%.

Dans un second temps, une analyse plus spécifique aux divers niveaux de combinaisons permet de montrer que l'apprentissage de la combinaison (Architecture *Moyenne* et Architecture *Tardive*) sont plus performantes (respectivement +4,11% et +3,62% du score d'exactitude) que la simple combinaison des décisions des architectures uni-modales (Architecture *Décision*). Ce gain de performances se manifeste principalement sur l'amélioration du taux de précisions.

Enfin, en analysant le score d'exactitude moyen des architectures apprenant la combinaison (Architecture *Moyenne* et Architecture *Tardive*), on peut observer que la combinaison à un niveau moyen est légèrement meilleure, 83,7% (± 6,67%), comparativement à la combinaison à un niveau tardif, 83,2% (± 7,25%). Ce gain de performance est semblable sur les scores de précision et de rappel. Malgré un gain léger, on peut donc conclure que

Architectures	Exactitude	Rappel	Précision
Décision	79,6% (± 6,61%)	90,6% (± 5,97%)	77,0% (± 7,89%)
Moyenne	83,7% (± 6,67%)	84,6% (± 9,01%)	86,3% (± 8,31%)
Tardive	83,2% (± 7,25%)	84,3% (± 8,48%)	85,9% (± 8,68%)

TABLE 5.7 – Résultats moyens sur les 100 répartitions pour les architectures développées en fonction du niveau de combinaison.



(a) Architecture *Décision*.

(b) Architecture *Moyenne*.

(c) Architecture *Tardive*.

FIGURE 5.4 – Matrice de confusion des architectures multi-modales en fonction du niveau de combinaison des modes. Résultats réalisés sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.

réaliser une combinaison tôt est une stratégie légèrement plus pertinente.

Pour aller plus loin, nous proposons d'analyser avec des diagrammes de Venn les erreurs de ces architectures. La figure 5.5 considère comme ensembles les erreurs des architectures *Audio*, *VidéoCrop* et *Moyenne*, et la figure 5.6 celle de l'architecture *Audio*, *VidéoCrop* et *Tardive*. Ces diagrammes sont des extensions des diagrammes de Venn des architectures uni-modales *Audio* et *VidéoCrop* (Figure 5.3). En effet, les sommes des cercles violets avec la superposition des cercles bleus dans la figure 5.5 ou la figure 5.6 est égale à la valeur dans les cercles violets dans la figure 5.3. De la même façon, les sommes des cercles jaunes avec la superposition des cercles bleus dans la figure 5.5 ou la figure 5.6 est égale à la valeur dans les cercles jaunes dans la figure 5.3. Enfin, les sommes des superpositions des cercles violet et jaune avec la superposition des cercles bleu dans la figure 5.5 ou la figure 5.6 est égale à la valeur dans la superposition des cercles violet et jaune dans la figure 5.3. Dans ces diagrammes de Venn en figures 5.5 et 5.6, on peut remarquer que la combinaison à un niveau moyen permet de faire moins de nouvelles erreurs que la combinaison à un niveau tardif : 12,73 *vs.* 15,23 erreurs FP et 3,39 *vs.* 3,94 erreurs FN. En observant les erreurs communes entre les architectures, on peut remarquer que l'architecture combinant les signaux à un niveau moyen commet plus d'erreurs audio (16,36 + 16,93 *vs.* 14,85 + 16,58), alors que l'architecture combinant les signaux à un niveau tardif fait plus d'erreur vidéo (10,5 + 5,02 *vs.* 14,05 + 5,54).

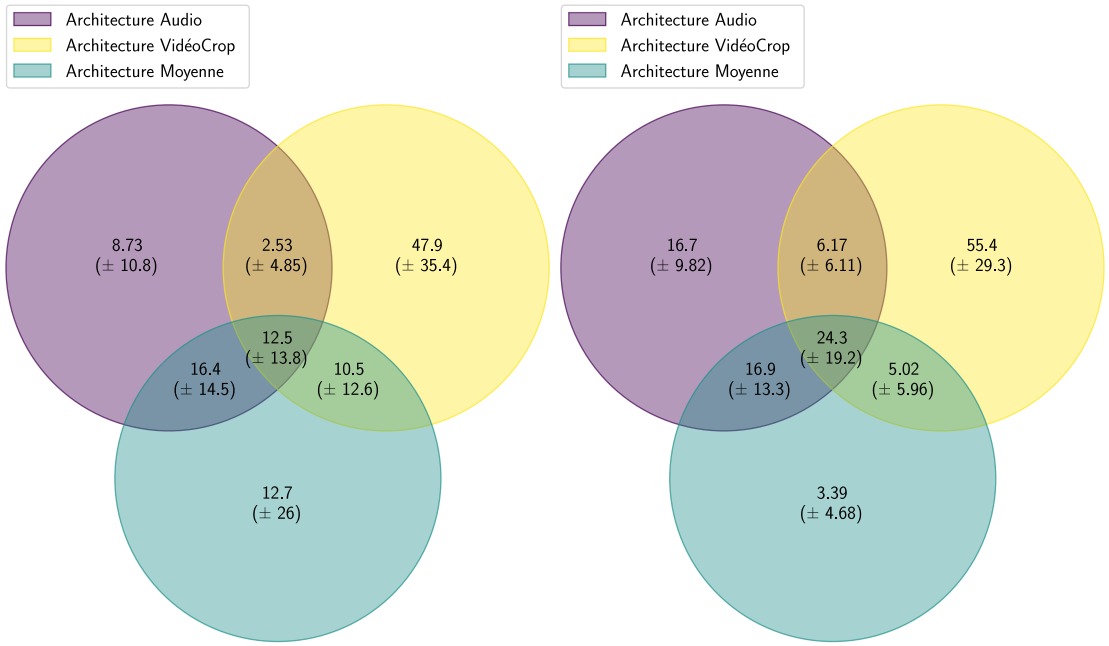
5.3.3 Stratégie de combinaison

La troisième phase d'évaluation quantitative de nos architectures a consisté à évaluer 3 stratégies de combinaison des branches audio et vidéo (*Concaténation*, *Porte* et *Attention*). Nous avons choisi d'effectuer ces évaluations de stratégies à un seul niveau de combinaison.

Au vu des résultats précédents, nous avons retenu le niveau de combinaison moyen.

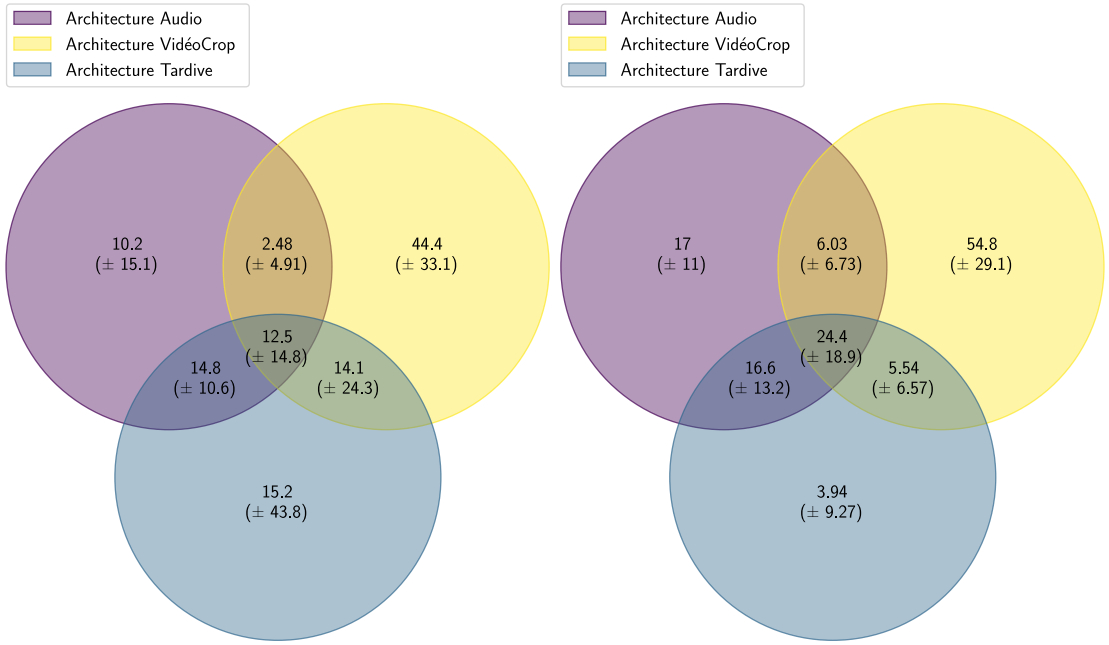
Les résultats présentés dans le tableau 5.8 sont les scores d'exactitude, de précision et de rappel des trois architectures en fonction des stratégies de combinaison proposées.

L'observation des scores d'exactitude du tableau 5.8 montre que la combinaison par concaténation (Architecture *Concaténation*) est plus performante que la combinaison par mécanisme à portes (Architecture *Porte*) et que la combinaison par attention croisée (Architecture *Attention*) (83,7% *vs.* 80,4% *vs.* 78,8%). Ce gain de performance est identique sur les scores de précisions de ces trois architectures. Cependant, sur les scores de rappels,



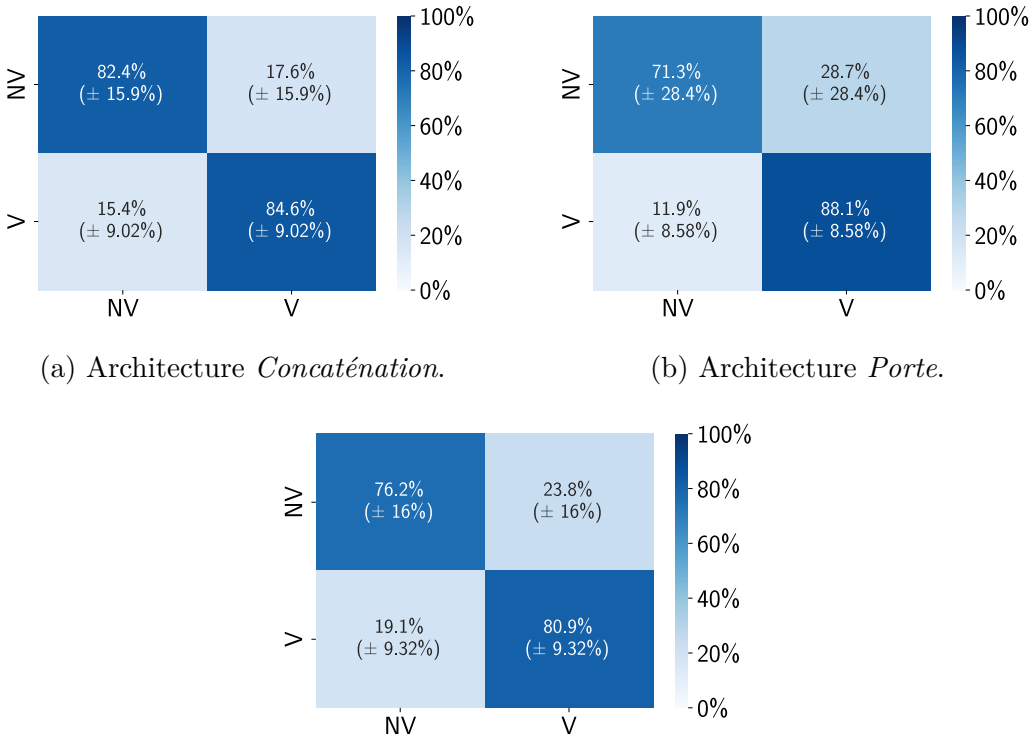
(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

FIGURE 5.5 – Diagrammes de Venn des erreurs de type faux positif FP (a) et de type faux négatif FN (b) des architectures *Audio*, *VidéoCrop* et *Moyenne* sur les 100 répartitions.



(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

FIGURE 5.6 – Répartition par diagrammes de Venn des erreurs FP (a) et FN (b) sur les 100 répartitions des architectures *Audio*, *VidéoCrop* et *Tardive*.



(a) Architecture *Concaténation*.

(b) Architecture *Porte*.

(c) Architecture *Attention*.

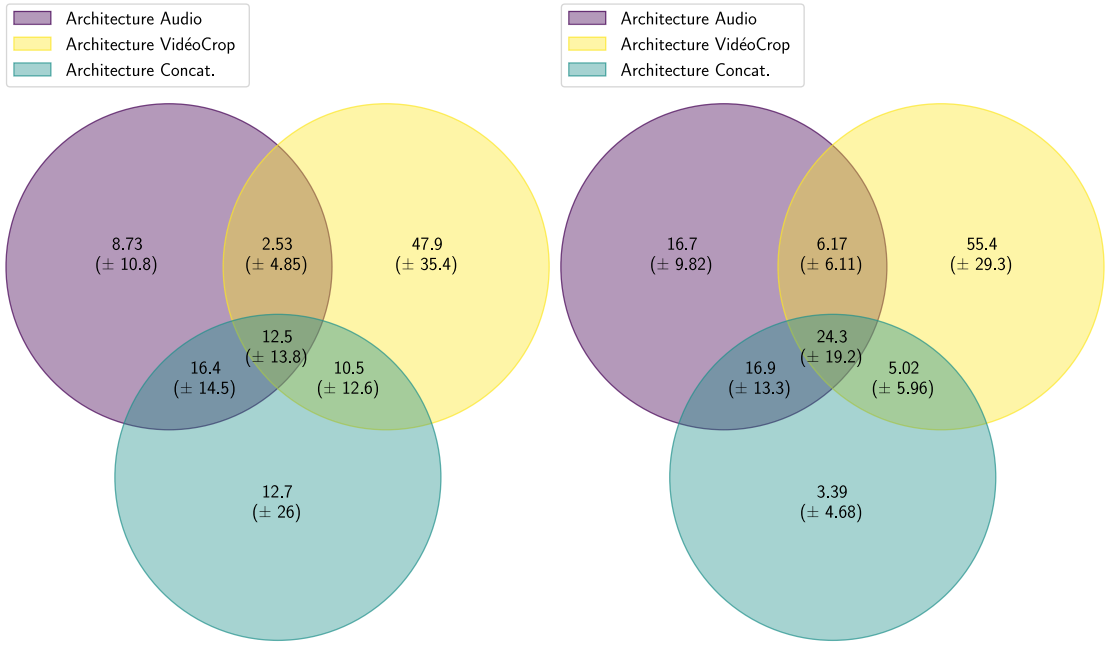
FIGURE 5.7 – Matrice de confusion des architectures multi-modales en fonction de la stratégie de combinaison à un niveau moyen de combinaison. Résultats réalisés sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.

on observe que la combinaison par mécanisme à porte est meilleure ; cette stratégie permet de réduire le nombre de faux négatifs. En analysant les performances plus finement avec les matrices de confusions, nous pouvons remarquer que la combinaison par concaténation permet d'obtenir de meilleures performances sur la reconnaissance des exemples sans violence (Figure 5.7) : l'architecture fait moins d'erreurs de fausses détections (faux positif, FP). D'un autre côté, la combinaison par mécanisme à porte permet d'obtenir de meilleures performances sur la reconnaissance des exemples avec violences car cette architecture fait moins d'erreurs de type faux négatif (FN). La combinaison par attention croisée quant à elle n'apporte aucun gain. Nous concluons que la concaténation est la meilleure stratégie de combinaison à un niveau moyen vu les faibles différences de résultats et des écarts de complexité qui existent entre la combinaison par concaténation et les combinaisons par mécanisme à portes et par attention croisée.

Comme précédemment, nous proposons d'analyser ensuite les erreurs avec des diagrammes de Venn avec comme ensembles les erreurs des architectures *Audio*, *VidéoCrop* et *Concaténation* en figure 5.8, les erreurs des architectures *Audio*, *VidéoCrop* et *Porte* en figure 5.9 et les erreurs des architectures *Audio*, *VidéoCrop* et *Attention* en figure 5.10.

Architectures	Exactitude	Rappel	Précision
Concaténation	83,7% (± 6,67%)	84,6% (± 9,01%)	86,3% (± 8,31%)
Porte	80,4% (± 11,0%)	88,1% (± 8,57%)	81,3% (± 12,1%)
Attention	78,8% (± 6,72%)	80,9% (± 9,32%)	81,3% (± 9,09%)

TABLE 5.8 – Résultats moyens sur les 100 répartitions pour les différentes architectures développées en fonction de la stratégie de combinaison à un niveau moyen.



