Reconnaissance multi-dimensionnelle de l'émotion par apprentissage profond de caractéristiques spatio-temporelles sur séquences vidéo

par

Thomas Teixeira

MÉMOIRE PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
COMME EXIGENCE PARTIELLE À L'OBTENTION DE LA MAÎTRISE
AVEC MEMOIRE EN GÉNIE ÉLECTRIQUE
M. Sc. A.

MONTRÉAL, LE 10 SEPTEMBRE 2020

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE:

M. Alessandro Lameiras Koerich, directeur
Département de génie logiciel et des TI à l'École de technologie supérieure

M. Éric Granger, co-directeur
Département de génie des systèmes à l'École de technologie supérieure

M. Patrick Cardinal, président du jury
Département de génie logiciel et des TI à l'École de technologie supérieure

M. Christian Desrosiers, examinateur externe
Département de génie logiciel et des TI à l'École de technologie supérieure

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 18 AOUT 2020

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

En premier lieu, je souhaite remercier vivement l'ensemble des personnes qui ont contribué à la réalisation de ce mémoire.

Je souhaite remercier mes directeurs de recherche, Alessandro Lameiras Koerich et Éric Granger de s'être rendu disponible, de m'avoir guidé et conseillé dans mes activités de mémoire,ainsi que de m'avoir donner l'opportunité de poursuivre mes études au sein du laboratoire de recherche LIVIA.

Je remercie l'ensemble des étudiants du LIVIA, pour leur aide et conseils dans mes travaux de recherche.

Enfin je tiens à remercier ma famille et mes amis qui m'ont toujours soutenu dans mon parcours.

Reconnaissance multi-dimensionnelle de l'émotion par apprentissage profond de caractéristiques spatio-temporelles sur séquences vidéo

Thomas Teixeira

RÉSUMÉ

L'informatique affective et la reconnaissance d'émotions ont connu un intérêt croissant dans plusieurs domaines de recherche durant ces dernières décennies. En particulier, les expressions faciales représentent un des moyens les plus efficaces pour le relevé des éléments caractéristiques du comportement humain et décrire un état émotionnel. Néanmoins, même pour l'homme, identifier les expressions faciales est une tâche complexe, et les systèmes automatiques de reconnaissance d'expressions faciales (REF) basés sur l'image ont souvent souffert du manque de données pour l'entraînement de modèles d'apprentissage profond de caractéristiques. Avec la transition progressive des systèmes vers l'analyse de séquences vidéo, récoltées en conditions naturelles, et avec des modèles plus complexes de représentation de l'émotion tels que le modèle dimensionnel du circumplex (caractérisant l'émotion selon deux axes de valeurs : la valence et l'excitation), les systèmes REF sont capables d'apprendre des caractéristiques du visage plus précises et plus discriminantes.

Par ailleurs, la plupart des modèles présentés actuellement, basés sur les réseaux de neurones à convolutions (CNNs) et combinés avec des réseaux de neurones récurrents (RNNs) ont été proposés pour reconnaître l'émotion mais souvent repose sur des modèles de représentation de l'affect en catégorie d'émotions. Et encore peu d'études font cas de modèle 3D-CNN pour la reconnaissance d'émotions basée sur des modèles de représentation multi-dimensionnels. De plus encore peu de modèles 3D-CNN pré-entraînés pour des tâches de reconnaissance d'expression faciales sont actuellement disponible. Ce qui au vue de la quantité de données disponibles rend complexe le développement de modèles 3D-CNN.

Dans ce mémoire, nous proposons de développer deux types d'approches faisant actuellement référence pour la représentation de caractéristiques spatio-temporelle du visage et la régression des valeurs de valence et excitation (arousal) de l'émotion. D'une part nous nous sommes intéressés à une architecture en cascade de type CNN-LSTM. D'autres part, une architecture 3D-CNN, pour laquelle nous appliquons le principe d'inflation des poids de modèles 2D-CNN proposé par Carreira & Zisserman (2017) afin d'opérer le transfert d'apprentissage si essentiel à notre application. Le transfert d'apprentissage nous permet en effet ,de manière préliminaire, de spécialiser nos modèles à des applications se rapprochant le plus possible de notre application finale, et ainsi obtenir une meilleure convergence lors de l'apprentissage.

En premier lieu, notre étude fait une description des différentes étapes importantes pour la conception de modèles d'apprentissage automatiques (pré-traitement, transfert d'apprentissage, post-traitement). Nous détaillons ensuite les étapes de développement de chacune des architectures, et les variables inhérentes à leur conception. La conception de modèles i3D (inflated-3D-CNN) se montre notamment très flexible quant à l'initialisation des paramètres des modèles et nous a permis de développer une nouvelle technique d'apprentissage. Grâce à

l'inflation des poids, il est notamment possible de faire la distinction entre les poids initiaux 2D et les poids étendus, différenciant ainsi les poids associés au domaine spatial de ceux associés au domaine temporel. Enfin la dernière partie, détaille les résultats expérimentaux de nos différentes approches expérimentales validant plusieurs hypothèses de la littérature quant aux modèles à convolutions.

Mots-clés: Informatique affective, Reconnaissance d'émotions, Expressions faciales, Apprentissage profond, Modèle du circumplex pour la représentation de l'émotion, Réseaux de neurones à convolutions, 3D-CNN, Transfert d'apprentissage

Multi-dimensional emotion recognition with deep learning of spatio-temporal features on video sequences

Thomas Teixeira

ABSTRACT

Affect computing and emotion recognition have shown an increased interest in several research areas for the past decades. Notably, facial expressions are one of the most powerful ways for depicting specific patterns in human behavior and describing human emotional state. Nevertheless, even for human, identifying facial expressions is difficult, and automatic facial expression recognition (FER) systems based on images have often suffered from a lack of various and cross-cultures training data. With the slight shift to video sequences with in-the-wild settings and more complex emotion representation such as dimensional models, deep FER systems has the ability to learn more accurate and discriminative features.

Furthermore, most models, based on Convolutional Neural Networks (CNNs) and combined with Recurrent Neural Networks (RNNs), have been proposed for recognizing emotions but often lied on short video sequences for categorical model predictions. And still, few studies are interested in 3D-CNN models for recognizing emotion and based on multi-dimensional representation. Moreover, few pre-trained 3D-CNN models are currently available for FER tasks. Which make the development of 3D-CNN more complex, regarding the amount of available data.

In this thesis, we propose to develop two approaches for representing spatio-temporal face features and performing the regression of valence/arousal values for emotion. On the one hand, we have a deep look into a cascaded network with a CNN-LSTM architecture, as a baseline. On the other hand, we developped a 3D-CNN architecture, thanks to the weights inflation of pre-trained 2D-CNN models, a method proposed by Carreira & Zisserman (2017) in order to operate the essential transfer learning for our application. Indeed Transfer learning allows us preliminary to specialize our models to applications close to our final application, thus obtaining better convergence for model learning.

Firstly, our study describe the different main stages for the design of deep FER models (preprocessing, transfer learning, post-processing). Then, we detail, the development steps of each architecture, and the related variables for our approach. The design of i3D models showed particularly flexible regarding the initialization of model parameters and allowed us to develop a new fine tuning method of deep architecture. Thanks to the weight inflation method, it is possible to make distinction between initial 2D weights and extended weights, thus differenciating weights associated to the spatial domain from weights associated to the temporal domain. Finally, the last part, details the experimental results of our different approaches validating several assumptions from the litterature regarding convolutional models.

Keywords: Affect Computing, Emotion recognition, Facial expressions, Deep learning, Dimensional model of emotion, Circumplex model for emotion representation, Convolutional Neural Networks, 3D-CNN, Transfer learning

TABLE DES MATIÈRES

Page

LISTE DES TABLEAUX

Page

LISTE DES FIGURES

Page

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

REF	Reconnaissance d'Expressions Faciales
FACS	Facial Action Coding System
LBP	Local Binary Pattern
SIFT	Scale Invariant Feature Transform
SVM	Support Vector Machine
CNN	Convolutionnal Neural Network
RNN	Recurrent Neural Network
LSTM	Long-Short Term Memory
CCC	Concordance Correlation Coefficient
MTCNN	Multi-Task Convolutional Neural Network
ReLU	Rectified Linear Unit
MAE	Erreur Moyenne Absolue (Mean Absolute Error)
PEAM	Pourcentage d'erreur absolue Moyenne
PCC	Pearson Correlation Coefficient
CCC	Concordance Correlation Coefficient
GPU	Graphical Physical Unit
MIL	Multiple Instance Learning

INTRODUCTION

L'informatique affective et la reconnaissance d'émotions ont connu un intérêt croissant ces dernières décennies dans plusieurs domaines de recherches. Les expressions faciales, notamment, sont l'une des méthodes les plus efficaces pour le relevé de schémas et formes caractéristiques particulières du comportement d'êtres humains permettant ainsi de décrire un état émotionnel. Plus généralement, la reconnaissance des émotions a montré un intérêt pratique nouveau pour la robotique (l'amélioration de l'interaction entre homme et machine), la découverte de nouveaux traitements médicaux (grâce à la détection du stress ou encore de la dépression) mais aussi dans l'anticipation de comportements en situation de danger comme pour la voiture autonome. La recherche scientifique dans ce domaine a donc pour but d'encoder l'information émotionnelle humaine à partir de biais psychologiques, de données physiologiques humaines et de sujets intrinsèquement différents.

Originellement, l'informatique affective a été instanciée par Picard (1995), qui a permis de poser les bases du transcodage de l'information émotionnelle par la machine, d'une part pour signifier quels sont les enjeux de la transcription de l'émotion et quelles étaient les implications pratiques (i.e. quelles réactions sont attendues de la part de la machine). Depuis l'informatique affective est devenue une branche moderne de l'informatique, suscitant un grand intérêt dans la recherche pour les sciences cognitives, mais aussi pour les sciences des données.

Avec les récentes avancées au début des années 2010 en termes de capacité de calcul avec les Graphical Processing Units (GPUs), l'apprentissage profond de représentations à partir de sources de données variées est aujourd'hui devenu incontournable afin de constamment améliorer la performance de systèmes automatiques experts en relevé de caractéristiques. Les GPUs sont des processeurs graphiques, initialement installés sur des cartes graphiques pour entre autres le rendu 3D ou la gestion de la mémoire vidéo. Aujourd'hui ces circuits intégrés sont largement utilisés pour l'apprentissage machine du fait de leur important parallélisme pour le

calcul matriciel. De plus en plus de modèles se développent et montrent de meilleures capacités de généralisation transposable à diverses tâches de reconnaissance et de détection. Par ailleurs, la quantité et la nature des données récoltées pour la reconnaissance d'émotions s'accroît de jour en jour et dynamise particulièrement la recherche dans le domaine. Notamment, les sources de données disponibles pour la reconnaissance d'expressions faciales ont fait progressivement la transition entre deux domaines différents, mais servant la même application. D'images de sujets pris en laboratoire, permettant la catégorisation de l'émotion, de plus en plus d'études font cas de séquences vidéos de personnes en conditions réelles sur des modèles multidimensionnels permettant un relevé plus fin de leurs états émotionnels.

Problématique

La reconnaissance d'expressions faciales est une tâche complexe que même les sciences humaines peinent à expliquer, ainsi nos espoirs de comprendre au mieux la psyché humaine résident dans le développement de modèles toujours plus complexes tirant profit de données. Cependant le choix de modèles de représentation efficaces de l'émotion est essentiel afin de donner sens et précision aux systèmes de reconnaissance automatique des expressions faciales. Le modèle en catégorie d'émotions proposé par Ekman (1994) dans les années 1990 s'est imposé ces dernières décennies comme modèle de référence, cependant celui-ci est de plus en plus remis en cause pour sa simplicité qui n'est pas généralisable interculturellement, mais peut cependant donner de bons indicateurs émotionnels. Laissant ainsi place aux modèles multidimensionnels plus précis.

Par ailleurs, l'analyse de séquence vidéo de l'émotion relève d'une analyse particulièrement difficile, associée à de nombreuses phases de bruit, comme les occlusions dues à l'environnement réel, les différentes poses de la tête, les conditions d'illumination ou encore les biais d'identité de la personne (genre, âge, ethnie). Cette grande variabilité de phénomènes associée à l'analyse vidéo impose naturellement aux architectures de réseaux de neurones de se montrer

d'autant plus robustes à de nouveaux exemples pour l'apprentissage machine dont le nombre de paramètres et ressources nécessaires aux calculs s'accroît proportionnellement à la complexité du problème. Il est nécessaire de trouver un compromis satisfaisant entre base de données et algorithmes de détection de l'émotion. Ce qui, à l'heure actuelle, relève d'un certain défi quant à la disponibilité et nature des ensembles de données.

Avoir suffisamment de données annotées relève en effet d'un certain défi pour le développement de systèmes REF. Afin d'obtenir une certaine robustesse de nos modèles pour l'analyse de nouvelles données, il est nécessaire de disposer de données d'entraînement présentant une bonne variation de populations (diversité inter-culturelles) s'exprimant dans des environnements diverses. Ainsi, l'ensemble de ces difficultés imposent de concevoir des architectures plus efficaces dans la compréhension des données, notamment par des phases de pré-traitement, d'initialisation des systèmes d'apprentissage et de post-traitement particulièrement adaptées à la régression linéaire de modèles multidimensionnels de l'émotion.

Objectifs

L'objectif principal de ce mémoire est de développer un système performant pour la reconnaissance d'expressions faciales sur des séquences vidéos prises en conditions réelles de stimulation de l'émotion et avec une représentation suivant le modèle du circumplex qui s'agit d'un modèle multidimensionnel de l'émotion constitué de deux axes de valeurs : valence et excitation (arousal). Ce modèle de représentation étend et complète le modèle en catégorie, mais redéfinit grandement notre manière de concevoir les systèmes de Reconnaissance d'Expressions Faciales (REF). Ainsi nous allons développer différents modèles d'apprentissage profond permettant la détection de valeurs continues du circumplex et faire état de leur performance. Notre méthode se basera sur différentes architectures de réseaux à convolutions permettant la représentation de séquences vidéos dans de nouveaux espaces de caractéristiques spatio-temporelles. Ces architectures encodent de manière particulière l'information spatio-temporelle, notre ob-

jectif sera ainsi de les comparer et d'analyser l'apprentissage des informations spatiales et temporelles qui s'opère de manière séquentielle ou simultanée. En détail, notre méthode sera constituée des étapes suivantes :

- implémentation d'une méthode de transfert d'apprentissage afin d'initialiser adéquatement notre modèle de référence de type réseaux de neurones à convolutions (CNN)
- implémentation d'une méthode de pré-traitement permettant la détection de valeur continue pour des mini-clips vidéos ;
- implémentation d'une méthode d'expansion d'architecture à convolutions 2D en 3D adaptée à des modèles de régression pour la reconnaissance d'expressions faciales ;
- évaluation de l'impact de dilutions temporelles de convolutions 3D.