(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

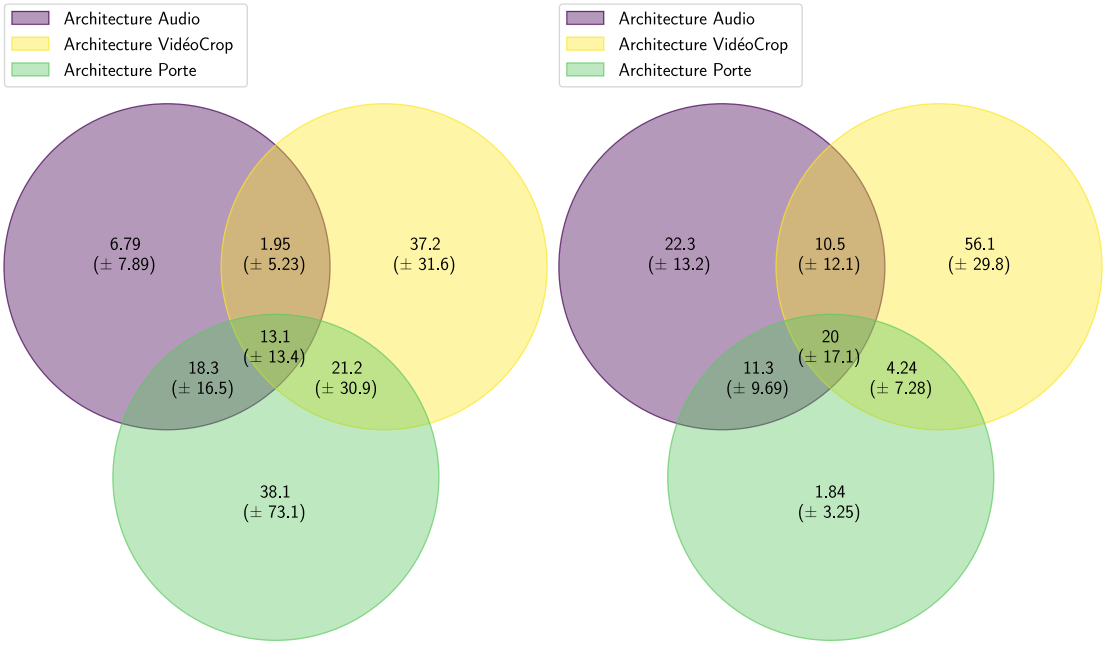
FIGURE 5.8 – Diagrammes de Venn des erreurs de type faux positif FP (a) et faux négatif FN (b) sur 100 répartitions des architectures *Audio*, *VidéoCrop* et *Concaténation*.

Ces diagrammes sont, comme pour l'analyse des niveaux de combinaisons, des extensions des diagrammes de Venn des architectures uni-modales *Audio* et *VidéoCrop* (Figure 5.3). Pour rappel ces combinaisons ont été réalisées à un niveau moyen, le diagramme de Venn des architectures *Audio*, *VidéoCrop* et *Concaténation* (Figure 5.8) est donc le même que le diagramme de Venn des architectures *Audio*, *VidéoCrop* et *Moyenne* (Figure 5.5). Tout d'abord, l'analyse des diagrammes de Venn de l'architecture de combinaison par concaténation (Figure 5.8) montre que cette stratégie permet de réduire le nombre de faux positifs. Ensuite, l'analyse des diagrammes de Venn de l'architecture de combinaison par mécanisme de porte (Figure 5.9) montre que cette stratégie permet de réduire le nombre d'erreurs FN. Enfin, comme avec les matrices de confusion, l'analyse des diagrammes de Venn de l'architecture de combinaison par attention croisée (Figure 5.10) montre que cette stratégie n'apporte aucun gain hormis la réduction des erreurs FP (13,41 et 11,17) et FN (15,69 et 21,83) communes avec les architectures *Audio*.

5.3.4 Stratégie d'apprentissage

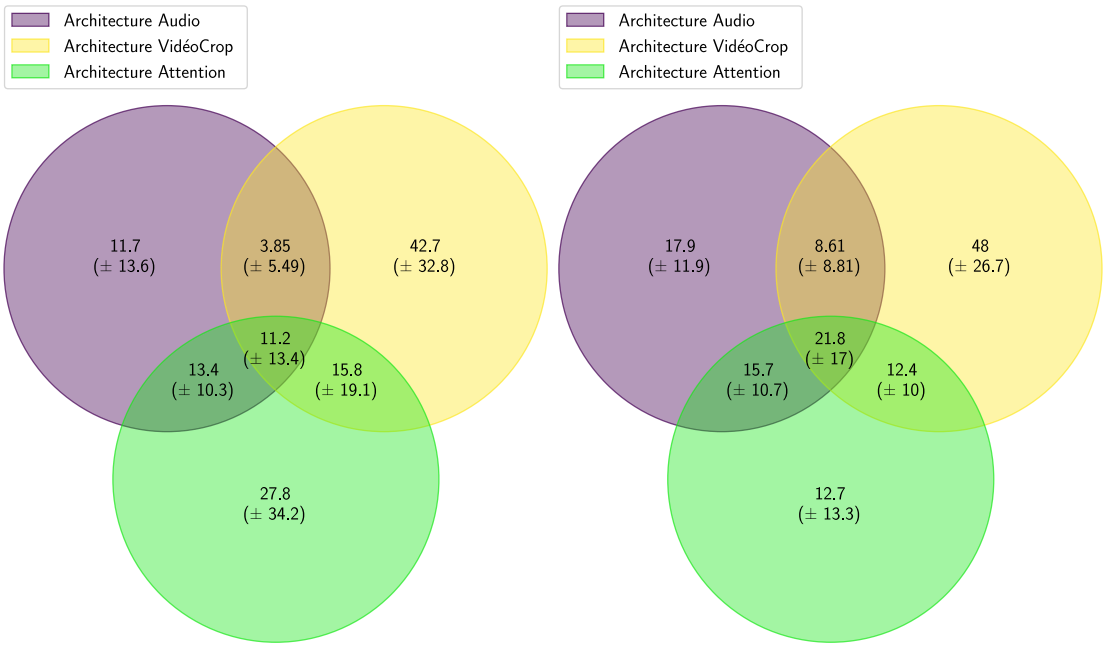
La quatrième et dernière phase d'évaluation quantitative de nos architectures a consisté à évaluer deux stratégies d'apprentissage. Ces stratégies d'apprentissage ont été évaluées avec une combinaison à un niveau moyen par concaténation. Les résultats sont présentés dans le tableau 5.9. Ce tableau présente les scores d'exactitude, de précision et de rappel des architectures évaluant les deux stratégies d'apprentissage proposées. Si l'on considère les scores d'exactitude et de précision, l'apprentissage standard (Architecture *Standard*) semble être légèrement plus performant que l'apprentissage par contrainte (Architecture *Contrainte*) avec des valeurs de 83,7% *vs.* 82,1% pour l'exactitude et 86,3% *vs.* 83,5% pour la précision. En considérant à cela les valeurs de rappel, l'apprentissage avec contrainte serait plus précis que l'apprentissage standard et l'apprentissage standard réaliserait moins d'oublis que l'apprentissage par contrainte.

En analysant les performances plus finement avec les matrices de confusions, nous pouvons remarquer que l'apprentissage standard tend à obtenir de meilleures performances



(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

FIGURE 5.9 – Diagrammes de Venn des erreurs de type faux positif FP (a) et faux négatif FN (b) sur 100 répartitions des architectures *Audio*, *VidéoCrop* et *Porte*.



(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

FIGURE 5.10 – Répartition par diagrammes de Venn des erreurs FP (a) et FN (b) sur les 100 répartitions des architectures *Audio*, *VidéoCrop* et *Attention*.

Architectures	Exactitude	Rappel	Précision
Standard	83,7% (± 6,67%)	84,4% (± 9,01%)	86,3% (± 8,31%)
Contrainte	82,1% (± 9,11%)	86,6% (± 8,70%)	83,5% (± 10,55%)

TABLE 5.9 – Résultats moyens sur 100 répartitions pour les stratégies d'apprentissage "Standard" ou "Contrainte" mises en œuvre pour une architecture à un niveau moyen par concaténation.

sur la reconnaissance des exemples sans violence (Figure 5.11), alors que l'apprentissage contraint tend à augmenter légèrement le taux de reconnaissance des exemples avec violences.

Comme précédemment, l'analyse des résultats par diagramme de Venn consiste à comparer les erreurs en termes de faux positifs et faux négatifs obtenus avec les architectures *Audio*, *VidéoCrop* et l'architecture multi-modale par apprentissage standard (Architecture *Standard*) (Figure 5.12), avec ceux obtenus avec ces mêmes architectures uni-modales et l'architecture multi-modale par apprentissage contraint (Architecture *Contrainte*) (Figure 5.13). Pour rappel, ces stratégies d'apprentissage ont été réalisées à un niveau moyen par concaténation. Ainsi, le diagramme de Venn des architectures *Audio*, *VidéoCrop* et *Standard* (Figure 5.12) est un rappel de ceux vus en figure 5.5) et figure 5.8. D'une manière générale, ces différents diagrammes de Venn nous confirment simplement les observations réalisées avec les matrices de confusion, à savoir :

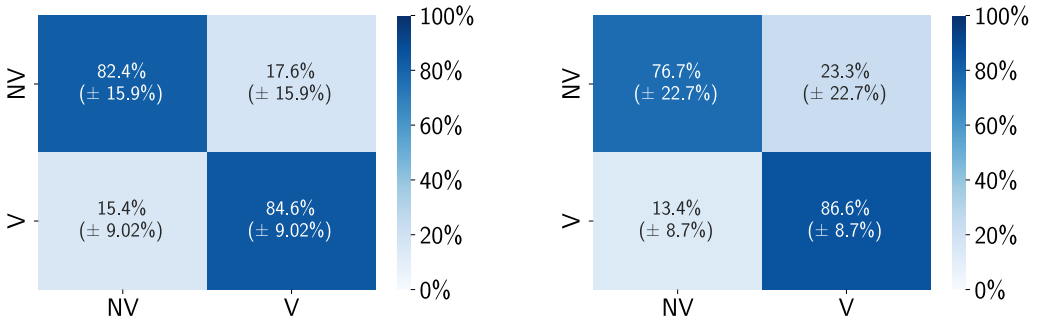
— l'architecture multi-modale apprise de manière standard tend à réduire légèrement le nombre d'erreurs de faux positifs (FP) (Figure 5.12) par rapport à l'architecture multi-modale apprise avec contrainte (Figure 5.13) (52,08 (16,36 + 12,49 + 10,5 + 12,73) *vs.* 70,67 (17,59 + 12,86 + 15,59 + 24,63)).

— l'architecture multi-modale apprise par contrainte (Figure 5.13) montre une légère réduction du nombre de faux négatifs (FN) vis-à-vis de celle apprise classiquement (Figure 5.12) (43,45 (15,17 + 21,56 + 4,07 + 2,65) *vs.* 49,61 (16,93 + 24,27 + 5,02 + 3,39)).

La conclusion de cette évaluation est que l'apprentissage avec contrainte n'apporte pas concrètement l'effet désiré, à savoir d'améliorer les résultats en terme de précision (bonnes détections).

5.4 Analyses descriptives des résultats sur la base de données (*R2N*)

Cette section présente des analyses plus qualitatives de nos résultats, rendues possibles par la procédure d'enregistrement et d'annotation. Nous les avons concentrés sur l'architecture multi-modale ayant obtenu les résultats les plus convaincants à savoir l'architecture combinant les modes audio et vidéo par concaténation à un niveau moyen sans



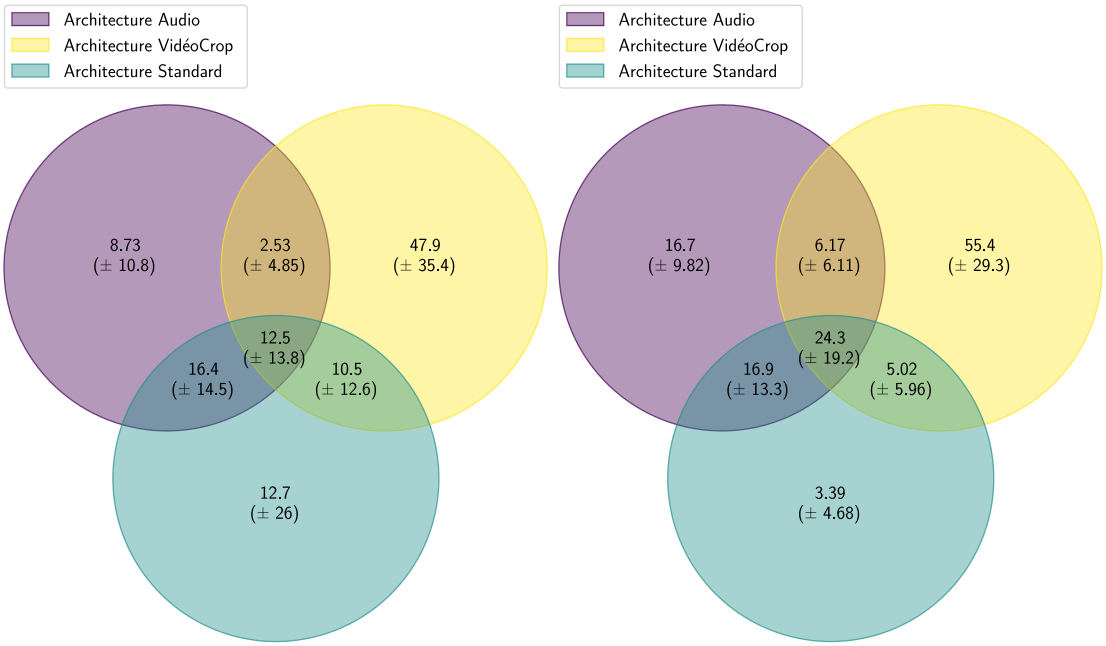
(a) Architecture *Standard*.

(b) Architecture *Contrainte*.

FIGURE 5.11 – Matrice de confusion de l'architecture multi-modale avec une combinaison par concaténation à un niveau moyen apprise avec la stratégie d'apprentissage "Standard" ou "Contrainte". Résultats réalisés sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.

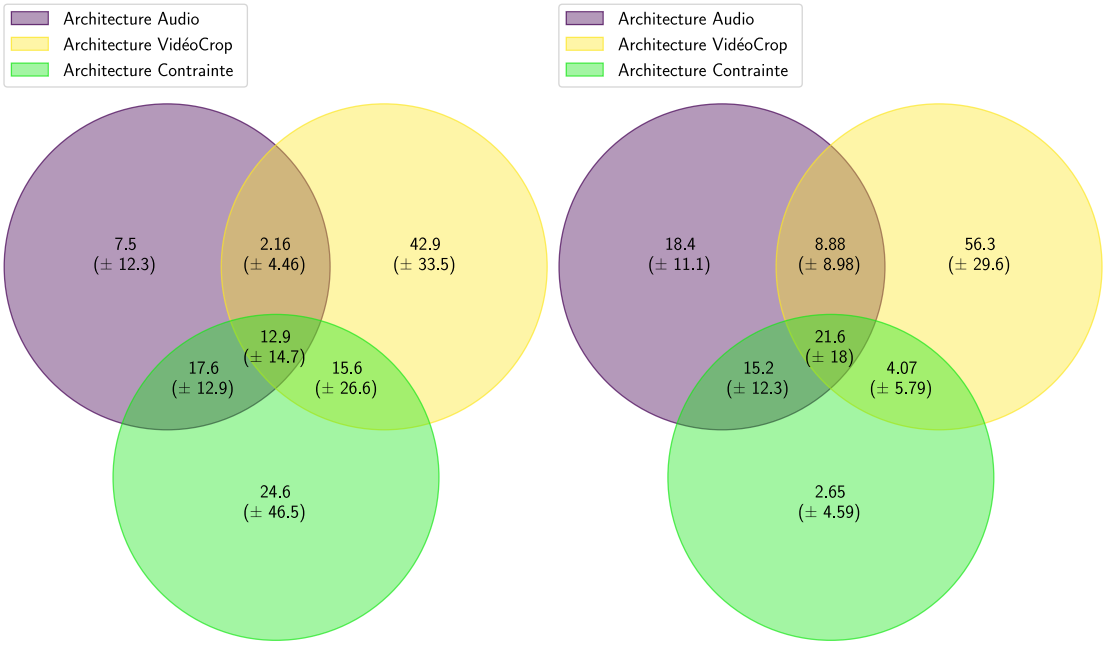
contrainte à l'apprentissage dont nous rappelons ci-dessous la matrice de confusion, les scores d'exactitude, de rappel et de précision.

Ces analyses ont pour objectif de décrire les résultats de reconnaissance de violence en fonction des paramètres suivants : le mode de perception de la violence, la distance au capteur de la violence, le niveau d'occultation, le niveau de violence ainsi que la durée de la violence. Nous présentons ces résultats sous le format utilisé jusqu'à maintenant (Matrice de confusion + scores) en décomposant les observations de violence en fonction des caractéristiques précédemment cités. Les observations de non violence n'étant pas



(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

FIGURE 5.12 – Répartition par diagrammes de Venn des erreurs FP (a) et FN (b) sur les 100 répartitions des architectures *Audio*, *VidéoCrop* et *Standard*.



(a) Faux Positifs (FP). (b) Faux Négatifs (FN).

FIGURE 5.13 – Répartition par diagrammes de Venn des erreurs FP (a) et FN (b) sur les 100 répartitions des architectures *Audio*, *VidéoCrop* et *Contrainte*.

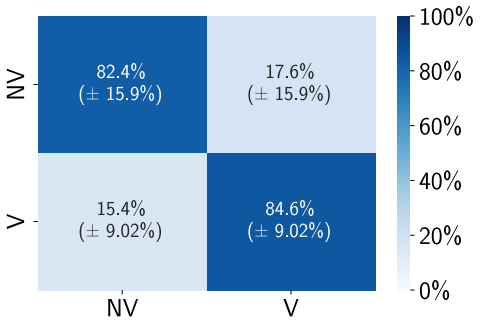


FIGURE 5.14 – Matrice de confusion de l'architecture multi-modales *Moyenne* retenue pour les analyses descriptives des résultats. Résultats réalisés sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.