Contributions

La contribution principale de notre étude est l'implémentation de modèles REF en cascade avec convolutions 2D et réseaux de neurones récurrents ainsi que de modèles à convolutions 3D pour la régression des valeurs de valence et excitation. Ces modèles aujourd'hui reférence pour des applications de vision par ordinateur nous permettent de répondre à nos objectifs et de mettre en oeuvre notre méthodologie. Les modèles en cascade ont pour particularité d'encoder l'information spatio-temporelle de manière séquentielle tandis que les 3D-CNN le font de manière simultanée. Néanmoins nos contributions mineures sont en partie les suivantes :

- l'évaluation expérimentale des performances de modèles à convolutions 2D et 3D pour la détection des émotions avec différents types de pré-entraînement ;
- l'évaluation des performances des modèles suivant différents paramètres inhérents au pré-traitement des séquences vidéos (longueur des mini-clips, fusion des annotations) ;
- l'évaluation des performances des modèles suivant l'initialisation de paramètres variés pour les modèles 3D-CNN ;

- l'application de la dilution temporelle de convolutions à des modèles à convolutions 3D.

Organisation du document

Le document est organisé de la manière suivante. Dans le premier chapitre, nous présentons les principaux défis que représente la reconnaissance d'émotions au travers de l'état de l'art présent dans la littérature. Nous décrivons tout d'abord succinctement les différentes sources d'informations et modèles de représentation de l'émotion. Dans un bref historique, nous énumérons les différentes méthodes de détection de l'émotion développées au cours des dernières décennies, puis détaillons le développement des différentes bases de données nécessaires à l'apprentissage des systèmes automatiques de détection de l'émotion. Ensuite nous exposons successivement la chaîne de processus pour la détection de l'émotion (pré-traitement, domaine de représentation des expressions faciales, prédictions), puis nous listons les différentes méthodes et algorithmes d'apprentissage machine issues de la littérature.

Le second chapitre présente notre approche par l'intermédiaire des différentes étapes de conception d'un modèle de représentation spatio-temporelle de l'émotion. Ceci comprend les étapes de transfert d'apprentissage, de pré-traitement des données, de développement d'architectures à base de réseaux de neurones à convolutions et enfin les étapes de post-traitement permettant l'optimisation des modèles.

En troisième chapitre, nous examinons les différents ensembles de données utilisés pour le transfert d'apprentissage ainsi que pour notre application, puis analysons nos résultats expérimentaux démontrant la pertinence ou non de certains paramètres variables inhérents à notre méthodologie.

Le quatrième chapitre conclut notre étude en mettant en relief les principaux résultats, identifie les éléments ayant limité les performances ou pouvant les améliorer. Puis finalement dans quelle direction de futurs travaux relatifs à notre étude pourraient être conduits.

CHAPITRE 1

REVUE DE LITTÉRATURE

Dans cette partie, nous développons les principales implications du développement d'un système REF. Premièrement nous décrirons de manière générale les sources de l'émotion, les différents modèles de représentation de l'émotion et nous ferons un bref historique des travaux pionniers dans la détection des expressions faciales. Nous décrirons ensuite les éléments essentiels à la conception d'un système REF pour l'analyse spatio-temporelle de vidéos puis en ferons un examen détaillé selon les différentes étapes développées : pré-traitement, algorithmes d'apprentissage machine, et post-traitement.

1.1 Sources d'information de l'émotion

Par le biais de la vision par ordinateur et de l'apprentissage machine plusieurs modèles de détection et de reconnaissance des émotions ont été proposés, selon différentes représentations, mais également via différents médiums pour la récolte de données statistiques. En plus des expressions faciales, d'autres éléments physiques et physiologiques de l'être humain font également l'objet d'études pour la détection et la transcription de l'information émotionnelle. Le Tableau 1.1 ci-après liste les éléments physiologiques associés à leurs moyens techniques pour leur détection.

Parmi l'ensemble de ces mesures physiologiques, les expressions faciales présentent plusieurs avantages majeurs. D'une part elles présentent un moyen non-intrusif de mesures puisqu'elles sont relevées à distance par l'intermédiaire d'une caméra et ne nécessitent donc pas de contact direct avec les sujets, de plus les moyens techniques mis en œuvre sont à bas coût, puisque seule une caméra bon marché est nécessaire pour la récolte de données. D'autres part, il a été montré que les expressions faciales représentaient une source particulièrement abondante d'informations pour la communication des émotions (Darwin, 2013) (Tian *et al.*, 2001). De ce fait les expressions faciales constituent le signal le plus efficace et universel pour les êtres humains pour partager leurs états émotionnels et leurs intentions. De ce fait les expressions

Tableau 1.1 Mesures physiologiques associées à leur moyen techniques de détection
Tiré de Greene *et al.* (2016)

Moyen Technique	**Mesure Physique**
Activité Cérébrale	Électro-Encéphalographie (EEG)
Activité Cardiaque	Électrocardiographie (ECG)
Conductance Cutanée	Activité Électrodermale (EDA) / Réponse Cutanée Galvanique (GSR)
Activité Sanguine	Photopléthysmographie
Activité Musculaire	Électromyographie
Activité Respiratoire	Génération Électromagnétique/Piézoélectrique
Expressions Faciales	Automated Facial Expression Analysis (AFEA)
Activité Oculaire	IR Eye Tracking
Mouvements Corporels	Leveraging AFEA

faciales constituent le signal le plus efficace et universel pour les êtres humains pour partager leurs états émotionnels et leurs intentions.

1.2 Modèles de représentation de l'émotion

À partir de ces différents signaux physiques, il convient alors d'établir un système d'identification du statut émotionnel d'une personne et de caractériser celui-ci suivant un certain état dans un domaine particulier. Pour cela plusieurs modèles de représentation ont été proposé pour caractériser l'émotion.

1.2.1 Modèles en catégories

Dans un premier temps, Ekman & Friesen (1971) ont eu l'idée de créer un système de catégorisation de l'émotion supposant qu'il existe des similitudes dans l'expression des sentiments au travers des différentes cultures. Six expressions faciales types ont été identifiées : Joie, Tristesse, Colère, Dégoût, Peur, et Surprise. Par la suite se sont rajoutés, la Neutralité et le Mépris. Cependant, l'universalité de ce modèle parmi les différentes cultures est aujourd'hui facilement remise en question, et il est généralement admis que celui-ci manque de précision et introduit de nombreux biais dans l'identification de l'émotion (Jack *et al.*, 2012). La Figure 1.1

illustre huit catégories d'émotions basiques. De gauche à droite et de haut en bas : Neutre, Joie, Tristesse, Surprise, Peur, Dégoût, Colère, et Mépris.

Figure 1.1 Catégorisation de l'émotion d'après l'ensemble de données AffectNet
Tirée de Mollahosseini & Mahoor (2019)

1.2.2 Modèles dimensionnels et FACS

Afin de pallier au manque de complexité du modèle en catégorie, d'autres modèles tels que le «Facial Action Coding System (FACS)» ainsi que le modèle continu multidimensionnel sont apparus. Plus communément le modèle dimensionnel le plus utilisé est celui du circumplex (Posner *et al.*, 2005) (Warr *et al.*, 2014), caractérisant les émotions selon deux axes (valence et excitation), chacun de ses axes prenant des valeurs continues entre -1 et 1. La valence représente le degré de positivité de l'émotion alors que l'excitation définit le degré d'agitation. Afin de mieux se représenter le modèle dimensionnel, on peut placer certaines catégories au sein du circumplex en attribuant des coordonnées à une émotion et en l'associant à un couple de valeur (valence, excitation).

Typiquement un caractère dépressif aura pour coordonnées $(-1,-1)$ alors qu'une personne excitée au terme d'une victoire sportive aura une expression proche de $(+1,+1)$. L'émotion neutre étant placée au centre du circumplex et ayant pour coordonnées $(0,0)$. La Figure 1.2 illustre le modèle du circumplex.

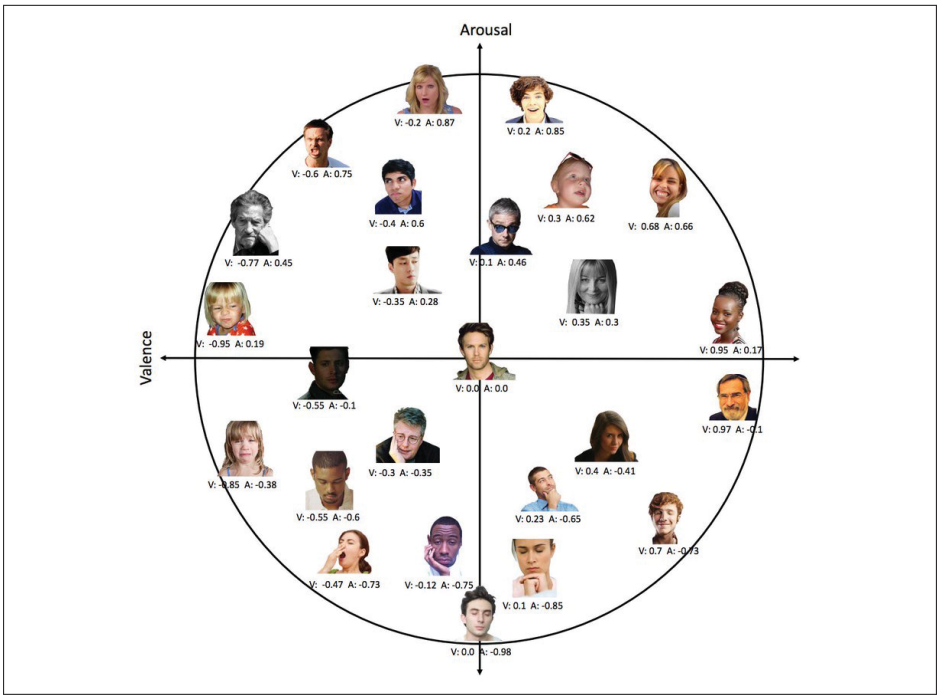

Figure 1.2 Modèle du circumplex
Tirée de Mollahosseini *et al.* (2016)

Le FACS développé par Ekman & Friesen (1978), est un autre mode de description de l'émotion se basant quant à lui sur le mouvement des muscles faciaux. À la différence des modèles présentés précédemment où le support de caractéristiques de l'expression faciale reste à définir. Le FACS divise un visage au préalable en 28 ou plus « Action Units (AUs) » et identifie un type d'émotion en fonction de la combinaison des activations de celles-ci. Une AU s'active lors du mouvement de points particuliers du visage en trois dimensions. Le FACS est un système fournissant un support de caractéristiques définissant des mouvements du visage qu'il reste à associer à une émotion particulière.

1.3 Bref historique du développement des bases de données et systèmes de reconnaissance d'expressions faciales

Tout d'abord, on différencie deux classes de systèmes pour la conception de modèles de reconnaissance d'émotions : méthodes statiques et méthodes dynamiques. À ses origines, la recherche s'est majoritairement focalisée sur une méthode statique, qui consistait à étudier les variations des textures de l'image ou la géométrie de points particuliers du visage. À cette méthode dite « statique » s'oppose la méthode « dynamique » qui s'intéresse quant à elle à la relation temporelle existant entre chaque élément d'une séquence d'images. De plus, ces modèles reposaient majoritairement sur une représentation discrète de l'émotion, regroupant les différents états émotionnels en catégories (généralement les classes d'expressions universelles (Ekman & Friesen, 1971).

L'analyse des expressions faciales consistant ensuite à définir un espace de représentations de caractéristiques du visage pouvant être associé à un état émotionnel. Pour cela les méthodes traditionnelles étaient artisanales et se reposaient sur des techniques comme les motifs binaires locaux ou Local Binary Pattern (LBP) et ses variantes LBP-TOP (Shan *et al.*, 2009; Liu *et al.*, 2014), Volume Local Binay Pattern (VLBP) (Zhao & Pietikainen, 2007b) ou encore Non-negative Matrix Factorization (NMF) (Zhi *et al.*, 2011), le « sparse learning » (Zhong *et al.*, 2012) ou les descripteurs SIFT (Lowe, 1999).

Cependant, la mise en œuvre de ces méthodes, numériquement lourdes en calcul et peu fiables en conditions réelles, ont été remise en question notamment grâce au développement de larges bases de données et de l'apprentissage profond. Depuis les années 2010, cette avancée dans le domaine a permis de concevoir des espaces de représentation du visage plus complexes et permis une plus forte capacité de généralisation des modèles, directement à partir de la distribution de données. En effet, disposer de suffisamment de données est devenu une étape importante dans la conception de systèmes de reconnaissance d'émotions, afin de bénéficier de l'entraînement de système d'apprentissage profond. Disposer de données récoltées en conditions réelles, soit dans des environnements non contraints, et ce de manière opposée au contexte contrôlé et simulé du laboratoire, est devenu par ailleurs un critère essentiel dans le développement de tel

système, en particulier pour améliorer leur robustesse et leur fiabilité. Les bases de données telles que CK+ (Tian *et al.*, 2001), MMI (Sagonas *et al.*, 2015), TFD (Susskind *et al.*, 2010) sont typiquement constituées d'images et séquences simulées en laboratoire pour lesquels un label a été attribués suivant six, voire sept catégories d'émotions. Plus récemment, les bases de données comme FER2013 (Goodfellow *et al.*, 2013), AFEW (Dhall *et al.*, 2011a), RAF-DB (Li *et al.*, 2017) ou AffectNet (Mollahosseini & Mahoor, 2019) ont permis d'engranger de plus grandes quantités d'images et séquences prises en conditions réelles et de natures plus complexes. La Figure 1.3 développe la chronologie des datasets utilisés avec les algorithmes associés. Li & Deng (2018) ont fourni une analyse plus détaillée des ensembles de données évoqués précédemment que nous avons résumé par le Tableau 1.2

Un autre critère important du développement de nouvelles bases de données pour la reconnaissance d'émotions, est la croissance de sources de données comme Internet ou de contenu multimédia comme les films amateurs ou professionnels mis à disposition pour la recherche. La récolte de données a donc pu être réalisée notamment grâce à la puissance de moteurs de recherche et de méthodes de production participative ou « crowd sourcing ». Par exemple les bases de données RAF-DB et AffectNet ont utilisés un certain nombre de mots clés afin de réunir une quantité d'images suffisantes présentant l'expression d'un certain affecte. De plus, la base de données SEWA-DB a été établie grâce à plusieurs volontaires participants à une expérience collective. Cette expérience demandait aux sujets de se mettre mutuellement en scène afin de partager leurs émotions.

Enfin, la plupart des systèmes de reconnaissance d'émotions à l'heure actuelle fonctionnant en mode supervisé, c'est-à-dire que chaque source de donnée est associée à une annotation vers laquelle le système doit idéalement identifier et faire correspondre la source. Il est nécessaire d'attribuer un niveau suffisant de confiance dans ces annotations, principalement afin d'améliorer la convergence de ces systèmes. C'est pourquoi les méthodes de production participatives permettent de faire intervenir un grand nombre d'annotateurs et de minimiser les biais d'annotations des données et finalement obtenir une meilleure généralisation. Puisque plusieurs personnes interviennent dans l'annotation des données, il est nécessaire de définir

des techniques d'attribution d'une seule et unique annotation pour chaque donnée, pour cela des techniques comme le calcul de moyennes ou le vote majoritaire sont employées.