Architectures	Exactitude	Rappel	Précision
Moyenne	83,7% (± 6,67%)	84,6% (± 9,01%)	86,3% (± 8,31%)

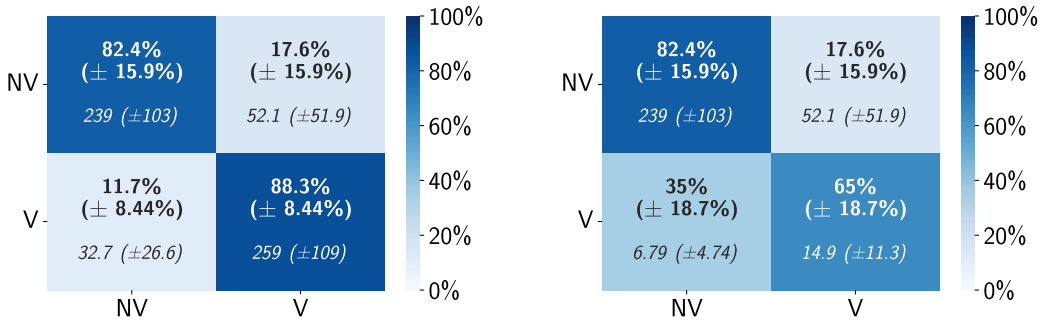
TABLE 5.10 – Résultats moyens sur les 100 répartitions de l'architecture *Moyenne* retenue pour les analyses descriptives des résultats.

atteintes par ces conditions d'analyses impliquent que les lignes des matrices de confusion correspondantes aux résultats de la reconnaissance de segments non violents seront par définition les mêmes pour chacune des matrices. Enfin, les observations de violence et non violence ayant été équilibrées, implique que la décomposition des violences en sous-catégorie d'analyse provoquera un déséquilibre entre ces sous-catégories de violences et l'ensemble des données non-violentes.

5.4.1 Analyse des résultats en fonction du mode de perception de la violence

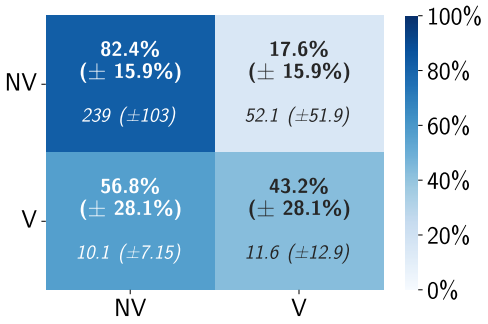
Comme décrit en section 4.9, cette première analyse s'appuie sur l'annotation précisant le mode par laquelle la violence a été perçue : violence perçue sur le signal audio et le signal vidéo (selon l'annotation *Globale* étendue à 5s), violence perçue sur le signal audio indépendamment du signal vidéo (annotation *Audio* étendue à 5s) et violence perçue sur le signal vidéo indépendamment du signal audio (annotation *Vidéo* étendue à 5s) Ainsi, nous avons répertorié les résultats de l'architecture multi-modale en fonction du mode de perception des segments de violence.

Pour commencer, nous considérons les matrices de confusion de la figure 5.15. Chacune de ces trois matrices présentes les résultats en fonction de la modalité de perception. Ces matrices de résultats traduisent que notre architecture a su pleinement modéliser les différentes complémentarités des signaux audio et vidéo : lorsque la violence est perçue simultanément sur le signal audio et le signal vidéo, le taux de reconnaissance est le plus élevé avec un taux de 88,3%. Par contre, notre architecture a eu plus de mal à considérer les "incohérences" de perception. En effet, les scores ne sont pas équivalents sur les trois matrices de confusion : le manque éventuel d'information de violence sur le signal vidéo vis-à-vis du signal audio implique une baisse de -23,3% de bonne de reconnaissance. Ce fait est plus accentué lorsque la perception de la violence n'est considérée que sur le signal vidéo avec une baisse -45,1% de bonne reconnaissance. Selon nous, la raison principale de cette considération "non égale", pour ces trois modes de perception de la part de notre modèle, vient du fait que la présence de violence se perçoit en majorité simultanément sur les deux modes. Nos données sont par nature fortement déséquilibrées en terme de



(a) Perception Audio & Vidéo.

(b) Perception Audio.



(c) Perception Vidéo.

FIGURE 5.15 – Matrice de confusion en fonction de la perception des violences pour l'architecture combinant les signaux par concaténation à un niveau moyen. Résultats, en pourcentage et nombre d'instances, obtenus sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences. Entre parenthèse : variance en pourcentage et en nombre d'instances.

mode de perception : nous avons en moyenne pour les données d'apprentissage 1732 données pour des violences perçues simultanément sur les deux modes (équilibrées avec les données non violentes), et en moyenne 148 données perçues que par le signal audio d'une part et que par le signal vidéo d'autre part, après un équilibrage entre ces deux modes (Tableau 4.2). Le modèle a donc vu lors de son apprentissage principalement des données "cohérentes" entre les deux modes et peu de données "incohérentes. Enfin, les différences de performances analysées au travers des annotations des modes uniques, sont dans la même lignée que les résultats obtenus avec les architectures uni-modales (Figure 5.1b et 5.2) : Il existe à l'origine une plus grande difficulté à discriminer les scènes violentes des scènes non violentes pour le mode vidéo.

5.4.2 Analyse des résultats en fonction de la distance aux capteurs

Cette deuxième analyse s'appuie sur la distance des violences aux capteurs. On rappelle que les violences ont été jouées à trois distances différentes : Zone 1 scène de violence proche des capteurs (de 0 à 3 mètre), Zone 2 scène de violence à moyenne distance (de 3 à 6 mètre) et Zone 3 scène de violence loin des capteurs (de 6 à 9 mètre). Les scores d'exactitude, de précision et de rappel (Tableau 5.11) et les matrices de confusion (Figure 5.16) présentent les résultats en fonction des zones où ont été jouées les scènes de violence.

À la lecture du tableau 5.11, nous constatons que, d'un point de vue général, l'architecture multi-modales a su modéliser la violence d'une manière assez homogène selon

les différentes zones. Avec un regard plus précis, la distance a un impact léger entre la détection des violences en zone 1 et 2 avec une légère baisse pour la zone 2 sur tous les scores (avec un maximum de baisse pour la précision égale à -2,5% en moyenne sur 100 répartitions). Cette influence est un peu plus accentuée en considérant la zone 3 avec un maximum de baisse de -11,2% pour le rappel. Ces résultats se retrouvent à travers les matrices de confusion définies pour chaque zone (Figure 5.16) : en Zone 1 et 2 le taux de bonne détection de la violence est égale ou au-delà de 87,4% alors que pour la Zone 3 celui-ci est de 76,7%.

Nous avons déjà observé que les résultats de détection pour la part vidéo sont fonction de la distance. Pour rappel, un modèle à 3 branches purement vidéo permettait d'augmenter les résultats de détection de violence à 71,8% quand ceux-ci étaient de 67,0% avec une seule branche vidéo (Figure 5.1a). Avec cette architecture, la multi-modalité a permis de porter ce score à 84,6% (Figure 5.4b). Avec ces résultats, la multi-modalité de notre architecture montre une certaine efficacité si on considère les valeurs de bonne détection en Zone 1 et en Zone 2. On constate que la Zone 3 est quant à elle plus délicate. Cette zone qui est plus éloignée est associée indiscutablement à une perception sonore et visuelle affaiblies.

Enfin, nous noterons que les scores de précision sont plus faibles pour la reconnaissance de violences en fonction des zones, que celui présente sans critère de zone (Tableau 5.10) : ceci est dû au déséquilibre provoqué par la décomposition des données violences (équilibrées avec les non-violences) en 3 catégories de violence fonctions des zones. Cette remarque est également valable pour la considération des valeurs d'exactitudes même si l'impact (normalement à la hausse) n'est pas aussi marqué.

5.4.3 Analyse des résultats en fonction des degrés d'occultation

Pour cette nouvelle analyse, nous considérons le degré d'occultation, avec pour rappel : occultation de degré 1 pour les scènes de violence avec un faible degré d'occultation, de degré 2 pour les scènes de violence avec un degré d'occultation moyen et occultation de degré 3 pour les scènes de violence avec un degré d'occultation important.

Les résultats sont présentés dans le tableau 5.12 : Le score de rappel le plus faible est celui où l'occultation est la plus forte en moyenne (sur 5s) avec une valeur de 81,5%. Cependant, cette valeur n'est pas "dramatique" si on considère le score obtenu en moyenne par une faible occultation (de degrés 1) égale à 85,4%, lui-même paradoxalement plus faible que celui obtenu par des violences ayant subi une occultation de degrés 2. Le modèle semble en général peu impacté par le problème d'occultation. De plus, ces performances ne sont pas corrélées avec ceux obtenus avec l'éloignement de la scène de violence (les scores de rappel décroissent en fonction de l'éloignement, *cf.* section 5.4.2), alors que l'éloignement d'une scène peut être un contexte favorable à l'occultation. L'explication vient du fait que l'éloignement en zone peut avoir un impact sur la vidéo mais également sur l'audio avec une perception sonore amoindrie (niveaux sonores vis-à-vis du bruit ambiant, de la diffusion et d'une source directe plus affaiblie, etc.). Par définition, la considération de

	Exactitude	Rappel	Précision
Zone1	84,2% (± 10,4%)	87,9% (± 7,34%)	71,6% (± 14,1%)
Zone2	83,8% (± 10,9%)	87,4% (± 8,81%)	69,1% (± 15,1%)
Zone3	81,1% (± 10,6%)	76,7% (± 13,7%)	65,2% (± 16,0%)

TABLE 5.11 – Résultats moyens sur 100 répartitions en fonction de la distance des violences aux capteurs pour l'architecture multi-modale combinant les signaux par concaténation à un niveau moyen.

	Exactitude	Rappel	Précision
Occultation degré 1	83,3% (± 12,8%)	85,4% (± 12,4%)	54,0% (± 17,6%)
Occultation degré 2	84,4% (± 9,64%)	87,6% (± 9,70%)	76,3% (± 13,6%)
Occultation degré 3	81,8% (± 11,6%)	81,5% (± 12,1%)	61,1% (± 19,1%)

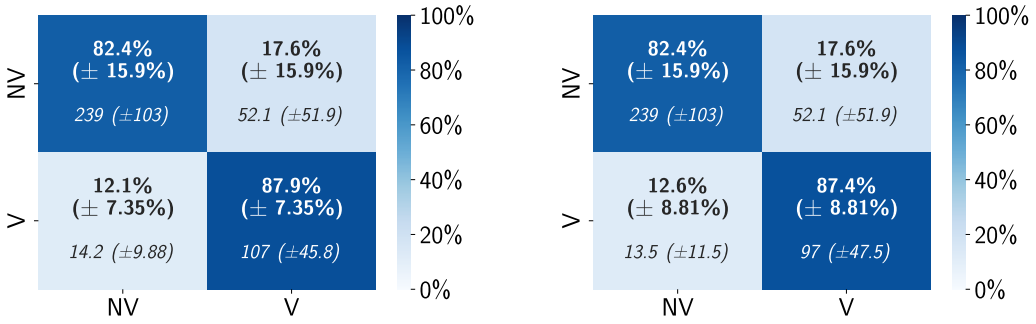
TABLE 5.12 – Résultats moyens sur 100 répartitions en fonction du degré d'occultation des violences pour l'architecture combinant les signaux par concaténation à un niveau moyen.

la seule occultation n'affecte que le signal vidéo et pas la perception sonore. On constate donc que l'information sonore permet au modèle audio-visuel d'être robuste au problème spécifique de l'occultation.

Enfin comme précédemment les valeurs de précision et d'exactitude souffrent du déséquilibre entre les classes "Non Violence" et les sous-classes "Violence" issues de la décomposition en 3 sous-catégories distinctes d'occultation.

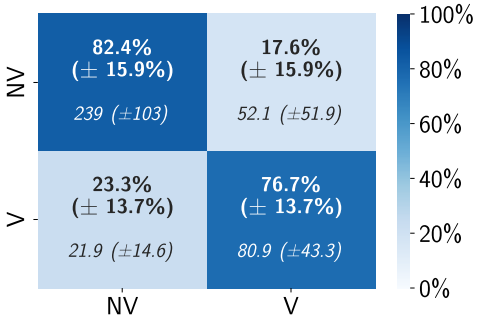
5.4.4 Analyse des résultats en fonction des degrés de violence

Cette analyse s'appuie sur le degré de violence perçu à travers le mode vidéo (sans considérer le signal audio). Pour rappel : la violence de degré 1 équivaut à des scènes de violence avec un faible degré de mouvement, la violence de degré 2 équivaut à des scènes de violence avec un degré de mouvement moyen et la violence de degré 3 équivaut à des



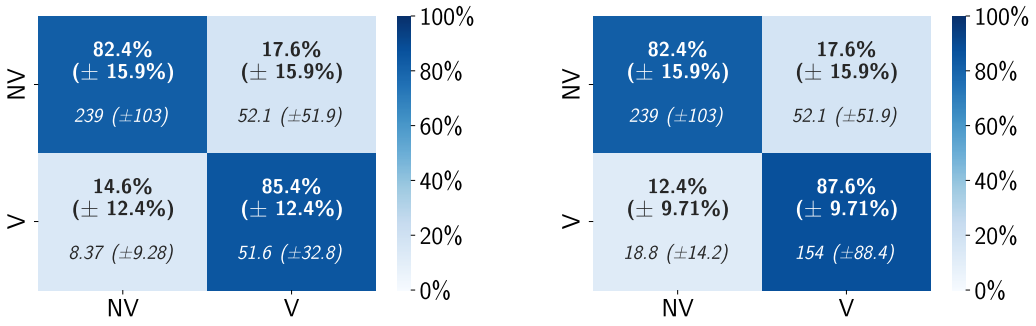
(a) Zone 1.

(b) Zone 2.



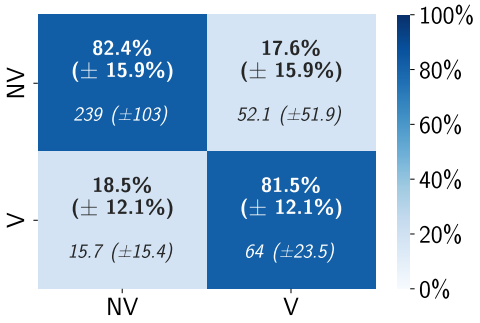
(c) Zone 3.

FIGURE 5.16 – Matrice de confusion pour l'architecture multi-modale combinant les signaux par concaténation à un niveau moyen en fonction de la distance des violences aux capteurs. Résultats, en pourcentage et nombre d'instances, réalisés sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences. Entre parenthèse : variance en pourcentage et en nombre d'instances.



(a) Occultation niveau 1.

(b) Occultation niveau 2.



(c) Occultation niveau 3.

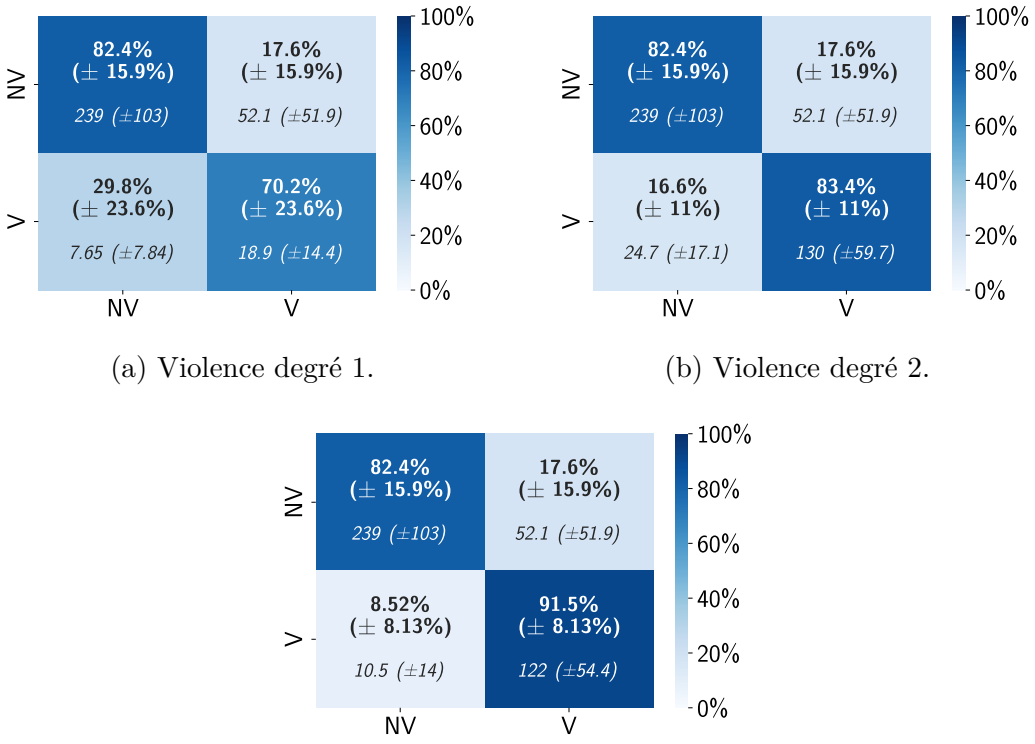
FIGURE 5.17 – Matrice de confusion en fonction du degré d'occultation pour une architecture combinant les signaux par concaténation à un niveau moyen. Résultats, en pourcentage et nombre d'instances, sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences. Entre parenthèse : variance en pourcentage et en nombre d'instances.

scènes de violence avec un degré de mouvement élevé. Les divers scores sont présentés dans le tableau 5.13 et les matrices de confusion pour chacun des degrés de violences en figure 5.18).

En se focalisant sur les scores de rappel, on observe que ces derniers croissent en fonction des degrés de violence perçus par le mode vidéo. Plus précisément, le degré de violence le plus élevé atteint un score de rappel (taux de bonne reconnaissance) de 91,5%. Le rappel du degré 2 est équivalent à celui du cadre général (84.6%, tableau 5.10). Par contre, le degré 1 présente un taux de reconnaissance bien plus faible avec en particulier un taux de rappel de 70,2%. Ainsi, bien que cette annotation du niveau de violence ait été établie uniquement sur le mode vidéo, le modèle audio-visuel dans son ensemble est affecté par ces différents niveaux. La première hypothèse expliquant ces résultats est qu'une violence faiblement perçue peut être facilement confondue avec des scènes non violentes, comme des déplacements rapides de passagers (indépendamment de l'éloignement aux capteurs dans ce cas). La deuxième hypothèse est la cohérence qu'il

	Exactitude	Rappel	Précision
Violence degré 1	81,6% (± 14,2%)	70,2% (± 23,6%)	29,4% (± 17,3%)
Violence degré 2	82,9% (± 9,39%)	83,4% (± 11,0%)	74,8% (± 12,6%)
Violence degré 3	85,2% (± 10,4%)	91,5% (± 8,12%)	73,3% (± 14,6%)

TABLE 5.13 – Résultats moyens sur 100 répartitions en fonction du degré de violence pour l'architecture combinant les signaux par concaténation à un niveau moyen.



(a) Violence degré 1.

(b) Violence degré 2.

(c) Violence degré 3.

FIGURE 5.18 – Matrice de confusion en fonction du degré de violence pour l'architecture combinant les signaux par concaténation à un niveau moyen. Résultats, en pourcentage et en nombre d'instances, sur 100 répartitions, avec "NV" pour les exemples sans violences et "V" les exemples avec violences. Entre parenthèse : variance en pourcentage et en nombre d'instances.

peut exister entre la perception sonore et la perception visuelle : une scène perçue peu violente visuellement peut l'être également par le son. De plus, notre modèle semble souffrir également d'un manque de données à l'apprentissage pour ce degré 1 de violence (en moyenne 159 observations pour l'apprentissage contre 917 pour le degré 2 et 799 pour le degré 3, tableau 5.1). Cet aspect se manifeste à travers les écart-types des valeurs de rappel obtenues sur les 100 répartitions : l'écart type du degré 1 et quasi le double du degré 2 et presque le triple du degré 3, signifiant ainsi une certaine inefficacité de notre modèle pour ce degré de violence. Enfin, notons, comme nous l'avions déjà exprimé en section 3.1.6, que la frontière entre la non-violence et la violence du plus faible degré peut être parfois difficile à objectiver, ce qui peut être également une conséquence de ce résultat plus faible.

Pour finir, les valeurs de précision et d'exactitude souffrent du déséquilibre entre la classe "Non Violence" et ces 3 nouvelles classes violences. Néanmoins, et relativement aux valeurs de précision des degrés 2 et 3, la faible valeur du degré 1 tend à souligner la difficulté de notre modèle à différencier une faible violence d'une "Non Violence" (sur peu de données).

5.4.5 Analyse des résultats en fonction de la durée de perception des violences

Cette sous-section est dédiée à une analyse des résultats en considérant la durée de perception des violences définie dans la section 4.9, avec pour rappel : durée de perception

	Score de rappel		
	Audio	*Vidéo*	*Globale*
Durée de perception 1	51,7% (± 17,8%)	74,3% (± 13,6%)	41,4% (± 20,4%)
Durée de perception 2	81,0% (± 11,7%)	87,2% (± 10,0%)	73,6% (± 13,7%)
Durée de perception 3	93,3% (± 5,18%)	94,9% (± 6,94%)	91,7% (± 6,67%)

TABLE 5.14 – Résultats moyens du rappel sur 100 tirages en fonction des niveaux de durée des violences définie par les annotations *Audio*, *Vidéo* et *Globale* résultant pour l'architecture combinant les signaux par concaténation à un niveau moyen.

	Score de précision		
	Audio	*Vidéo*	*Globale*
Durée de perception 1	28,5% (± 15,7%)	65,6% (± 15,3%)	24,5% (± 14,8%)
Durée de perception 2	54,6% (± 15,9%)	62,5% (± 15,6%)	47,7% (± 15,5%)
Durée de perception 3	82,4% (± 10,6%)	73,2% (± 14,1%)	83,9% (± 9,77%)

TABLE 5.15 – Résultats moyens de la précision sur 100 tirages en fonction des niveaux de durée des violences définie par les annotations *Audio*, *Vidéo* et *Globale* résultant pour l'architecture combinant les signaux par concaténation à un niveau moyen.

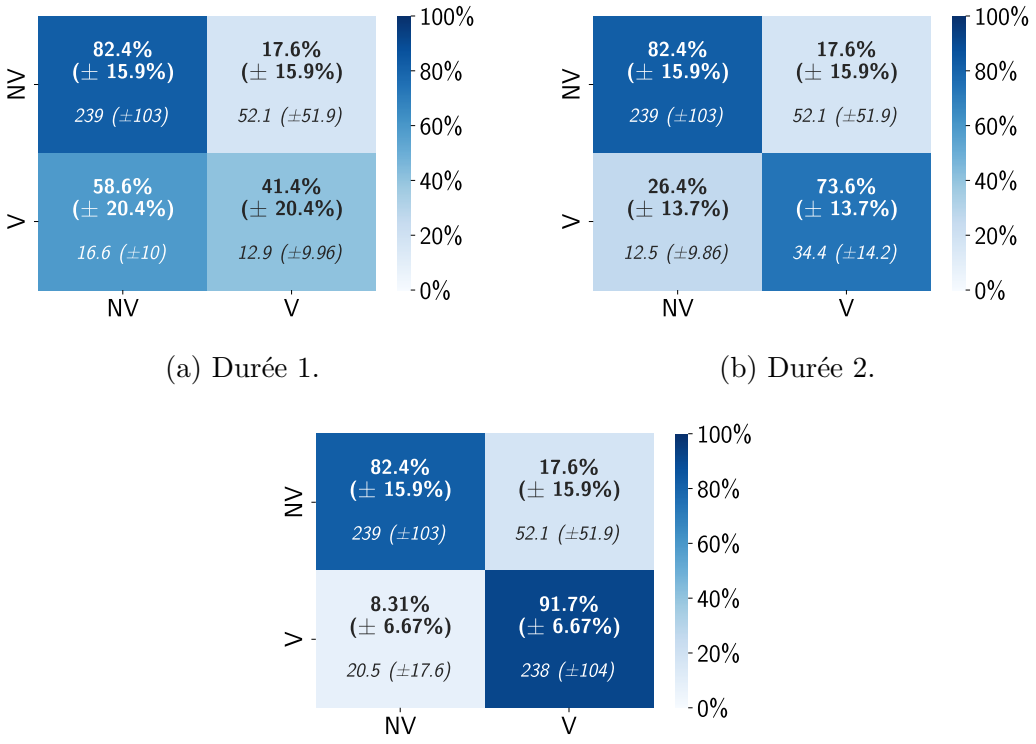
1 pour les scènes de violence avec une durée moyenne comprise entre 0 et 1,66 seconde, Durée de perception 2 pour les scènes de violence avec une durée moyenne comprise entre 1,66 et 3,33 secondes et Durée de perception 3 pour les scènes de violence avec une durée moyenne comprise entre 3,33 et 5,00 secondes.

Les résultats sont présentés dans les tableaux 5.14 et 5.15 montrent respectivement les scores de précision et de rappel en fonction des niveaux de durée des violences indexées par l'annotation *Audio*, par *Vidéo*, et par l'annotation *Globale* (Section 4.1.1) résultante. Le score d'exactitude n'a pas été retenu pour réaliser l'analyse des performances en fonction des niveaux de durée car il n'est que peu pertinent avec la distribution des données déséquilibrées entre ces niveaux de durées (Tableau 5.2). Globalement, au travers de ces tableaux, on peut observer que la durée de la violence dans le segment affecte significativement les performances, c'est-à-dire les performances sont plus faibles lorsque la durée est faible alors que les performances sont plus élevées lorsque la durée est plus longue. Cette conclusion semble particulièrement vraie pour la durée de perception *Audio* et *Globale*. La durée de perception sur le signal *Vidéo* a quant à elle un impact plus limité sur les performances. Ces résultats seront approfondis avec les matrices de confusions dans les sous-sections suivantes.

5.4.5.1 Durée de perception événement (annotation globale)

La figure 5.19 présente donc les matrices de confusions en fonction des durées de violence (niveaux 1, 2 et 3) définies par l'annotation *Globale*.

Comme cela était prévisible, plus la durée de la violence perçue est longue, plus les scores sont en hausses. Avec une durée de niveau 3 le score moyen de rappel est de 91,7%, baisse à 73.6% pour une durée de niveau 2 et à 41.4% pour le dernier niveau. Ce dernier résultat montre qu'une très courte durée de violence dans un segment de 5 secondes ne semble pas être correctement modélisée (-considérée-) par notre modèle. L'une des explications possibles à cela est la rareté de ces courtes durées de violence quand celles-ci sont définies quel que soit le signal : dans notre répartition présentée dans le tableau 5.2, ce paradigme représente moins de 10% de nos données de violence tant sur les données d'apprentissage que sur les données de test. Ce manque de données se reflète une nouvelle fois à travers l'efficacité de notre modèle avec un écart-type du score de rappel de 20,4%



(a) Durée 1.

(b) Durée 2.

(c) Durée 3.

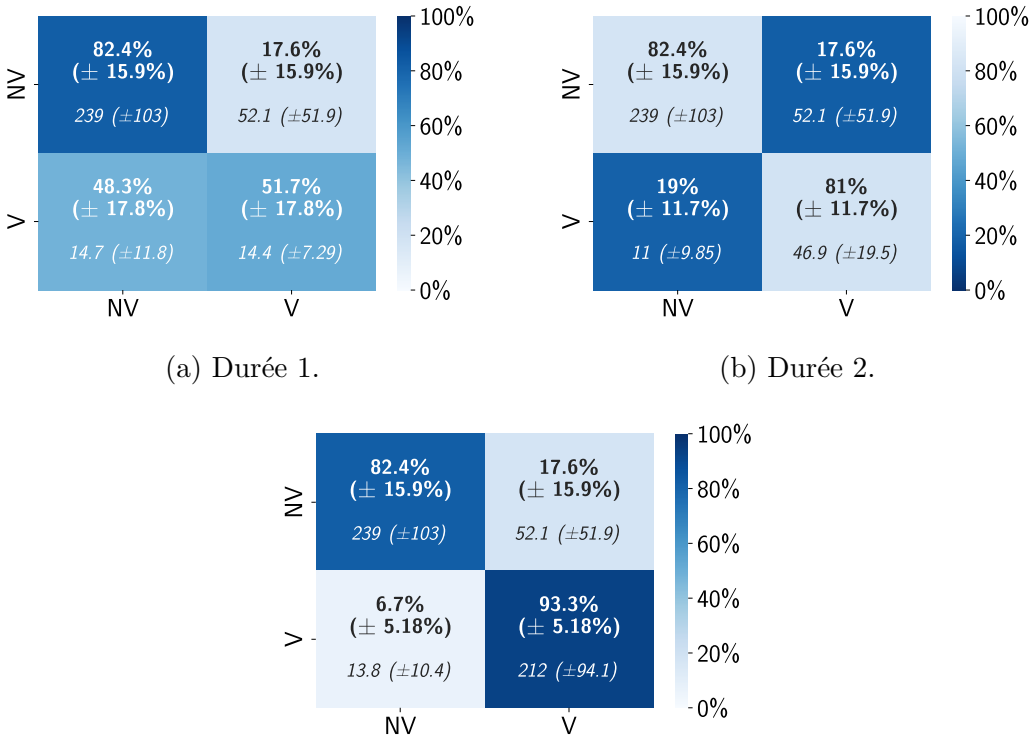
FIGURE 5.19 – Matrice de confusion par niveau de durée des violences selon l'annotation *Globale* pour l'architecture combinant les signaux par concaténation à un niveau moyen. Résultats, en pourcentage et nombre d'instances réel, sur 100 répartitions, avec "NV" pour les exemples sans violences et "V" les exemples avec violences. Entre parenthèse : variance en pourcentage et en nombre d'instances.

pour les durées les plus courtes (et les moins présentées), et de 6,67 % pour les durées de niveau 3 les plus longues (et également les plus représentées). En se rappelant que la segmentation de notre annotation n'est pas synchronisée précisément avec le début et la fin des scènes de violence, ces segments de 5s semblent simplement représenter quelques segments composés d'une courte durée de violence au début ou de fin d'une scène violence. L'impacte sur la détection de la scène de violence est donc minime puisque le segment de 5s précédent ou suivant (contenant une durée de violence plus grande) suivra avec des taux de rappels de durées plus élevées (de 91,7% à 73.6%). Ces résultats sont même au contraire intéressants dans le sens où les violences sont composées majoritairement (75%) d'une durée supérieure à 3,33s (niveau 3).

Nous présentons maintenant avec la figure 5.20 et la figure 5.21 les matrices de confusions équivalentes mais en considérant la durée des violences en fonction des annotations audio et vidéo qui ont servi à construire notre annotation Globale.

5.4.5.2 Durée de perception audio

Les matrices dédiées aux durées des annotations audio suivent les mêmes résultats que précédemment, avec un taux de bonne détection de 93,3% pour un niveau de durée 3, de 81,0% pour un niveau 2 et 51,7% pour un niveau 1. Cette équivalence entre les résultats se retrouve également dans la répartition des durées des signaux audio, à savoir que les durées de niveau 1 ne représentent que 11.0% des violences (60% pour le niveau 3). Ces équivalences entre répartitions et résultats tendent à aboutir aux mêmes conclusions que précédemment et par conséquences, indiquent que le signal audio semble être le mode



(a) Durée 1.

(b) Durée 2.

(c) Durée 3.

FIGURE 5.20 – Matrice de confusion par niveau de durée des violences en considérant l' annotation *Audio* pour l'architecture combinant les signaux par concaténation à un niveau moyen. Résultats, en pourcentage et valeur réelle, sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.

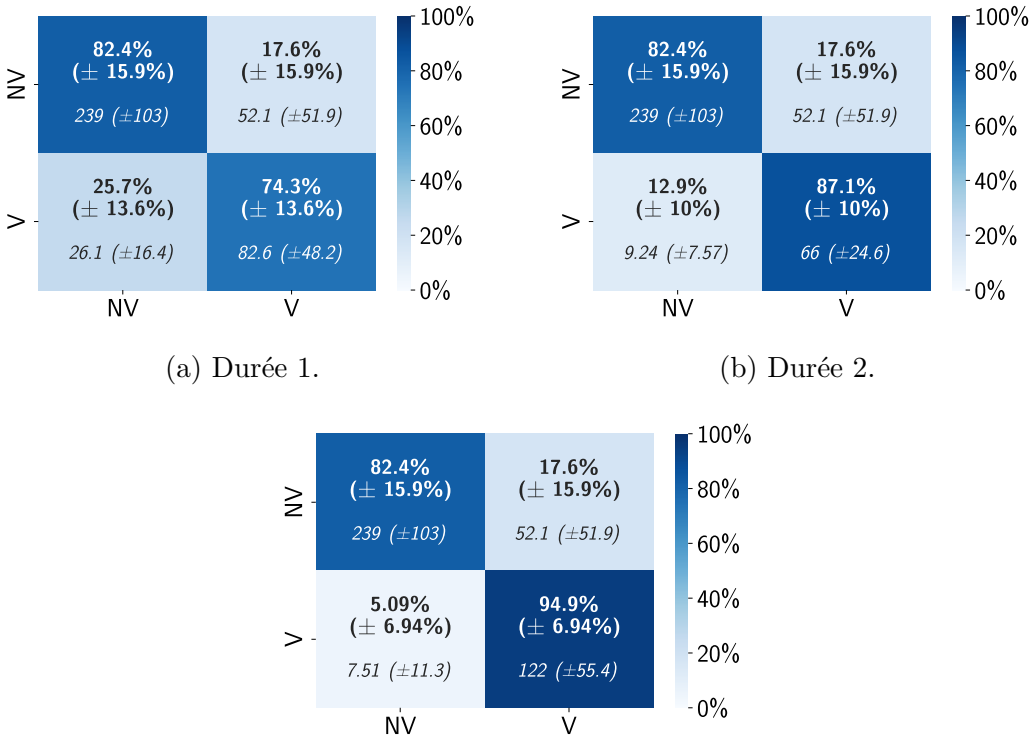
définissant le début et la fin des scènes de violence.

5.4.5.3 Durée de perception vidéo

En comparaison avec l'annotation audio, la figure 5.21 présente les matrices de confusions en fonction du niveau de durée de perception des violences sur le signal vidéo. Les résultats sont une nouvelle fois corrélés avec le niveau de durée des signaux avec des scores de rappel de 94,9% pour le niveau de durée 3, 87,1% pour le niveau 2 et 74,3% pour le niveau 1. Par contre, cette fois-ci, les violences de plus petites durées présentent un score de rappel bien plus élevé en comparaison au cas générale ou au cas audio. Ceci peut s'expliquer au regard de la répartition des niveaux de durée (Tableau 5.2) qui est également plus équilibrée avec une répartition de l'ordre de 35.5%, 23,8% et 41,1% pour les niveaux de durée 1, 2 et 3. Cette fois-ci, avec un nombre de données équivalent au niveau 3, ces matrices de confusion nous montrent que les segments composés de petites durées de violence (sur la vidéo) sont plus difficiles à être appréhendé par notre modèle en comparaison aux durées de perception de niveau 2 et 3.