La détection d'expressions faciales spontanées dans la nature reste un sujet complexe qui relève de nombreuses variables. Dépendamment de différents biais d'identité, tels que le genre, l'âge, la culture, les origines ethniques, mais aussi de la qualité de la source (illumination, orientation du visage, contexte, occlusions), les systèmes doivent pouvoir traiter une grande variabilité de données soumises à différentes sources de bruit que mêmes les sciences humaines et cognitives peinent à analyser.

Pour un déploiement de système REF dans un contexte réel, il est crucial de disposer de larges ensembles de données présentant un nombre suffisant de sujets s'exprimant dans un cadre naturel afin de s'assurer de la véracité de l'émotion. Une des faiblesses du développement de tels modèles repose sur leur mode d'apprentissage supervisé et la subjectivité des annotations associées aux expressions faciales. Il est encore aujourd'hui difficile de s'absoudre d'une telle méthode d'apprentissage nous permettant d'associer expressions faciales avec un certain domaine de représentation de l'émotion. En général, afin de limiter le biais des annotations, plusieurs annotateurs issus de différentes cultures (Mollahosseini & Mahoor (2019); Kollias *et al.* (2019)) ont pour tâche de fournir les valeurs cibles d'un ensemble de données, et seul les valeurs les plus fréquentes ou les moyennes sont calculées afin d'obtenir une meilleur certitude et précision. Dans un contexte naturel, il est inévitable dans certains cas d'avoir des visages présentant des occlusions, différentes poses de la tête, ou particulièrement dans le cas de vidéos d'observer des changements de plans, qui doivent être impérativement pris en compte par les systèmes REF. La qualité mais aussi la continuité de l'expression faciale doivent être assurées afin de ne présenter à nos modèles que l'information utile à la détection de l'émotion. Il s'agira alors soit d'éliminer ce bruit latent lors de phases de prétraitement ou au contraire d'apporter particulièrement à nos modèles une robustesse à ce genre de cas par un processus d'apprentissage. Notamment Wang *et al.* (2019) ont proposé un réseau de neurones basé sur les mécanismes d'attention pour isoler les régions du visage intéressantes pour le relevé de caractéristiques et mieux anticiper les changements de pose de la tête et les occlusions. Nous

verrons dans ce mémoire que ces types de bruits ont faiblement impacté notre travail du fait de la composition et du contexte dans lequel ont été réuni les ensembles de données. Ceci peut faciliter la convergence de nos modèles mais limiter l'échelle d'application des systèmes REF.

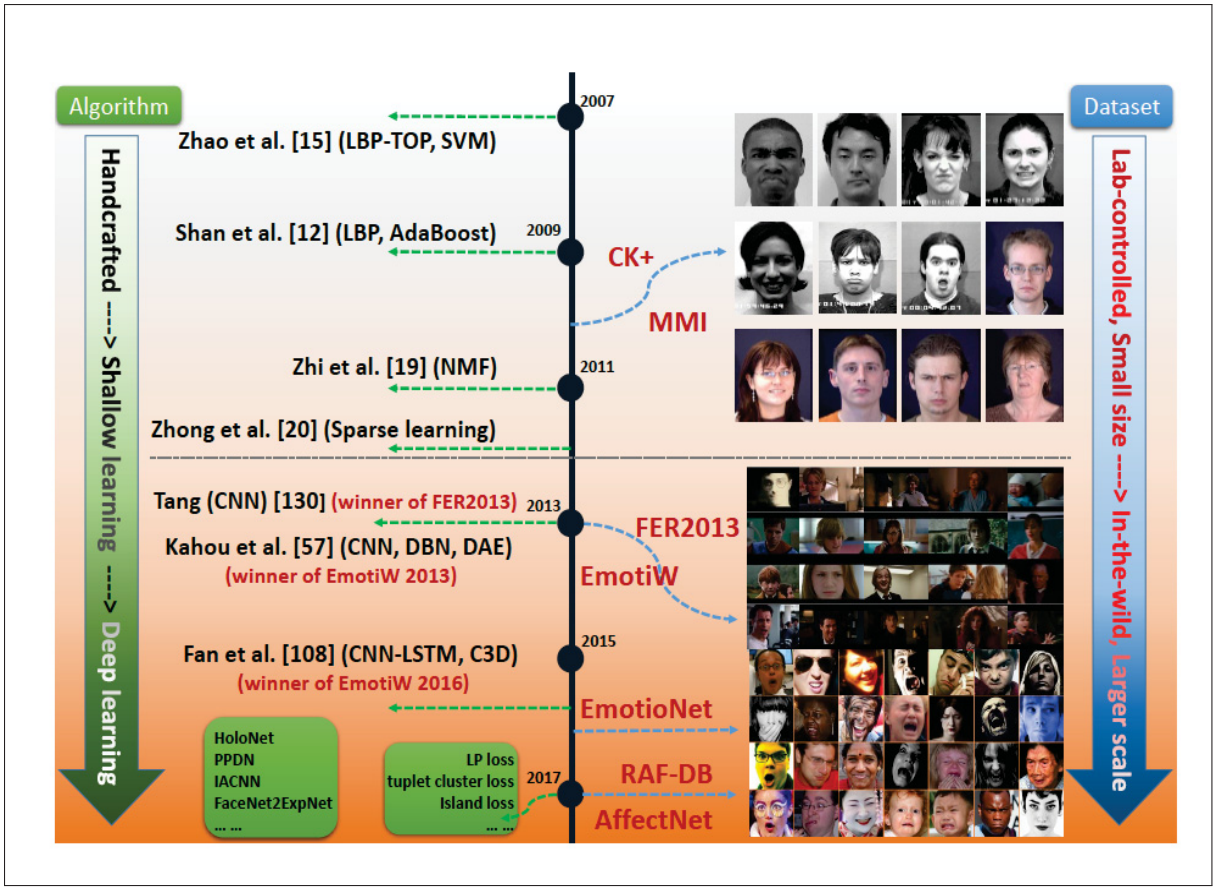

Figure 1.3 Évolution de bases de données et des algorithmes de détection de l'émotion au cours du temps
Tirée de Li & Deng (2018)

1.4 Chaîne de processus pour la détection de l'émotion

Dans cette sous-section nous introduisons, les principales étapes constituant la chaîne de processus de détection de l'affect humain comme décrit par les travaux antérieurs, tels que le pré-traitement des données, la description de l'affect, et enfin la prédiction de l'émotion.

Tableau 1.2 Résumé de principales bases de données pour les expressions faciales
Tiré de Li & Deng (2018)

Base de Données	Nb. Échantillons	Sujets	Cond.	Incit.	Distribution de l'Expression
CK+	593 séquences d'images	123	Lab.	P & S	6 Émotions Basiques + Mépris et Neutre
MMI	740 images et 2,900 vidéos	25	Lab.	P	6 Émotions Basiques + Neutre
JAFFE	213 images	10	Lab.	P	6 Émotions Basiques + Neutre
TFD	112,234 images	NA	Lab.	P	6 Émotions Basiques + Neutre
FER-2013	35,887 images	NA	Web	P & S	6 Émotions Basiques + Neutre
AFEW 7.0	1,809 videos	NA	Film	P & S	6 Émotions Basiques + Neutre
SFEW 2.0	1,766 images	NA	Film	P & S	6 Émotions Basiques + Neutre
Multi-PIE	755,370 images	337	Lab.	P	Sourire, Surpris, Louche, Dégoût, Crie, Neutre
BU-3DFE	2,500 images	100	Lab.	P	6 Émotions Basiques + Neutre
Oulu-CASIA	2,880 séquences d'images	80	Lab.	P	6 Émotions Basiques
RaFD	1,608 images	67	Lab.	P	6 Émotions Basiques + Mépris et Neutre
KDEF	4,900 images	70	Lab.	P	6 Émotions Basiques + Neutre
EmotioNet	1,000,000 images	NA	Web	P & S	23 Émotions Basiques ou Expressions Composées
RAF-DB	29,672 images	NA	Web	P & S	6 Émotions Basiques + Neutre et 12 Émotions Composées
AffectNet	450,000 images	NA	Web	P & S	6 Émotions Basiques + Neutre
ExpW	91,793 images	NA	Web	P & S	6 Émotions Basiques + Neutre