Conclusions

Au travers de ce chapitre de résultats, nous avons observé quelques statistiques sur les 100 tirages de nos différents ensembles, montrant que nos données sont constituées de violences assez souvent occultées, avec un degré de violence ("visuel") assez élevé, et réalisées équitablement dans trois zones de l'espace observé. Ensuite, nous avons appris



(a) Durée 1.

(b) Durée 2.

(c) Durée 3.

FIGURE 5.21 – Matrice de confusion par niveau de durée des violences en considérant l'annotation *Vidéo* pour l'architecture combinant les signaux par concaténation à un niveau moyen. Résultats, en pourcentage et valeur réelle, sur 100 répartitions avec "NV" pour les exemples sans violences et "V" les exemples avec violences.

et testé notre architecture uni-modale vidéo sur trois jeux de données de référence dédiés la reconnaissance de la violence dans la communauté, ainsi que sur le jeu de données que nous avons construit. Dans la continuité, nous avons évalué trois architectures de la communauté sur notre jeu de données. Au vu de ces premiers résultats, nous avons pu observer que notre jeu de données semble comporter des difficultés supplémentaires mettant à mal les performances des autres travaux. Ensuite, nous avons effectué une évaluation quantitative en utilisant les scores d'exactitude, de rappel, de précision, ainsi que les matrices de confusion et les diagrammes de Venn des architectures que nous avons proposées et apprises à partir de nos données. Il en ressort principalement qu'un modèle "Audio-visuel", quelle que soit sa mise en œuvre, est plus performant qu'une fusion simple des décisions issue de deux modèles audio et vidéo indépendants. Enfin, nous avons détaillé l'analyse qualitative du meilleur modèle en utilisant les différentes méta-annotations disponibles sur notre jeu de données. Cette dernière étude nous a permis de montrer que la distance aux capteurs à un impact sur les performances, avec une décroissance du score de reconnaissance en fonction de l'éloignement. Nous avons constaté que le degré de violence et sa durée dans une observation de 5 secondes, peuvent influer sur les performances. Toutefois, les scores les plus faibles ne sont pas uniquement dus à des segments de 5 secondes comportant de "faibles" violences ou des violences de courtes durées, peu représentatives d'une scène de violence dans son ensemble. Enfin, nous avons montré que notre modèle est invariant au problème de l'occultation, montrant sur ce point toute l'importance du traitement combiné des signaux audio et vidéo.

Conclusions et Perspectives

Ces travaux sur la reconnaissance de violence dans un environnement ferroviaire ont été menés dans le but d'étudier l'utilisation d'un système de surveillance automatique basé sur l'IA pour améliorer la sûreté dans les transports. L'objectif est de ne plus se limiter à la collecte de preuves lors du dépôt de plainte, mais plutôt de reconnaître les incidents violents dès qu'ils se produisent. Cependant, cet environnement présente des spécificités complexes qui rendent difficile l'utilisation des solutions proposées pour d'autres environnements. Nos travaux ont donc consisté à traiter ce problème de reconnaissance de violence dans un cadre le plus opérationnel et réaliste possible dans cet environnement spécifique qu'est une enceinte de véhicule de transport en commun ferroviaire. Pour réaliser cette tâche, nous avons proposé de traiter conjointement des signaux audio et vidéo pour pallier les contraintes liées à l'environnement ferroviaire et permettre de détecter le plus tôt possible les événements violents.

Tout d'abord, le chapitre 1 nous a permis d'introduire les concepts généraux et les fondements relatifs au *Machine Learning*, expression la plus utilisée pour parler de la modélisation statistique par apprentissage. Nous y avons dédié une section propre aux traitements conjoints de données multimodales (Section 1.4, *Les combinaisons multimodales*). Ce socle de connaissances théoriques présenté, le chapitre 2 a eu pour but d'introduire l'état de l'art propre à notre étude. Nous avons introduit ce chapitre par une revue des jeux de données dédiée à la reconnaissance d'actions, d'événements, de violences, ainsi qu'à celle de la combinaison de signaux. Ensuite, nous avons présenté et mis en évidence diverses approches d'apprentissage profond pour (1) la reconnaissance d'actions en considérant le signal vidéo et (2) la reconnaissance d'événements en considérant le signal audio. Une attention particulière a été accordée à la combinaison des signaux audio et vidéo traitée dans la communauté pour la reconnaissance d'événements. Enfin, nous avons terminé ce chapitre par la présentation de travaux spécifiques à la reconnaissance de violences, notamment ceux appliqués dans le contexte ferroviaire. Ces premiers travaux, axés sur la reconnaissance de violence dans un environnement ferroviaire, nous ont permis de constater, qu'au démarrage de notre étude, l'état de l'art ne considérait pas dans notre contexte précis (1) les dernières approches d'apprentissage profond et (2) les combinaisons des signaux audio et vidéo. Enfin, cet état de l'art nous a permis de constater qu'il n'existait malheureusement pas dans la communauté, de jeu de données répondant aux contraintes spécifiques liées à un environnement ferroviaire.

Le chapitre 3 a eu pour but de présenter le cœur de nos travaux. Ne disposant pas de données appropriées à notre contexte étude, nous avons donc choisi d'acquérir notre propre jeu de données. Le protocole que nous avons mis en place pour construire notre jeu de données a été détaillé en première partie de ce chapitre en section 3.1. Cette première phase de travaux nous a permis de disposer de données contrôlées, de travailler avec les contraintes de l'environnement ferroviaire, et d'obtenir des données de qualité incluant à la fois les signaux audio et vidéo synchrones. Suite à l'acquisition, nous avons défini une stratégie d'apprentissage consistant à annoter les observations suivant un découpage des données en segments consécutifs de deux secondes. Cette annotation a été réalisée indépendamment sur les signaux audio et les signaux vidéo. L'annotation était

constituée de 2 classes : "Non Violence" et "Violence". Une procédure de combinaison des deux annotations indépendantes a été établie pour fournir une annotation conjointe nommée *Globale*. Enfin, toute une série de "méta-annotations" ont été présentées dans l'objectif de mieux analyser les résultats dans notre environnement spécifique. En seconde partie (Section 3.2), nous avons présenté les diverses architectures proposées. Ces dernières utilisaient toutes des extracteurs de caractéristiques pré-établis reconnus dans la communauté (*I3D-RGB* pour la branche de traitement vidéo et *OpenL3* pour la branche de traitement audio). Nous avons commencé par proposer des architectures uni-modales (audio ou vidéo) afin d'évaluer la pertinence de chaque signal pour notre tâche. Ensuite, nous avons proposé des architectures multi-modales (audio et vidéo) pour évaluer la pertinence d'une combinaison des signaux dans un modèle de reconnaissance de violences. Pour ces architectures multi-modales, nous avons présenté différents niveaux de combinaison (moyenne ou tardive), différents types de combinaison (concaténation, mécanisme à porte et attention croisée).

Notre base de données établie et nos architectures définies, nous avons décrit en chapitre 4.1 la démarche de mis en œuvre pour l'estimation et l'évaluation de nos travaux. Cela a consisté à définir une durée d'observation des signaux, fixée à 5 secondes. L'annotation sur 2 secondes a donc permis de définir les références "Violence" ou "Non Violence" sur 5 secondes en faisant le choix que toutes présences de violence de 2 secondes définissait comme "Violent" un segment de 5 secondes. Nous avons ensuite présenté les différents paramètres de nos architectures, les procédures d'apprentissage et d'évaluation. Nous avons entre autres proposé, en plus de l'apprentissage "standard", un apprentissage prenant en compte la représentation de la violence à travers sa perception multi-modale.

Enfin, le Chapitre 5 a été dédié à la présentation des résultats obtenus. Les premiers constats concernent le contenu de notre base de données. Premièrement, nous pouvons conclure que l'utilisation d'observations sur 5 secondes et des annotations associées reste cohérente au regard des annotations réalisées sur 2 secondes. En effet, nous avons constaté qu'à travers 100 répartitions sur les ensembles d'entraînement, de validation et de test, les répartitions des données "Violence" et "Non Violence" correspondaient en moyenne à la répartition des données sur 2 secondes. Ensuite, les diverses analyses nous ont montré/confirmé qu'une "violence" pouvait être perçue différemment en fonction des modes : premièrement, la considération de l'occultation du champ de vue vidéo est majeure quand celle-ci apparaît sur une grande partie des données et qu'elle est par définition inexistante sur le mode sonore. La deuxième différence majeure est celle de la durée : une violence perçue en "audio-visuel" semble être en moyenne perçue sur l'ensemble des données sonores, présentant ainsi peu d'observations de 5 secondes comportant une courte durée de violence. A contrario, la vidéo semble plus parcimonieuse, avec des durées moyennes de violences perçues qui sont équilibrées sur 5 secondes.

Nos conclusions portent ensuite sur l'étude préliminaire réalisée sur l'état de l'art. Les résultats ont principalement montré que les données utilisées lors de l'apprentissage d'un modèle ne suffisent pas à la réalisation complète d'une tâche, mais peuvent conditionner (et de manière objective) les résultats qui en découlent : notre base de données, très spécifiques par son contexte, ne permet pas à des algorithmes de référence d'obtenir des performances équivalentes à celles obtenues sur une base de données de la littérature. La différence est principalement due au contenu de ces bases, certaines étant plus orientées pour réaliser la distinction entre scènes violentes et non-violentes, d'autres, comme celle que nous avons développée, se rapprochant plus d'une détection de violence dans un cadre opérationnel. Ce dernier implique de considérer des contraintes supplémentaires non présentes dans les bases de données de référence : nombre de personnes présentes dans une scène, variabilité de distance entre scènes de violences et capteurs, considération des occultations, etc.

Finalement, au regard des résultats que nous avons obtenus sur nos diverses architectures, nous pouvons conclure que dans nos travaux, le traitement conjoint de données audio et vidéo est préférable à la combinaison de décision de modèles audio et vidéo indépendants : sans information *à priori*, ce dernier modèle ne peut facilement prendre une décision simple quand les données observées ne sont pas cohérentes (violences perçues que sur un seul mode), incohérences causées principalement par l'occultation et les différences entre les durées de perceptions sonores et visuelles de la violence. De plus, ces incohérences permettent de justifier de l'utilité même du traitement conjoint. Dans ce sens, les examens des diagrammes de Venn confirment que peu d'erreurs sont communes entre les architectures uni-modales, justifiant l'intérêt de combiner les signaux pour la reconnaissance des violences. Pour finir sur ces conclusions, les meilleurs résultats sont obtenus avec une architecture audio et vidéo qui combine par concaténation à un niveau moyen les caractéristiques de chaque mode provenant de la couche LSTM. Les paramètres de cette architecture sont estimés avec la stratégie d'apprentissage standard, l'apprentissage considérant une information sur le mode de perception de la violence n'ayant pas apporté une différence majeure.

À la suite de ces travaux, plusieurs perspectives sont potentiellement envisageables :

— Il serait intéressant d'approfondir l'estimation des paramètres des architectures en tenant compte des incohérences entre les signaux en optimisant mutuellement différents critères considérant "indépendamment" les diverses annotations *Audio*, *Vidéo* et (ou non) l'annotation *Globale* (ceci éventuellement à différents niveaux de l'architecture). Des tests préliminaires ont été réalisés dans ce sens mais n'ont pas donné de résultats concluants. N'ayant pas eu la possibilité d'approfondir cette idée, nous n'avons pas eu les moyens de la présenter dans cette étude. Il pourrait être bénéfique, dans la même idée, de limiter la rétro-propagation des gradients en fonction des erreurs sur chaque branche modale, ainsi que de faire varier le taux d'apprentissage sur chaque branche en fonction de la perception des violences, etc.

— Toujours dans l'idée d'exploiter ces annotations réalisées indépendamment sur l'audio et la vidéo, une perspective serait de sortir du cadre bayésien défini par le cadre "multi-classes" et de se placer dans un cadre "multi-label". L'objectif serait alors de prédire la présence de plusieurs classes à un instant donné : dans notre contexte, nous aurions à prédire la présence ou non de la classe "violence sur audio" et la présence ou non de la classe "violence sur vidéo". Cette approche laisserait plus de liberté et de souplesse au modèle pour apprendre et définir des espaces propres aux modes audio et vidéo en fonction de la présence ou non de la violence sur chaque mode. Nous avions ouvert notre étude à cette éventualité, mais nous n'avons encore là pas eu le temps de la mettre en œuvre.

— Une autre perspective qui est naturelle dans un système de détection à deux classes est de remplacer la fonction *Softmax* à deux sorties par une fonction *Sigmoid*. Dans cette dernière, la décision d'appartenance à l'une des deux classes se fait par comparaison de la valeur de sortie de la *Sigmoid* à un seuil (égale à 0,5 par définition quand, en équivalence, on détermine la classe par la sortie de la *Softmax* ayant la valeur la plus élevée parmi l'ensemble des sorties). Dans cette éventualité, nous utiliserions le critère de l'*AUC* (et d'autres équivalents) afin d'estimer une valeur de seuil optimisant le nombre de bonnes détections en fonction du nombre de mauvaises détections.

— Une alternative à notre étude serait de ne plus considérer indépendamment les caméras et les microphones d'une même salle. Bien que nous aurions moins de données, nous pourrions alors mettre en place une architecture multi-vue qui tirerait parti de l'installation des capteurs, offrant une vue croisée d'une même salle. Nous

n'avons pas fait le choix de cette stratégie car (1) nous aurions alors réduit le nombre de données de notre base et (2) il aurait été nécessaire d'assurer la synchronisation de tous les capteurs ou a minima une datation suffisamment précise, étape difficile à mettre en place dans ce contexte expérimental en exploitation.

— Il pourrait aussi être intéressant d'évaluer l'impact de la durée des signaux d'entrée sur les architectures multi-modales, en réduisant par exemple le temps d'analyse de 5 secondes à 2 secondes. Cette approche présenterait l'avantage supplémentaire de correspondre aux annotations produites, évitant ainsi le risque de segments annotés comme "Violence" contenant principalement de la "Non Violence". Dans cette même idée une perspective de travailler avec des annotations indexées précisément dans le temps (définition de début et de fin des évènements violents) pourrait éventuellement permettre de mieux appréhender les différences temporelles de perception de la violence entre mode audio et vidéo.

— Nous pourrions envisager d'enrichir notre base de données d'annotations réalisées par plusieurs personnes. Certaines de nos annotations étant subjectives, cette variété d'annotations supplémentaires permettrait de renforcer l'appréciation et le crédit de nos différents degrés (d'intensité de violence, d'occultation), ainsi que la perception même de la présence de violence.

— Enfin, malgré l'acquisition d'un grand jeu de données, la problématique de l'accès aux données demeure un facteur qui impacte les performances. Pour atténuer cette problématique, il serait intéressant d'utiliser des techniques d'augmentation de données multi-modales tout en préservant la corrélation entre les signaux. Ces techniques d'augmentation de données pourraient être appliquées, par exemple, à l'espace des caractéristiques de chaque branche [100]. Une autre solution consisterait à générer des données uni-modales ou multi-modales afin d'effectuer une préestimation généralisante des paramètres des architectures. De plus, il serait pertinent d'approfondir la pondération des données lors de l'apprentissage en utilisant les différentes méta-annotations, afin d'éviter un équilibrage brutal tout en améliorant les performances dans les situations les moins fréquentes. Enfin, lors de la mise en production, il serait intéressant d'évaluer des techniques d'apprentissage continu ou actif.

D'un point de vue plus général et industriel, le déploiement de ces architectures à grande échelle nécessitera de réaliser un travail d'optimisation des performances de calcul. L'objectif sera d'alléger ces architectures afin qu'elles puissent être exploitées sur des calculateurs embarqués en environnement ferroviaire. De plus, il faudra aborder les problématiques légales liées à la protection des données personnelles (CNIL, RGPD, futur RIA) afin de garantir le respect de la vie privée des usagers des transports lors de l'utilisation de ce type de système. Enfin, un travail opérationnel sera requis pour établir, en collaboration avec les services de sûreté, une procédure de gestion des alertes lorsque des violences sont détectées par nos modèles.

Bibliographie

[1] Sajjad Abdoli, Patrick Cardinal, and Alessandro Lameiras Koerich. End-to-end environmental sound classification using a 1D convolutional neural network. *Expert Systems with Applications*, 136 :252–263, 2019.

[2] Jakob Abeßer. USM-SED-A Dataset for Polyphonic Sound Event Detection in Urban Sound Monitoring Scenarios. *CoRR*, 2021.

[3] Sharath Adavanne, Giambattista Parascandolo, Pasi Pertilä, Toni Heittola, and Tuomas Virtanen. Sound Event Detection in Multichannel Audio Using Spatial and Harmonic Features. In *Detection and Classification of Acoust. Scenes and Events Workshops (DCASE-W)*, Budapest, Hungary, September 2016.

[4] Sharath Adavanne, Pasi Pertilä, and Tuomas Virtanen. Sound event detection using spatial features and convolutional recurrent neural network. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 771–775, New Orleans, LA, USA, March 2017.

[5] Relja Arandjelovic and Andrew Zisserman. Look, Listen and Learn. In *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 609–617, Venice, Italy, October 2017.

[6] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. SoundNet : Learning Sound Representations from Unlabeled Video. In *Neural Information Processing Systems (NeurIPS)*, volume 29, pages 892–900, Barcelona, Spain, December 2016.

[7] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt. Sequential Deep Learning for Human Action Recognition. In *Human Behavior Unterstanding (HBU) Int. Workshop*, pages 29–39, Amsterdam, The Netherlands, November 2011.

[8] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *CoRR*, 2018.

[9] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, 2006.

[10] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as Space-Time Shapes. In *IEEE Int. Conf. on Computer Vision (ICCV)*, pages 1395–1402, Beijing, China, October 2005.

[11] Hervé Bourlard and Christian Wellekens. Links between Markov models and multilayer perceptrons. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(12) :1167–1178, 1990.

[12] Mathilde Brousmiche. *Interaction intermodale dans les réseaux neuronaux profonds pour la classification et la localisation d'évènements audiovisuels*. phdthesis, Université de Mons / Université de Sherbrooke, January 2021.

[13] Mathilde Brousmiche, Stéphane Dupont, and Jean Rouat. AVECL-UMONS database for audio-visual event classification and localization. *CoRR*, 2020.

[14] Mathilde Brousmiche, Stéphane Dupont, and Jean Rout. Intra and Inter-Modality Interactions for Audio-Visual Event Detection. In *Int. Workshop on Human-Centric Multimedia Analysis (HuMA)*, pages 5–11, Seattle, WA, USA, October 2020.

[15] Mathilde Brousmiche, Jean Rouat, and Stéphane Dupont. Audio-Visual Fusion And Conditioning With Neural Networks For Event Recognition. In *IEEE Int. Workshop on Machine Learning for Signal Process. (MLSP)*, pages 1–6, Pittsburgh, PA, USA, October 2019.

[16] Emre Cakir, Giambattista Parascandolo, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen. Convolutional Recurrent Neural Networks for Polyphonic Sound Event Detection. *IEEE/ACM Trans. on Audio, Speech, and Lang. Processing (TASLP)*, 25 :1291–1303, June 2017.

[17] João Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A Short Note about Kinetics-600. *CoRR*, 2018.

[18] João Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A Short Note on the Kinetics-700 Human Action Dataset. *CoRR*, 2019.

[19] João Carreira and Andrew Zisserman. Quo Vadis, Action Recognition ? A New Model and the Kinetics Dataset. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733, Honolulu, HI, USA, July 2017.

[20] Claire Charavel and Alexis Gerbeaux. Les vols et violences enregistrés dans les réseaux de transports en commun en 2021. Technical report, Interstats - SSMSI, 2022.

[21] Ming-Yu Chen and Alexander Hauptmann. MoSIFT : Recognizing Human Actions in Surveillance Videos. *CoRR*, September 2009.

[22] Ming Cheng, Kunjing Cai, and Ming Li. RWF-2000 : An Open Large Scale Video Database for Violence Detection. In *IEEE Int. Conf. on Pattern Recognition (ICPR)*, pages 4183–4190, Milano, Italy, January 2021.

[23] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. P-CNN : Pose-Based CNN Features for Action Recognition. In *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 3218–3226, Santiago, Chile, December 2015.

[24] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the Properties of Neural Machine Translation : Encoder–Decoder Approaches. In *Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, October 2014.

[25] François Chollet. Xception : Deep Learning with Depthwise Separable Convolutions. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807, Honolulu, HI, USA, July 2017.

[26] Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-Based Models for Speech Recognition. In *Advances in Neural Information Processing Systems*, volume 28, pages 577–585, Montreal, Quebec, Canada, December 2015.

[27] Heidi Christensen, Jon Barker, Ning Ma, and Phil D. Green. The CHiME corpus : a resource and a challenge for computational hearing in multisource environments. In *Int. Speech (INTERSPEECH)*, pages 1918–1921, Chiba, Japan, September 2010.

[28] Christine Clarin, Judith Ann M. Dionisio, Michael T. Echavez, and Prospero C. Naval. DOVE : Detection of Movie Violence using Motion Intensity Analysis on Skin and Blood. *CoRR*, 2005.

[29] C. Clavel, I. Vasilescu, L. Devillers, G. Richard, and T. Ehrette. Fear-type emotion recognition for future audio-based surveillance systems. *Speech Communication*, 50(6) :487–503, 2008.

[30] Chloé Clavel, Thibaut Ehrette, and Gaël Richard. Events detection for an audio-based surveillance system. In *IEEE Int. Conf. on Multimedia and Expo (ICME)*, pages 1306–1309, Amsterdam, The Netherlands, July 2005.

[31] Federico Colangelo, Federica Battisti, Marco Carli, Alessandro Neri, and Francesco Calabró. Enhancing audio surveillance with hierarchical recurrent neural networks. In *IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6, Lecce, Italy, August 2017.

[32] Ronan Collobert, Christian Puhrsch, and Gabriel Synnaeve. Wav2Letter : an End-to-End ConvNet-based Speech Recognition System. *CoRR*, 2016.

[33] Jason Cramer, Ho-Hsiang Wu, Justin Salamon, and Juan Pablo Bello. Look, Listen, and Learn More : Design Choices for Deep Audio Embeddings. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3852–3856, Brighton, UK, May 2019.

[34] Wei Dai, Chia Dai, Shuhui Qu, Juncheng Li, and Samarjit Das. Very deep convolutional neural networks for raw waveforms. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 421–425, March 2017.

[35] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling Egocentric Vision : The EPIC-KITCHENS Dataset. In *European Conf. on Computer Vision (ECCV)*, pages 753–771, Munich, Germany, September 2018.

[36] Ankur Datta, Mubarak Shah, and Niels Da Vitoria Lobo. Person-on-person violence detection in video data. In *IEEE Int. Conf. on Pattern Recognition (ICPR)*, volume 1, pages 433–438, Rochester, NY, USA, August 2002.

[37] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Tran. on Acoustics, Speech, and Signal Processing*, 28(4) :357–366, 1980.

[38] Gert Dekkers, Steven Lauwereins, Bart Thoen, Mulu Weldegebreal Adhana, Henk Brouckxon, Bertold Van den Bergh, Toon van Waterschoot, Bart Vanrumste, Marian Verhelst, and Peter Karsmakers. The SINS Database for Detection of Daily Activities in a Home Environment Using an Acoustic Sensor Network. In *Detection and Classification of Acoustic Scenes and Events (DCASE)*, pages 32–36, Munich, Germany, November 2017.

[39] Claire-Hélène Demarty, Cédric Penet, Guillaume Gravier, and Mohammad Soleymani. The Mediaeval 2011 affect task : Violent scene detection in Hollywood movies. In *MediaEval 2011 Workshop*, Pisa, Italy, September 2011.

[40] Claire-Hélène Demarty, Cédric Penet, Guillaume Gravier, and Mohammad Soleymani. A Benchmarking Campaign for the Multimodal Detection of Violent Scenes in Movies. In *European Conf. on Computer Vision (ECCV) Workshops and Demonstrations*, pages 416–425, Florence, Italy, October 2012.

[41] Claire-Hélène Demarty, Cédric Penet, Guillaume Gravier, and Mohammad Soleymani. The Mediaeval 2012 affect task : Violent scenes detection. In *MediaEval 2012 Workshop*, Pisa, Italy, October 2012.

[42] Claire-Hélène Demarty, Cédric Penet, Mohammad Soleymani, and Guillaume Gravier. VSD, a public dataset for the detection of violent scenes in movies : design, annotation, analysis and evaluation. *Multim. Tools Appl.*, 74(17) :7379–7404, 2015.

[43] Giovanna Maria Dimitri. A Short Survey on Deep Learning for Multimodal Integration : Applications, Future Perspectives and Challenges. *MDPI Computers*, 11(11), 2022.

[44] Harishchandra Dubey, Dimitra Emmanouilidou, and Ivan J. Tashev. CURE Dataset : Ladder Networks for Audio Event Classification. In *IEEE Pacific Rim Conf.*

on Communications, Computers and Signal Processing (PACRIM), pages 1–6, Auckland, New Zealand, August 2019.

[45] Miquel Espi, Masakiyo Fujimoto, Keisuke Kinoshita, and Tomohiro Nakatani. Exploiting spectro-temporal locality in deep learning based acoustic event detection. *EURASIP J. Audio Speech Music. Processing*, 2015 :26, 2015.

[46] Miquel Espi, Masakiyo Fujimoto, Yotaro Kubo, and Tomohiro Nakatani. Spectrogram patch based acoustic event detection and classification in speech overlapping conditions. In *IEEE Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, pages 117–121, Villers-les-Nancy, France, May 2014.

[47] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. SlowFast Networks for Video Recognition. In *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 6201–6210, Seoul, Korea, October 2019.

[48] A. Fleury, N. Noury, M. Vacher, H. Glasson, and J.-F. Seri. Sound and speech detection and classification in a Health Smart Home. In *IEEE Int. Conf. Engineering in Medicine and Biology Society*, pages 4644–4647, August 2008.

[49] Pasquale Foggia, Nicolai Petkov, Alessia Saggese, Nicola Strisciuglio, and Mario Vento. Reliable detection of audio events in highly noisy environments. *Pattern Recognition Letters*, 65 :22–28, 2015.

[50] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. FSD50K : An Open Dataset of Human-Labeled Sound Events. *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, 30 :829–852, 2022.

[51] Peter Foster, Siddharth Sigtia, Sacha Krstulovic, Jon Barker, and Mark D. Plumbley. CHiME-Home : A dataset for sound source recognition in a domestic environment. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 1–5, New Paltz, NY, USA, October 2015.

[52] Guillermo Garcia-Cobo and Juan C. SanMiguel. Human skeletons and change detection for efficient violence detection in surveillance videos. *Computer Vision and Image Understanding*, 233 :103739, 2023.

[53] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio Set : An ontology and human-labeled dataset for audio events. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780, New Orleans, LA, USA, March 2017.

[54] Oguzhan Gencoglu, Tuomas Virtanen, and Heikki Huttunen. Recognition of acoustic events using deep neural networks. In *IEEE European Signal Processing Conf. (EUSIPCO)*, pages 506–510, Lisbon, Portugal, September 2014.

[55] Luigi Gerosa, Giuseppe Valenzise, Marco Tagliasacchi, Fabio Antonacci, and Augusto Sarti. Scream and gunshot detection in noisy environments. In *IEEE European Signal Processing Conf. (EUSIPCO)*, pages 1216–1220, Poznan, Poland, September 2007.

[56] Theodoros Giannakopoulos, Dimitrios I. Kosmopoulos, Andreas Aristidou, and Sergios Theodoridis. Violence Content Classification Using Audio Features. In *Helenic Conf. on AI - Advances in Artificial Intelligence*, volume 3955, pages 502–507, Heraklion, Crete, Greece, May 2006.

[57] Theodoros Giannakopoulos, Alexandros Makris, Dimitrios Kosmopoulos, Stavros Perantonis, and Sergios Theodoridis. Audio-Visual Fusion for Detecting Violent Scenes in Videos. In *Artificial Intelligence : Theories, Models and Applications*, pages 91–100, Athens, Greece, May 2010.

[58] Theodoros Giannakopoulos, Aggelos Pikrakis, and Sergios Theodoridis. A Multi-Class Audio Classification Method With Respect To Violent Content In Movies Using Bayesian Networks. In *IEEE Workshop on Multimedia Signal Processing*, pages 90–93, Chania, Crete, Greece, October 2007.

[59] Melvyn A. Goodale and A.David Milner. Separate visual pathways for perception and action. *Trends in Neurosciences*, 15(1) :20–25, 1992.

[60] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[61] Chunhui Gu, Chen Sun, David A. Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. AVA : A Video Dataset of Spatio-Temporally Localized Atomic Visual Actions. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6047–6056, Salt Lake City, UT, USA, June 2018.

[62] Karim Guirguis, Christoph Schorn, Andre Guntoro, Sherif Abdulatif, and Bin Yang. SELD-TCN : Sound Event Localization & Detection via Temporal Convolutional Networks. In *European Signal Processing Conference (EUSIPCO)*, pages 16–20, Amsterdam, Netherlands, January 2020.

[63] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. Conformer : Convolution-augmented Transformer for Speech Recognition. In *International Speech Conference*, pages 5036–5040, Shanghai, China (Virtual), October 2020.

[64] Ryo Hachiuma, Fumiaki Sato, and Taiki Sekii. Unified keypoint-based action recognition framework via structured keypoint pooling. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 22962–22971, June 2023.

[65] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning Spatio-Temporal Features With 3D Residual Networks for Action Recognition. In *IEEE/CVF Int. Conf. on Computer Vision Workshops (ICCV-W)*, Venice, Italy, October 2017.

[66] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet ? In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6546–6555, Salt Lake City, UT, USA, June 2018.

[67] Tal Hassner, Yossi Itcher, and Orit Kliper-Gross. Violent flows : Real-time detection of violent crowd behavior. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPR-W)*, pages 1–6, Providence, RI, USA, June 2012.

[68] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, NV, USA, June 2016.

[69] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet : A large-scale video benchmark for human activity understanding. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 961–970, Boston, MA, USA, June 2015.

[70] Toni Heittola, Annamaria Mesaros, Antti J. Eronen, and Tuomas Virtanen. Context-dependent sound event detection. *EURASIP J. Audio Speech Music. Processing*, 2013, 2013.

[71] David W. Henderson. Venn Diagrams for More than Four Classes. *The American Mathematical Monthly*, 70(4) :424–426, 1963.

[72] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin Wilson. CNN architectures for large-scale audio classification. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 131–135, New Orleans, LA, USA, March 2017.

[73] Geoffrey E. Hinton. A Practical Guide to Training Restricted Boltzmann Machines. In Grégoire Montavon, Genevieve B. Orr, and Klaus-Robert Müller, editors, *Neural Networks : Tricks of the Trade - Second Edition*, volume 7700, pages 599–619. Springer, 2012.

[74] Sepp Hochreiter and JürgenSchmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8) :1735–1780, 1997.

[75] Weimin Huang, Tuan Kiang Chiew, Haizhou Li, Tian Shiang Kok, and Jit Biswas. Scream detection for home applications. In *IEEE Conf. on Industrial Electronics and Applications*, pages 2115–2120, Taichung, Taiwan, June 2010.

[76] Bogdan Ionescu, Jan Schlüter, Ionut Mironica, and Markus Schedl. A Naive Mid-Level Concept-Based Fusion Approach to Violence Detection in Hollywood Movies. In *ACM Int. Conf. on Multimedia Retrieval (ICMR)*, pages 215–222, Dallas, TX, USA, April 2013.

[77] Noussaiba Jaafar and Zied Lachiri. Multimodal fusion methods with deep neural networks and meta-information for aggression detection in surveillance. *Expert Systems with Applications*, 211(118523), 2023.

[78] Maxime Janvier, Xavier Alameda-Pineda, Laurent Girin, and Radu Horaud. Sound-event recognition with a companion humanoid. In *IEEE-RAS Int. Conf. on Humanoid Robots*, pages 104–111, Osaka, Japan, November 2012.

[79] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(1) :221–231, Jan 2013.

[80] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale Video Classification with Convolutional Neural Networks. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1725–1732, Columbus, OH, USA, June 2014.

[81] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The Kinetics Human Action Video Dataset. *CoRR*, 2017.

[82] Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. EPIC-Fusion : Audio-Visual Temporal Binding for Egocentric Action Recognition. In *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 5492–5501, Seoul, Korea, October 2019.

[83] Ho-Joon Kim, Joseph S. Lee, and Hyun-Seung Yang. Human Action Recognition Using a Modified Convolutional Neural Network. In *Int. Symposium on Neural Networks (ISNN)*, pages 715–723, Nanjing, China, June 2007.

[84] Zvi Kons and Orith Toledo-Ronen. Audio event classification using deep neural networks. In *Int. Speech (INTERSPEECH)*, pages 1482–1486, Lyon, France, August 2013.

[85] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Neural Information Processing Systems (NeurIPS)*, pages 1097–1105, Lake Tahoe, NV, USA, December 2012.

[86] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso A. Poggio, and Thomas Serre. HMDB : A large video database for human motion recognition. In *IEEE Int. Conf. on Computer Vision (ICCV)*, pages 2556–2563, Barcelona, Spain, November 2011.

[87] Pierre Laffitte. *Automatic Detection of Screams and Shouts in the Metro.* Theses, Université Lille 1 Nord de France, December 2017.

[88] Pierre Laffitte, David Sodoyer, Charles Tatkeu, and Laurent Girin. Deep neural networks for automatic detection of screams and shouted speech in subway trains. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6460–6464, Lujiazui, Shanghai, China, March 2016.

[89] Pierre Laffitte, Yun Wang, David Sodoyer, and Laurent Girin. Assessing the performances of different neural network architectures for the detection of screams and shouts in public transportation. *Expert Systems with Applications (ESWA)*, 117 :29–41, March 2019.

[90] Catherine Lamy-Bergot. Boss : On board wireless secured video surveillance, 2006–2009.

[91] Ivan Laptev. On Space-Time Interest Points. *Int. Journal of Computer Vision (IJCV)*, 64(2-3) :107–123, June 2005.

[92] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, Anchorage, AK, USA, June 2008.

[93] Yann LeCun. Generalisation and network design strategies. Technical report, University of Toronto, 1989.

[94] Iulia Lefter, Gertjan J. Burghouts, and Leon J. M. Rothkrantz. Automatic Audio-Visual Fusion for Aggression Detection Using Meta-information. In *IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance (AVSS)*, pages 19–24, Beijing, China, September 2012.

[95] Hyungui Lim, Jeongsoo Park, Kyogu Lee, and Yoonchang Han. Rare Sound Event Detection Using 1D Convolutional Recurrent Neural Networks. In *Detection and Classification of Acoust. Scenes and Events Workshops (DCASE-W)*, Munich, Germany, November 2017.