P=Posé, S=Spontané ; Cond.= Condition de collection ; Lab.= Laboratoire, Web, Film ;
Incit. = méthode d'incitation de l'émotion ; NA : Non Acquis

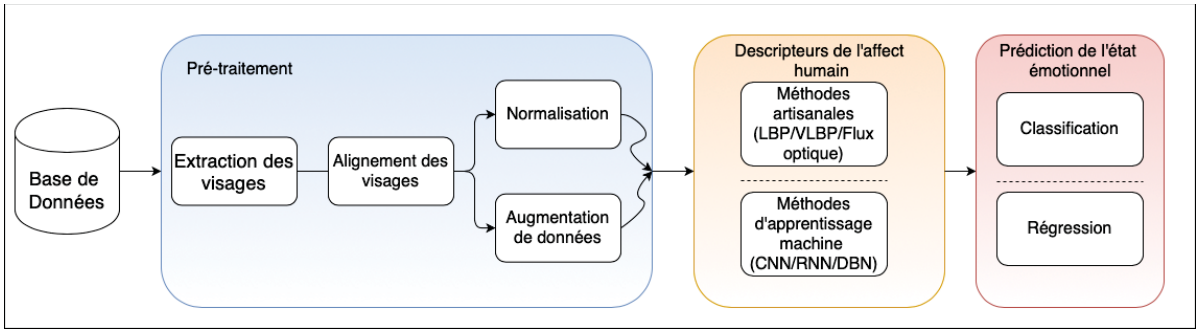

Figure 1.4 Présentation du pipeline sommaire d'un système REF

Pré-traitement : En général, la distribution particulière des données, fait intervenir un certain nombre de phénomènes pouvant être une source non négligeable de bruit et se présente impropre à la détection des expressions faciales au sein de l'image. Opérer certaines modifications préliminaires à l'image permet de maximiser la potentialité de l'information émotionnelle des données et ainsi d'uniformiser l'information qui ne sera pas effective à la détection de l'émotion. Par exemple, la présence de différents contextes, illuminations, poses de la tête dûs à différents environnements et prises de vue. Pour cela des techniques d'alignement du visage et de normalisation sont opérées. De plus l'augmentation de données peut être envisageable dans le cas de modèles d'apprentissage profond.

Descripteurs de l'affect humain : Une fois les données uniformisées au même domaine, le processus suivant est d'extraire l'ensemble des caractéristiques inhérent à chaque instance de la base de données, sous la forme d'un vecteur de dimension fixe appartenant à un espace de caractéristiques particulier, dans lequel ces vecteurs pourront être projetés. Comme nous l'avons vu précédemment, historiquement différentes techniques se sont imposées et ont incroyablement évoluées, l'apprentissage profond marquant une révolution incroyable dans la description spatio-temporelle d'images.

D'une part les méthodes artisanales (comme LBP, LBP-TOP et le flux optique) se sont intéressées aux textures des images et ont fourni une base à l'analyse des caractéristiques du visage, mais d'autres part celles -ci ont laissé progressivement la place aux techniques d'apprentissage profond pour la détection automatique de descripteurs faciaux.

Prédiction de l'émotion : Une fois l'ensemble de données transposées à un nouvel espace de caractéristiques, l'étape finale consiste à obtenir les prédictions associées à un modèle spécifique de représentation de l'émotion. Les méthodes usuelles d'apprentissage machine sont entre autres la classification et la régression pour le mode supervisé lorsqu'il s'agit de faire correspondre un modèle de représentation avec les caractéristiques extraites mais aussi le « clustering » dans un mode non supervisé où le modèle de représentation vient de la distribution même des vecteurs caractéristiques.

Naturellement, le modèle en catégorie d'émotions est associé à la classification alors que la détection de valeurs continue de modèles dimensionnels (e.g valence et excitation) se fera par régression.

1.5 Pré-traitement des données

1.5.1 Algorithmes d'alignement des visages

Afin de ne garder que l'information essentielle à la reconnaissance d'expression faciale, il est nécessaire d'isoler dans un premier temps les visages des sujets des séquences vidéos. Pour ce faire des algorithmes comme Viola-Jones (Viola & Jones, 2001) ou encore Mixture of Trees (MoT) (Zhu & Ramanan, 2012) ont été massivement utilisés, mais plus récemment les méthodes d'apprentissage profond se sont révélées plus performantes comme avec Tasks-Constrained Deep Convolutional Network (TCDCN) (Zhang *et al.*, 2014b) et Multi-task CNN (MTCNN) (Zhang *et al.*, 2016).

MTCNN pour la détection des visages : Cette architecture de réseaux de neurones combine trois réseaux à convolutions (P-Net, R-Net et O-Net) connectés en cascade afin de sélectionner séquentiellement les meilleurs fenêtres candidates à la présence de visages dans l'image. La Figure 1.5 développe la constitution de chacun de ces blocs de convolution. On y retrouve les trois réseaux P-Net, R-Net, O-Net, opérant des séries de convolutions aux images afin de produire par régression la hiérarchisation des visages, les valeurs des quatre coins du cadre contenant le visage potentiellement détecté, ainsi que les cinq points d'intérêt du visage (facial landmarks). Chaque CNN correspond à un étage pour lequel un ensemble de cadre vont être sélectionnés selon leur probabilité de contenir un visage. Le dernier étage va produire le meilleur cadre contenant l'intégralité du visage de l'image ainsi que les cinq coordonnées des principaux points d'intérêt du visage (ou facial landmarks) : deux pour les yeux, deux pour la bouche et un pour le nez.

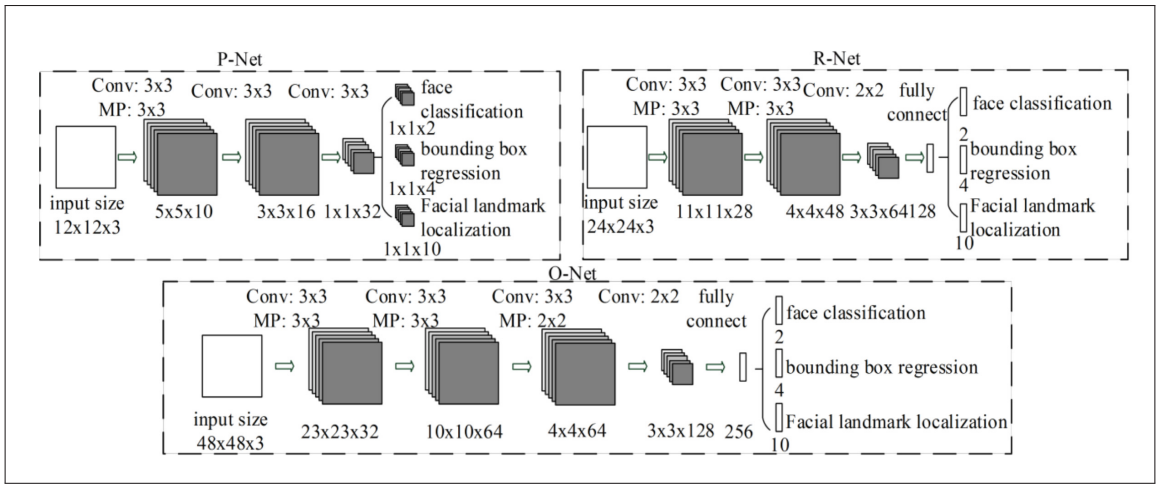

Figure 1.5 Représentation architecturale du réseau en cascade MTCNN
Tirée de Zhang *et al.* (2016)

Dans chacun des blocs de convolutions, en mode supervisé, sur chacun des échantillons x_i candidats pour l'isolation d'un visage, l'objectif est :

- de détecter la probabilité de présence d'un visage suivant une classification à deux classes (présence de visage ou non). Ceci est opéré par une fonction de coût de type entropie croisée (cross entropy) :

$$L_i^{\text{det}} = -\left(y_i^{\text{det}} \log\ (p_i)\ + (1 - y_i^{\text{det}}) \log\ (1 - p_i)\ )\right) \tag{1.1}$$

  où L_i^{det} est la fonction de coût à minimiser, y_i^{det} est le label de l'échantillon pour la présence du visage (typiquement 0 ou 1) et p_i est la probabilité de présence d'un visage produite par le réseau.