[96] Ji Lin, Chuang Gan, and Song Han. TSM : Temporal Shift Module for Efficient Video Understanding. In *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 7082–7092, Seoul, Korea (South), October 2019.

[97] Min Lin, Qiang Chen, and Shuicheng Yan. Network In Network. In *Int. Conf. on Learning Representations (ICLR)*, Banff, AB, Canada, April 2014.

[98] Richard P. Lippmann. Review of Neural Networks for Speech Recognition. *Neural Computation*, 1(1) :1–38, 1989.

[99] Jingen Liu, Jiebo Luo, and Mubarak Shah. Recognizing realistic actions from videos "in the wild". In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1996–2003, Miami, FL, USA, June 2009.

[100] Zichang Liu, Zhiqiang Tang, Xingjian Shi, Aston Zhang, Mu Li, Anshumali Shrivastava, and Andrew Gordon Wilson. Learning Multimodal Data Augmentation in Feature Space. *CoRR*, 2022.

[101] Chih-Yao Ma, Min-Hung Chen, Zsolt Kira, and Ghassan AlRegib. TS-LSTM and temporal-inception : Exploiting spatiotemporal dynamics for activity recognition. *Signal Processing Image Commun.*, 71 :76–87, 2019.

[102] Marcin Marszalek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2929–2936, Miami, FL, USA, June 2009.

[103] Tony Marteau, David Sodoyer, Sébastien Ambellouis, and Sitou Afanou. Level fusion analysis of recurrent audio and video neural network for violence detection in railway. In *IEEE European Signal Processing Conf. (EUSIPCO)*, Belgrade, Serbie, 2022.

[104] Annamaria Mesaros, Toni Heittola, Antti Eronen, and Tuomas Virtanen. Acoustic event detection in real life recordings. In *IEEE European Signal Processing Conf. (EUSIPCO)*, pages 1267–1271, August 2010.

[105] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. TUT database for acoustic scene classification and sound event detection. In *European Signal Processing Conf. (EUSIPCO)*, pages 1128–1132, Budapest, Hungary, August 2016.

[106] Daniel Michelsanti, Zheng-Hua Tan, Shi-Xiong Zhang, Yong Xu, Meng Yu, Dong Yu, and Jesper Jensen. An Overview of Deep-Learning-Based Audio-Visual Speech Enhancement and Separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29 :1368–1396, 2021.

[107] Abdel-rahman Mohamed, George E. Dahl, and Geoffrey E. Hinton. Acoustic Modeling Using Deep Belief Networks. *IEEE Trans. Speech Audio Processing (TASP)*, 20(1) :14–22, 2012.

[108] Hamid Mohammadi and Ehsan Nazerfard. Video violence recognition and localization using a semi-supervised hard attention model. *Expert Systems with Applications*, 212 :118791, 2023.

[109] Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman. Voxceleb : Large-scale speaker verification in the wild. *Computer Speech Language*, 60, 2020.

[110] Satoshi Nakamura, Kazuo Hiyane, Futoshi Asano, Takanobu Nishiura, and Takeshi Yamada. Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition. In *Int. Conf. on Language Resources and Evaluation (LREC)*, Athens, Greece, May 2000.

[111] Jeho Nam, Masoud Alghoniemy, and Ahmed H. Tewfik. Audio-visual content-based violent scene characterization. In *IEEE Int. Conf. on Image Processing (ICIP)*, pages 353–357, Chicago, IL, USA, October 1998.

[112] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets : Deep networks for video classification. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4694–4702, Boston, MA, USA, June 2015.

[113] Juan Carlos Niebles, Chih-Wei Chen, and Li Fei-Fei. Modeling Temporal Structure of Decomposable Motion Segments for Activity Classification. In *European Conf. on Computer Vision (ECCV)*, pages 392–405, Heraklion, Crete, Greece, September 2010.

[114] Enrique Bermejo Nievas, Oscar Deniz Suarez, Gloria Bueno García, and Rahul Sukthankar. Violence Detection in Video Using Computer Vision Techniques. In *Int. Conf. Computer Analysis of Images and Patterns (CAIP)*, pages 332–339, Berlin, Heidelberg, August 2011.

[115] Adrián Núñez-Marcos, Gorka Azkune, and Ignacio Arganda-Carreras. Vision-Based Fall Detection with Convolutional Neural Network. *Wirel. Commun. Mob. Comput.*, 2017, 2017.

[116] Andrew Owens and Alexei A. Efros. Audio-Visual Scene Analysis with Self-Supervised Multisensory Features. In *European Conf. on Computer Vision (ECCV)*, pages 639–658, Munich, Germany, September 2018.

[117] Giambattista Parascandolo, Heikki Huttunen, and Tuomas Virtanen. Recurrent neural networks for polyphonic sound event detection in real life recordings. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6440–6444, Lujiazui, Shanghai, China, March 2016.

[118] Bruno Peixoto, Bahram Lavi, Paolo Bestagini, Zanoni Dias, and Anderson Rocha. Multimodal Violence Detection in Videos. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2957–2961, Barcelona, Spain, May 2020.

[119] Bruno M. Peixoto, Bahram Lavi, Zanoni Dias, and Anderson Rocha. Harnessing high-level concepts, visual, and auditory features for violence detection in videos. *Journal of Visual Communication and Image Representation*, 78 :103174, 2021.

[120] Cédric Penet, Claire-Hélène Demarty, Guillaume Gravier, and Patrick Gros. De la détection d'évènements sonores violents par SVM dans les films. In *ORASIS - Congrès des jeunes chercheurs en vision par ordinateur*, Praz-sur-Arly, France, June 2011. INRIA Grenoble Rhône-Alpes.

[121] Cédric Penet, Claire-Hélène Demarty, Guillaume Gravier, and Patrick Gros. Variability modelling for audio events detection in movies. *Multim. Tools Appl.*, 74(4) :1143–1173, 2015.

[122] Cédric Penet, Claire-Hélène Demarty, Guillaume Gravier, and Patrick Gros. Multimodal information fusion and temporal integration for violence detection in movies. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2393–2396, Kyoto, Japan, March 2012.

[123] Cédric Penet, Claire-Hélène Demarty, Guillaume Gravier, and Patrick Gros. Audio event detection in movies using multiple audio words and contextual Bayesian networks. In *IEEE Int. Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 17–22, Veszprém, Hungary, June 2013.

[124] Mauricio Perez, Alex C. Kot, and Anderson Rocha. Detection of Real-world Fights in Surveillance Videos. In *IEEE Int. Conf. on Acoust., Speech and Signal Processing (ICASSP)*, pages 2662–2666, Brighton, UK, May 2019.

[125] Quoc-Cuong Pham, Agnès Lapeyronnie, Christelle Baudry, Laurent Lucat, Patrick Sayd, Sébastien Ambellouis, David Sodoyer, Amaury Flancquart, Alain-Claude Barcelo, Frédéric Heer, Fabrice Ganansia, and Vincent Delcourt. Audio-video surveillance system for public transportation. In *Int. Conf. on Image Processing Theory, Tools and Applications (IPTA)*, pages 47–53, Paris, France, July 2010.

[126] Huy Phan, Lars Hertel, Marco Maaß, and Alfred Mertins. Robust Audio Event Recognition with 1-Max Pooling Convolutional Neural Networks. In *Int. Speech (INTERSPEECH)*, pages 3653–3657, San Francisco, CA, USA, September 2016.

[127] Karol J. Piczak. Environmental sound classification with convolutional neural networks. In *IEEE Int. Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, Boston, MA, USA, September 2015.

[128] Karol J. Piczak. ESC : Dataset for Environmental Sound Classification. In *ACM Int. Conf. on Multimedia*, pages 1015–1018, Brisbane, Australia, October 2015.

[129] Will Price. *The role of time in video understanding.* PhD thesis, University of Bristol, 2022.

[130] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks. In *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 5534–5542, Venice, Italy, October 2017.

[131] Lawrence Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2) :257–286, 1989.

[132] Kishore K. Reddy and Mubarak Shah. Recognizing 50 human action categories of web videos. *Machine Vision Applications*, 24(5) :971–981, 2013.

[133] Jimmy S. J. Ren, Yongtao Hu, Yu-Wing Tai, Chuan Wang, Li Xu, Wenxiu Sun, and Qiong Yan. Look, Listen and Learn - A Multimodal LSTM for Speaker Identification. In *AAAI Conf. on Artificial Intelligence*, pages 3581–3587, Phoenix, AZ, USA, December 2016.

[134] I. Rodomagoulakis, N. Kardaris, V. Pitsikalis, E. Mavroudi, A. Katsamanis, A. Tsiami, and P. Maragos. Multimodal human action recognition in assistive human-robot interaction. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2702–2706, Lujiazui, Shanghai, China, March 2016.

[135] Mikel D. Rodriguez, Javed Ahmed, and Mubarak Shah. Action MACH a spatio-temporal Maximum Average Correlation Height filter for action recognition. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, Anchorage, AK, USA, June 2008.

[136] Paolo Rota, Nicola Conci, Nicu Sebe, and James M. Rehg. Real-life violent social interaction detection. In *IEEE Int. Conf. on Image Processing (ICIP)*, pages 3456–3460, Québec, Canada, September 2015.

[137] Frank Ruskey, Carla D. Savage, and Stan Wagon. The Search for Simple Symmetric Venn Diagrams. In *Notices of the American Mathematical Society*, pages 1304–1311, 2006.

[138] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.*, 115(3) :211–252, 2015.

[139] Mostafa Sadeghi, Simon Leglaive, Xavier Alameda-Pineda, Laurent Girin, and Radu Horaud. Audio-Visual Speech Enhancement Using Conditional Variational Auto-Encoders. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28 :1788–1800, 2020.

[140] Allah Bux Sargano, Plamen Angelov, and Zulfiqar Habib. A Comprehensive Review on Handcrafted and Learning-Based Action Representation Approaches for Human Activity Recognition. *Applied Sciences*, 7(1), January 2017.

[141] Christian Schüldt, Ivan Laptev, and Barbara Caputo. Recognizing Human Actions : A Local SVM Approach. In *IEEE Int. Conf. on Pattern Recognition (ICPR)*, pages 32–36, Cambridge, UK, August 2004.

[142] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in Homes : Crowdsourcing Data Collection for Activity Understanding. In *European Conf. Computer Vision (ECCV)*, pages 510–526, Amsterdam, The Netherlands, October 2016.

[143] Karen Simonyan and Andrew Zisserman. Two-Stream Convolutional Networks for Action Recognition in Videos. In *Int. Conf. on Neural Information Processing Systems (NeurIPS)*, pages 568–576, Cambridge, MA, USA, December 2014.

[144] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Int. Conf. on Learning Representations (ICLR)*, pages 1–14, San Diego, CA, USA, May 2015.

[145] Nishu Singla. Motion detection based on frame difference method. *Int. Journal of Information & Computation Technology*, 4(15) :1559–1565, 2014.

[146] Lucas Smaira, João Carreira, Eric Noland, Ellen Clancy, Amy Wu, and Andrew Zisserman. A Short Note on the Kinetics-700-2020 Human Action Dataset. *CoRR*, 2020.

[147] Mohamed Mostafa Soliman, Mohamed Hussein Kamal, Mina Abd El-Massih Nashed, Youssef Mohamed Mostafa, Bassel Safwat Chawky, and Dina Khattab. Violence Recognition from Videos using Deep Learning Techniques. In *Int. Conf. on Intelligent Computing and Information Systems (ICICIS)*, pages 80–85, December 2019.

[148] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101 : A Dataset of 101 Human Actions Classes From Videos in The Wild. *CoRR*, 2012.

[149] S. S. Stevens, J. Volkmann, and E. B. Newman. A Scale for the Measurement of the Psychological Magnitude Pitch. *The Journal of the Acoustical Society of America*, 8(3) :185–190, 1937.

[150] Dan Stowell, Dimitrios Giannoulis, Emmanouil Benetos, Mathieu Lagrange, and Mark D. Plumbley. Detection and Classification of Acoustic Scenes and Events. *IEEE Trans. on Multimedia (TMM)*, 17(10) :1733–1746, October 2015.

[151] Yukun Su, Guosheng Lin, Jin-Hui Zhu, and Qingyao Wu. Human interaction learning on 3d skeleton point clouds for video violence recognition. In *European Conference Computer Vision (ECCV)*, pages 74–90, Glasgow, UK, August 2020.

[152] Swathikiran Sudhakaran and Oswald Lanz. Learning to detect violent videos using convolutional long short-term memory. In *IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6, Lecce, Italy, August 2017.

[153] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-World Anomaly Detection in Surveillance Videos. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6479–6488, Salt Lake City, UT, USA, June 2018.

[154] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, Boston, MA, USA, June 2015.

[155] Naoya Takahashi, Michael Gygli, Beat Pfister, and Luc Van Gool. Deep Convolutional Neural Networks and Data Augmentation for Acoustic Event Recognition. In *Int. Speech (INTERSPEECH)*, pages 2982–2986, San Francisco, CA, USA, September 2016.

[156] Graham W. Taylor, Rob Fergus, Yann LeCun, and Christoph Bregler. Convolutional Learning of Spatio-temporal Features. In *European Conf. on Computer Vision (ECCV)*, pages 140–153, Heraklion, Crete, Greece, September 2010.

[157] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *European Conf. on Computer Vision (ECCV)*, pages 252–268, Munich, Germany, September 2018.

[158] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning Spatiotemporal Features With 3D Convolutional Networks. In *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 4489–4497, Santiage, Chile, December 2015.

[159] Du Tran, Jamie Ray, Zheng Shou, Shih-Fu Chang, and Manohar Paluri. ConvNet Architecture Search for Spatiotemporal Feature Learning. *CoRR*, 2017.

[160] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A Closer Look at Spatiotemporal Convolutions for Action Recognition. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6450–6459, Salt Lake City, UT, USA, June 2018.

[161] Ivo Trowitzsch, Jalil Taghia, Youssef Kashef, and Klaus Obermayer. The NIGENS General Sound Events Database. *CoRR*, 2019.

[162] Giuseppe Valenzise, Luigi Gerosa, Marco Tagliasacchi, Fabio Antonacci, and Augusto Sarti. Scream and gunshot detection and localization for audio-surveillance systems. In *IEEE Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, pages 21–26, Londres, UK, September 2007.

[163] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. WaveNet : A Generative Model for Raw Audio. In *Speech Synthesis Workshop (SSW)*, page 125, Sunnyvale, CA, USA, September 2016.

[164] Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-Term Temporal Convolutions for Action Recognition. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 40(6) :1510–1517, June 2018.

[165] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, Long Beach, CA, USA, December 2017.

[166] John Venn. On the diagrammatic and mechanical representation of propositions and reasonings. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 10(59) :1–18, 1880.

[167] John Venn. On the employment of geometrical diagrams for the sensible representations of logical propositions. *Proceedings of the Cambridge Philosophical Society*, pages 47–59, 1880.

[168] Valentin Vielzeuf. *Deep learning for multimodal and temporal contents analysis.* Theses, Normandie Université, November 2019.

[169] Cheng Wang, Haojin Yang, and Christoph Meinel. Exploring multimodal video representation for action recognition. In *Int. Joint Conf. on Neural Networks (IJCNN)*, pages 1924–1931, Vancouver, BC, Canada, July 2016.

[170] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3169–3176, Colorado Springs, CO, USA, June 2011.

[171] Limin Wang, Yuanjun Xiong, Zhe Wang, and Yu Qiao. Towards Good Practices for Very Deep Two-Stream ConvNets. *CoRR*, 2015.

[172] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal Segment Networks : Towards Good Practices for Deep Action Recognition. In *European Conf. on Computer Vision (ECCV)*, pages 20–36, Amsterdam, The Netherlands, October 2016.

[173] Xianyuan Wang, Zhenjiang Miao, Ruyi Zhang, and Shanshan Hao. I3D-LSTM : A New Model for Human Action Recognition. *IOP Conf. Series : Materials Science and Engineering*, 569(3), July 2019.

[174] Yifan Wang, Jie Song, Limin Wang, Luc Van Gool, and Otmar Hilliges. Two-Stream SR-CNNs for Action Recognition in Videos. In *British Machine Vision Conf. (BMVC)*, York, UK, September 2016.

[175] Yun Wang, Leonardo Neves, and Florian Metze. Audio-based multimedia event detection using deep recurrent neural networks. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2742–2746, Lujiazui, Shanghai, China, March 2016.

[176] Daniel Weinland, Rémi Ronfard, and Edmond Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision. Image Understanding*, 104(2-3) :249–257, 2006.

[177] Gary M. Weiss. *Foundations of Imbalanced Learning*, chapter 2, pages 13–41. John Wiley & Sons, Ltd, 2013.

[178] Eric W. Weisstein. Venn Diagram : From MathWorld - A Wolfram Web Resource.

[179] Chung-Hsien Wu, Jen-Chun Lin, and Wen-Li Wei. Survey on audiovisual emotion recognition : databases, features, and data fusion strategies. *APSIPA Trans. on Signal and Information Processing*, 3, 2014.

[180] Peng Wu, Jing Liu, Yujia Shi, Yujia Sun, Fangtao Shao, Zhaoyang Wu, and Zhiwei Yang. Not only Look, But Also Listen : Learning Multimodal Violence Detection Under Weak Supervision. In *European Conf. on Computer Vision (ECCV)*, pages 322–339, Online, August 2020.

[181] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual SlowFast Networks for Video Recognition. *CoRR*, 2020.

[182] Zhenke Yang. *Multi-modal aggression detection in trains*. PhD thesis, Delft University of Technology, Netherlands, 2009.

[183] Md. Zaigham Zaheer, Jin Young Kim, Hyoung-Gook Kim, and Seung You Na. A Preliminary Study on Deep-Learning Based Screaming Sound Detection. In *Int. Conf. on IT Convergence and Security (ICITCS)*, pages 1–4, Kuala Lumpur, Malaysia, August 2015.

[184] Haomin Zhang, Ian McLoughlin, and Yan Song. Robust sound event recognition using convolutional neural networks. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 559–563, South Brisbane, Queensland, Australia, April 2015.

[185] Hong-Bo Zhang, Yi-Xiang Zhang, Bineng Zhong, Qing Lei, Lijie Yang, Ji-Xiang Du, and Duan-Sheng Chen. A Comprehensive Survey of Vision-Based Human Action Recognition Methods. *Sensors*, 19(5), February 2019.

[186] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal Relational Reasoning in Videos. In *European Conf. on Computer Vision (ECCV)*, pages 831–846, Munich, Germany, September 2018.

[187] Yi Zhu and Shawn D. Newsam. Depth2action : Exploring embedded depth for large-scale action recognition. In *European Conf. on Computer Vision Workshops (ECCV-W)*, pages 668–684, Amsterdam, The Netherlands, October 2016.

[188] Rhalem Zouaoui, Romaric Audigier, Sébastien Ambellouis, François Capman, Hamid Benhadda, Stéphanie Joudrier, David Sodoyer, and Thierry Lamarque. Embedded security system for multi-modal surveillance in a railway carriage. In *SPIE Security + Defence*, pages 75–90, Toulouse, France, September 2015.

[189] Şeymanur Aktı, Gözde Tataroğlu Ayşe, and Hazım Kemal Ekenel. Vision-based Fight Detection from Surveillance Cameras. In *IEEE Int. Conf. on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–6, Istanbul, Turkey, November 2019.

Annexe A

Les jeux de données de la communauté

A.1 Les jeux de données audio

Beaucoup de jeux de données audio s'intéressent à la classification d'environnement sonore (**RWCP** [110], **ESC-50** [128], **USM-SED** [2]). Malheureusement, dans ces jeux de données, seulement quelques classes peuvent être utilisées pour faire de la reconnaissance d'action.

• **DCASE2013** [150], introduit en 2015 par Stowell *et al.*, est un des premiers jeux de données introduit dans la communauté traitant spécifiquement la classification de scènes et la détection d'événements. La partie du jeu de données traitant la détection d'événement s'intéresse à 16 classes dans un environnement de type bureau. Elle a été construite avec des enregistrements réels et des enregistrements synthétiques. L'annotation temporelle des exemples d'environ 1 minute du jeu de données a été réalisée manuellement.

• **CHiME-Home** [51], introduit par Foster *et al.* en 2015, est un jeu de données construit à partir du jeu de données du projet CHiME [27]. Ce jeu de données s'intéresse à la reconnaissance de 9 classes représentant des activités humaines dans un environnement domestique.

• **TUT Sound Events** [105], introduit en 2016 par Mesaros *et al.*, est un jeu de données enregistré dans un environnement domestique (intérieur et extérieur). Le jeu de données annoté manuellement s'intéresse à la détection temporelle de 18 classes.

• **SINS** [38], introduit en 2017 par Dekkers *et al.*, est un jeu de données enregistré dans un environnement domestique qui s'intéresse à la détection d'activité quotidienne. Le jeu de données annoté manuellement s'intéresse à la détection temporelle de 16 classes.

• **AudioSet** [53], introduit par Gemmeke *et al.* en 2017, est un grand jeu de données partagé afin de fournir un jeu de données riche dans l'objectif de combler l'écart entre la communauté image et audio et de stimuler le domaine de la reconnaissance audio. Ce jeu de données partage des représentations audio pré-calculées. Il a été construit avec des données agrégées depuis la plateforme de partage de vidéo YouTube. Les observations de 10 secondes ont été annotées manuellement de manière moins précise, c'est-à-dire qui indique la présence d'une classe dans l'exemple sans en préciser les bornes temporelles. Le jeu de données s'intéresse à la reconnaissance de 527 classes.

• **CURE** [44], introduit en 2019 par Dubey *et al.*, est un jeu de données agrégé construit à partir de la plateforme de partage Freesound. Ce jeu de données de seulement 13 classes s'intéresse à des classes présentant un intérêt pour les personnes ayant perdu l'audition. Comme pour AudioSet, les exemples de 5s ont été annotés manuellement de manière moins précise sans préciser les bornes temporelles.

• **NIGENS-SE** [161], introduit par Trowitzsch *et al.* en 2020, est un jeu de données

de sons isolés et de grande qualité construit en agrégeant des données depuis StockMusic une plateforme de vente d'audio. Le jeu de données annoté manuellement s'intéresse à la détection temporelle de 14 classes.

• **FSD50K** [50], introduit en 2022 par Fonseca *et al.* est un jeu de données dans la continuité d'AudioSet qui cherche à résoudre les problématiques des jeux de données petites et/ou spécifiques à un domaine. Ce jeu de données, à la différence d'AudioSet, contient les signaux audio qui ont pu être agrégés depuis la plateforme de partage de son Freesound. Le jeu de données annoté manuellement s'intéresse à la reconnaissance de 200 classes.

A.2 Les jeux de données vidéo

Les jeux de données ont évolués avec le temps et ont accompagné la communauté en complexifiant les tâches.

• **KTH** [141], introduit en 2004 par Schuldt *et al.*, est un des premiers jeux de données public et partagé pour faire de la reconnaissance d'action sur le signal vidéo. Il s'agit d'un jeu de données scénarisés qui permet de traiter 6 classes jouées par 25 personnes dans 4 environnements à l'arrière-plan fixe. L'objectif de ce jeu de données est de reconnaître l'action présente dans l'exemple.

Quelques jeux de données similaires à KTH ont été proposés dans la communauté entre 2004 et 2011. Ces jeux de données augmentaient principalement le nombre d'actions à reconnaître (**Weizmann** [10], **IXMAS** [176], **UCF-Sports** [135], **Hollywood** [92], **Hollywood2** [102], **UCF11 (UCF YouTube)** [99], **Olympic** [113], **UCF50** [132]) passant de 6 à 50 classes.

• **HMDB51** [86], introduit par Kuehne *et al.* en 2011, est un jeu de données, intégrant 51 actions à reconnaître. Ce jeu de données est construit avec des données agrégées (vidéos d'internet et de films) et propose une annotation complète qui intègre de nombreuses métadonnées (occultation du corps : haut, bas ou tout le corps ; mouvement de caméra ; point de vue de la caméra : en face, derrière, à gauche ou à droite ; nombre de personnes prenant part à l'action et qualité de la vidéo : haute, moyenne et basse). Toutes ces informations permettent de mener des évaluations plus précises.

• **UCF101** [148], une extension d'UCF50 est un jeu de données introduit en 2012 par Soomro *et al.*. Ce nouveau jeu de données, construit avec des données agrégées de YouTube, dépasse toutes les statistiques des précédents jeux de données introduits. Le nombre de classes est multiplié par 2, passant de 51 à 101, et la durée totale du jeu de données est multiplié par 5, passant de moins de 6h à 27h, par rapport au jeu de données précédent HMDB51.

• **Sports1M** [80], contrairement aux jeux de données annotés manuellement détaillés précédemment, ce jeu de données se base sur une annotation d'observation faible. Ces annotations accompagnent les vidéos sur la plateforme de partage de vidéo YouTube. Introduit en 2014, par Karpathy *el al.*, ce jeu de données présente des statistiques supérieures aux jeux précédents avec 487 classes et 100 000h de durée totale.

• **Kinetics** [81, 17, 18, 146], introduit entre 2017 et 2020, sont des jeux de données construit avec des vidéos agrégées de la plateforme de partage de vidéo YouTube. Les observations d'environ 10 secondes ont été annotées manuellement de manière moins précise sans préciser les bornes temporelles. Ce jeu de données s'intéresse à la reconnaissance entre 400 classes, dans sa version originale, et 700 classes, dans la dernière version publiée, décrivant des activités humaines.

Les premiers jeux de données de la communauté, détaillés ci-dessus, se sont intéressés à la reconnaissance de la présence d'actions avec une annotation plus ou moins précises

des observations. D'autres chercheurs se sont aussi intéressés à :

— la détection temporelle des actions comme Heilbron *et al.* qui a partagé le jeu de données à large échelle **ActivityNet** [69].

— la détection temporelle et spatiale des actions comme Gu *et al.* qui ont partagé le jeu de données à large échelle **AVA** [61].

Par ailleurs, des jeux de données spécifiques sont partagés comme les jeux de données **Charades** [142] ou **EPIC-KITCHENS** [35] qui s'intéressent aux activités ménagères. Ces jeux de données ont été construits en enregistrant des actions dans les maisons des membres du laboratoire qui partage le jeu de données. Le jeu de données *Charades* s'intéresse à la reconnaissance de 157 actions et 46 objets. Les observations d'environ 30 secondes ont été annotées manuellement. Le jeu de données *EPIC-KITCHENS* s'intéresse à la détection spatiale de 149 actions et 323 objets. Les observations d'une durée variable ont été annotées manuellement.

Aujourd'hui, les jeux de données qui font référence dans la communauté sur les différentes tâches de reconnaissance et de détection spatiale et temporelle des actions sont respectivement Kinetics [146] et AVA [61].

A.3 Les jeux de données audio-vidéo

La majorité des jeux de données de la communauté pour faire de la reconnaissance d'action propose seulement un seul signal, souvent le signal vidéo. Cette forte disponibilité s'explique par l'intérêt de la communauté scientifique à faire de la reconnaissance d'action avec le signal vidéo. Malgré tout, quelques travaux se sont intéressés à la reconnaissance d'action avec les signaux audio et vidéo. Quelques jeux de données présentant ces deux signaux ont été partagés dans la communauté.

• **AVE** [157], introduit en 2018 par Tian *et al.*, est le premier jeu de données audio-vidéo à large échelle traitant la reconnaissance d'événement. Ce jeu de données, annoté manuellement pour faire de la détection temporelle de 28 classes, a été construit en agrégeant des données de la plate-forme de partage de vidéo YouTube.

• **AVECL-UMONS** [13], introduit en 2020 par Brousmiche *et al.*, est le dernier jeu de données audio-vidéo partagé dans la communauté. Ce jeu de données s'intéresse à la reconnaissance d'activité humaine dans un environnement de bureau. Il a été construit avec des données scénarisés qui ont été enregistrés par les auteurs. L'annotation réalisée manuellement permet de traiter une tâche de détection temporelle de 11 actions humaines.

A.4 Les jeux de données spécifiques à la reconnaissance de violence.

Dans ce champ de recherche, peu de jeux de données considèrent seulement le signal audio pour faire de la reconnaissance de violences. Le seul jeu de données considérant une tâche proche de la reconnaissance de violence avec le signal audio est **MIVIA AED** [49], introduit par Foggia *et al.* en 2015. Ce jeu de données s'intéresse aux problématiques de surveillance et plus particulièrement la reconnaissance d'événements de type bris de glace, coup de feu et cris. Les événements sonores sont mixés avec des arrière-plans composés de plusieurs sons. Le mixage des événements avec des arrière-plans permet de contrôler et d'évaluer l'impact du RSB.

Comme pour la reconnaissance d'action humaine, la majorité des jeux de données considérant la reconnaissance de violences partagés dans la communauté ont une annotation qui n'est pas précise, c'est-à-dire que l'annotation informe s'il y a la présence ou non d'une violence sur l'exemple.

• **Hockey Fight** [114] et **Violent Flow - Crowd Violence** [67], sont les premiers jeux de données publics considérant seulement la reconnaissance des actions violentes. Ils ont été partagés respectivement par Bermejo Nievas *et al.* en 2011 et Hassner *et al.* en 2012. Le jeu de données Hockey Fight a été construit en agrégeant des données provenant de match de Hockey de la National Hockey League (NHL). Ce jeu de données est composé de 1000 vidéos de 2 secondes avec une résolution de 720×576 pixels échantillonnées à 25 fps (images par secondes). Le jeu de données Violent Flow - Crowd Fight a quant à lui été conçu en agrégeant des données provenant de la plateforme de partage de vidéos YouTube. Les actions violentes considérées par les auteurs de la base de données se produisent dans des foules. Cette base de données de 246 vidéos présente des caractéristiques de résolution, de durée, d'actions et d'environnements variés. Ces deux jeux de données, qui traitent l'objectif de reconnaissance de violence, sont divisés en deux groupes équilibrés avec violence (Fight) et sans violence (No-Fight).

• **Real Life Violence Situations Dataset** (RLVS) [147] plus récemment a été partagé en 2019 pour faire de la reconnaissance de violences. Ce jeu de données a été construit selon deux stratégies de collecte. Une partie de ce jeu de données a été scénarisée et une autre partie a été agrégée depuis la plate-forme de partage de vidéo YouTube. Cette deuxième partie a été ajouté afin d'éviter la redondance des personnes et de l'environnement dans les vidéos capturées. Au total, ce jeu de données contient 2000 vidéos avec des résolutions et des durées variées.

• **Surveillance Camera Fight** [189] et **RWF-2000** [22] ont été partagés dans la communauté, en 2019 et 2020, pour faire de la reconnaissance de violences. Ces jeux de données ont été construits en agrégeant des données de vidéo-surveillance provenant de la plate-forme de partage de vidéo YouTube. Ces jeux de données contiennent respectivement 1000 vidéos de 2s et 2000 vidéos de 5s. Comme pour le jeu de données Violent Flow - Crowd Violence, ces deux nouveaux jeux de données présentent des caractéristiques de résolution, d'actions et d'environnements variés.

Dans la continuité de l'annotation des observations, comme pour le jeu de données *Sports1M*, certains jeux de données ont été partagés avec une annotation moins précise des observations, c'est-à-dire que l'annotation informe s'il y a la présence d'une violence ou non sur une partie de l'observation. Cette stratégie d'annotation permet d'acquérir des jeux de données à large échelle.

• **UCF-Crime** [153] et **XD-Violence** [180] ont été construits en agrégeant des données provenant de plateforme de partage de vidéo. Le jeu de données UCF-Crime contient des données provenant de caméra de surveillance et couvre 11 types de violences différentes. Le jeu de données XD-Violence contient des données provenant de différents supports couvre 6 types de violences.

D'une autre façon, l'annotation temporelle des événements a été considérée dans la communauté afin de faire de la détection temporelle des violences.

• **VSD** [39, 41, 40, 42], a été le premier jeu de données à considérer cette problématique. Ce jeu de données qui partage seulement des caractéristiques a été construit en agrégeant des vidéos de violences depuis des films. Il a été utilisé dans le cadre de l'atelier de travail sur la détection de violence de la conférence MediaEval entre 2011 et 2014. En plus de traiter la problématique de détection temporelle des violences, le jeu de données VSD permet aussi de distinguer 10 types de violences.

• **CCTV-Fight** [124], plus récemment, a aussi été partagé pour traiter la problématique de détection temporelle de violences. Ce jeu de données a été construit en agrégeant des vidéos contenant des violences depuis la plateforme YouTube. Il contient des données provenant de systèmes de vidéo-surveillance et des données provenant d'un capteur mobile (téléphone portable, caméra de bord, drones ou hélicoptère). Ce jeu de données est composé de 1000 observations de durée variable qui ont été annotées manuellement.

Enfin, l'annotation spatiale des violences dans les vidéos a été considérée seulement dans le jeu de données **RE-DID** [136]. Ce jeu de données a été construit en agrégeant des vidéos contenant des violences depuis la plateforme YouTube. Ce jeu de données est composé de 30 observations de durée variable qui ont été annotées manuellement.

Annexe A

Données

A.1 Pré-traitement

Les scènes de violences ayant été jouées à chaque fois à trois reprises à trois distances des capteurs par les mêmes binômes, il a été nécessaire de découper les vidéos en trois scénarios afin d'avoir seulement une seule violence jouée par scénario. Ces découpages ont été réalisés indépendamment pour chaque caméra, il n'a pas été conservé la synchronisation entre les caméras filmant les mêmes salles. Les fichiers de sorties ont été nommés en suivant la convention suivante :

Ce choix de nommage des scénarios permet d'avoir rapidement les informations suivantes :

1. Lieu d'enregistrement parmi les quatre salles.
2. Type de scénarios.
3. Numéro de la vidéo.
4. Numéro du scénario.
5. Numéro de la caméra.
6. Distance de la violence par rapport à la caméra (Zone1 pour les violences proches, Zone2 pour les violences à moyenne distance et Zone3 pour les violences éloignées).
7. Présence ou non de violence.

Par ailleurs, à cause d'une erreur de configuration sur les caméras de la plate-forme, la résolution de toutes les caméras a été réduite, passant du Full HD (1920×1080) à de l'HD (1280×720) toujours à 25 i/s.

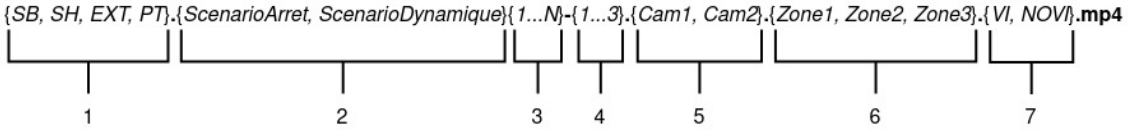


FIGURE A.1 – Organisation des noms des échantillons