- d'opérer une régression pour le calcul des coordonnées du cadre du visage. On cherche à minimiser la distance entre le cadre idéal à trouver et les coordonnées produites par le réseau. La fonction de coût utilisée est une distance Euclidienne.

$$L_i^{\text{det}} = \|\hat{y}_i^{box} - y_i^{box}\|_2^2 \tag{1.2}$$

L_i^{det} étant la fonction de coût à minimiser, $\widehat{y}_i^{box}$ le label de l'échantillon donnant les coordonnées du cadre à trouver, y_i^{box} la prédiction du réseau.

- d'opérer une régression (similaire à la précédente) pour la découverte des cinq points particuliers du visage.

$$L_i^{landmark} = \|\widehat{y}_i^{landmark} - y_i^{landmark}\|_2^2 \tag{1.3}$$

$L_i^{landmark}$ étant la fonction de coût à minimiser, $\widehat{y}_i^{landmark}$ le label de l'échantillon donnant les coordonnées des points particuliers à trouver, $y_i^{landmark}$ la prédiction du réseau. Enfin la Figure 1.6 résume l'ensemble des étapes développées précédemment avec une image de démonstration.

1.5.2 Normalisation

Les principaux problèmes posés par les données récoltées dans des conditions naturelles (non contraintes) sont les variations d'illuminations et de poses de la tête qui ne sont pas pertinentes pour la détection de l'émotion et au contraire la complexifie. Pour ce faire, il est parfois nécessaire d'appliquer des techniques de normalisation. Shin *et al.* (2016) ont utilisé plusieurs techniques telles que la diffusion anisotrope, la transformation en cosinus discrète, et la différence de Gaussiennes. La différence de Gaussiennes consiste a soustraire deux versions floutées de la même image par application d'un filtre Gaussien. Ceci revient à appliquer un filtre passe-bande sur l'image pour ne garder qu'un certain domaine de fréquences spatiales. La différence de Gaussienne est aussi utilisée pour améliorer la visibilité des contours et ainsi la netteté de l'image. Cette dernière technique complète particulièrement bien la diffusion anisotrope qui a pour effet d'homogénéiser la température de l'image comme la diffusion thermique en physique et qui a tendance à flouter les contours des objets. La transformation en cosinus discrète quant à elle a plutôt tendance à concentrer les basses fréquences de l'image et à éliminer ainsi

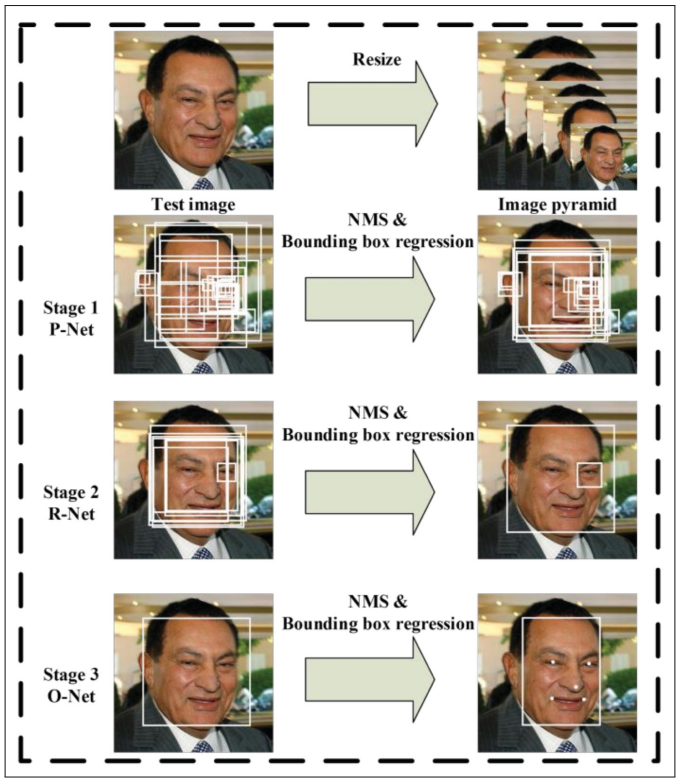

Figure 1.6 Représentation du pipeline de MTCNN avec une image test
Tirée de Zhang *et al.* (2016)

les zones de l'image présentant une trop forte illumination. Augmenter le contraste de l'image peut aussi s'avérer important afin de mieux différencier le fond du visage, pour ce faire les techniques de normalisation peuvent être associées à l'égalisation d'histogramme (Pitaloka *et al.*, 2017).

Enfin, en plus de l'illumination, la pose de la tête pouvant variée suivant la position de la prise de vue, peut représenter un problème du moment que certaines zones du visage ne sont pas directement visibles. Idéalement le visage doit être vu de face afin d'avoir un maximum d'information en une seule prise de vue. Initialement, la frontalisation du visage se fait en projetant les points particuliers dans un nouvel espace (Sagonas *et al.*, 2015). Plus récemment, l'utilisation de l'apprentissage machine pour générer de nouvelles images promet de meilleures performances, comme avec les architectures de type Generative Adversarial Networks (GAN) (Sagonas *et al.*, 2015), (Huang *et al.*, 2017), (L. Tran & Liu, 2017). Les Réseaux Antagonistes

Génératifs (GAN) sont des générateurs d'images où deux réseaux de neurones sont mis en compétition dans un mode non supervisé. Un des réseaux (générateur) aura pour tâche de créer de nouvelles images, quand l'autre (discriminateur) devra déterminer si l'image de sortie est réel ou le produit du modèle.

1.5.3 Augmentation de données

Dans le cas de l'apprentissage profond, un grand nombre de données est nécessaire pour une bonne capacité de généralisation des modèles et éviter le sur-apprentissage. En pratique, la récolte de données suffisantes pour entraîner des modèles d'apprentissage automatique est complexe. Un moyen simple pour pallier à ce problème est d'opérer l'augmentation de données, i.e des modifications sur l'image selon un ensemble de paramètres (e.g rotations, translations, augmentation du contraste, effet miroir, etc…).

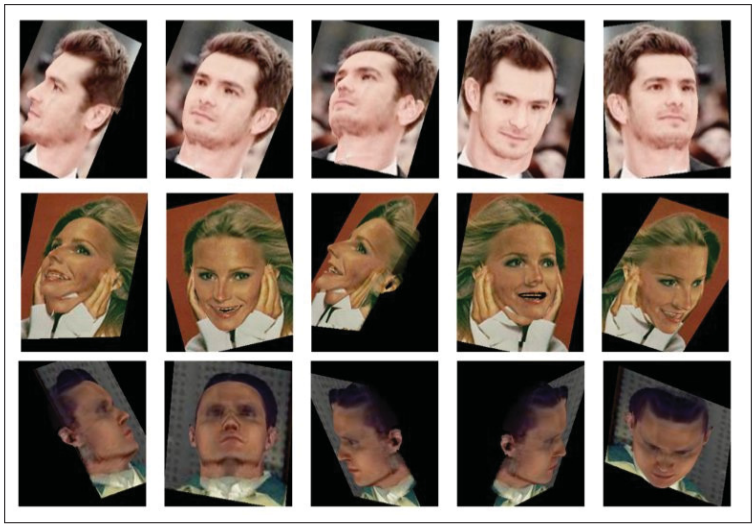

Figure 1.7 Exemple d'augmentation de données
Tirée de Crispell *et al.* (2017)

1.6 Méthodes d'apprentissage machine

Dans cette section, nous examinons les principales approches d'apprentissage machine pour la détection des émotions. Dans un premier temps, nous développerons le besoin d'extraire des

caractéristiques du visage afin de pourvoir les discriminer selon le modèle de représentation de l'émotion choisi, ensuite nous décrirons les méthodes d'apprentissage classique avec des machines à vecteurs de support (SVM) puis détaillerons l'avantage des réseaux de neurones pour l'extraction de caractéristiques discriminantes et la prédiction de l'émotion, formant ainsi la base de l'apprentissage profond.

1.6.1 Généralités

Traditionnellement les systèmes automatiques REF sont conçus de manière à apprendre de données annotées. Cette méthode particulière pour développer des algorithmes est appelée dans le domaine de l'apprentissage machine, mode supervisé. Cependant depuis 2005, et les années suivantes, les systèmes de reconnaissance d'émotions sont principalement reliés à deux facteurs clés : les bases de données et les algorithmes. Quand le premier enregistre l'information visuelle de visages, l'autre vise à modéliser les données comme un ensemble logique de caractéristiques projeté dans des espaces multidimensionnel. À cette fin, les représentations nouvellement arrangées peuvent être traitées par un classifieur tel que le SVM (Cortes & Vapnik, 1995), ou des couches de neurones densément connectées pour prédire l'émotion finale.

Comme vu précédemment, les chercheurs ont initialement utilisé des algorithmes artisanaux pour créer des caractéristiques à partir de bases de données contrôlées en laboratoire comme CK+ (Lucey *et al.*, 2010), MMI (Pantic *et al.*, 2005; Valstar & Pantic, 2010), JAFFE (Lyons *et al.*, 1998), en se basant sur les catégories primaires d'émotions. Parmi toutes les méthodes utilisées pour fournir des descripteurs faciaux, LBP, LBP-TOP, et SIFT ont été massivement utilisés. Notamment Zhao & Pietikainen (2007a) et Wang *et al.* (2015) ont utilisés LBP-TOP comme descripteur facial pour compacter l'information spatio-temporelle dans un vecteur.

1.6.2 Machine à Vecteur de Support (SVM)

La SVM est une technique d'apprentissage machine en mode supervisé permettant de discriminer un ensemble de vecteurs caractéristiques multidimensionnels en plusieurs catégories.

L'objectif du SVM est de trouver l'hyperplan qui maximise la marge existante entre deux ou plusieurs points, de différentes classes. La Figure 1.8 illustre graphiquement la recherche d'un hyperplan optimal. À gauche, plusieurs hyperplan (ici une droite) sont testés, à droite l'hyperplan maximisant la marge selon les critères d'entraînement a été trouvée. [1]

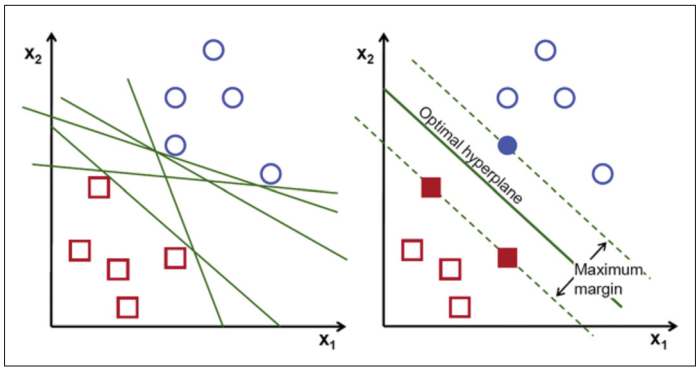

Figure 1.8 Exemple de phases d'entraînement d'un SVM à partir de vecteurs à deux dimensions

Ces vecteurs proches de la frontière théorique à découvrir sont appelés vecteurs support. Maximiser la marge d'erreur entre ces vecteurs permet d'améliorer le taux de confiance lors de l'inférence (phase de test où sont injectées de nouvelles données dans le système, différentes de celles utilisées pour l'entraînement). En mode supervisé, les classes et labels des données sont connus pour guider l'optimisation. Dans le cas du mode non supervisé, où les labels ne sont pas connus, SVM peut être appliqué dans le cadre d'un partitionnement des données (ou regroupement).

Ce principe est appliqué aussi bien pour des problèmes de classification que de régression. Pour la régression linéaire classique, on cherche à minimiser l'erreur entre les paires (x, y) d'entrée (source) et sortie (prédictions) en modélisant une fonction de transfert $y = f(x)$. Tandis qu'avec les régresseurs à vecteurs de support (SVR), le principe de la régression est appliqué selon un certain seuil de tolérance. On cherche à modéliser l'hyperplan permettant de minimiser l'erreur

1. https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

tout en maximisant une marge de tolérance. Par formalisation mathématique, on cherche à résoudre l'inéquation suivante :

$$Wx + b - \varepsilon < y < Wx + b + \varepsilon \tag{1.4}$$

où y est l'ensemble des prédictions, x les caractéristiques d'entrée, W la matrice de poids, b, le vecteur de biais, ε le seuil de tolérance.

1.6.3 Introduction aux réseaux de neurones

Avec les récentes avancées en matière de matériel informatique tels que les GPUs depuis 2012, les algorithmes d'apprentissage profond ont obtenu un intérêt croissant et sont extrêmement populaires dans le domaine de la vision par ordinateur. Mieux, l'apprentissage profond est même devenu le modèle d'excellence pour les systèmes REF et a permis de construire de robustes bases de références pour l'approfondissement de la recherche, et ce pour un large panel d'applications et tâches ayant des objectifs plus généraux tels que la classification, la segmentation et la détection. Les CNNs sont maintenant utilisés de manière conventionnelle pour résoudre des problèmes courants de vision par ordinateur lorsqu'il est nécessaire d'utiliser des images et vidéos en tant que source d'entrée. Les CNNs consistent en un agencement et fonctionnement particulier de réseaux de neurones convenant particulièrement à l'analyse d'éléments visuels au sein d'images. Afin de comprendre les mécanismes des CNNs, il convient de prime abord d'établir le fonctionnement des neurones artificiels.

A l'origine McCulloch et Pitts (Palm, 1986; Lettvin *et al.*, 1959) ont publié en 1950 l'article « What the frog's eye tells the frog's brain » modélisant un neurone formel à partir de l'analogie avec un neurone biologique (illustré par la Figure 1.9). Le neurone artificiel constitue alors une unité de mémoire définie par une matrice de poids W et un biais b. Le neurone agit comme une fonction de transfert, qui opère une série de transformation sur ses entrées et active ou non sa sortie suivant une valeur seuil définie par une fonction d'activation. L'agencement de

plusieurs neurones en réseaux selon différentes topologies de connexions permet ainsi de créer des modèles capables de modifier de manière autonome leur architecture (i.e leur paramètres de poids et biais) selon certaines règles d'apprentissage. L'objectif de l'apprentissage de réseaux de neurones est de minimiser l'erreur sur les prédictions du système à mesure que des données lui sont présentées.

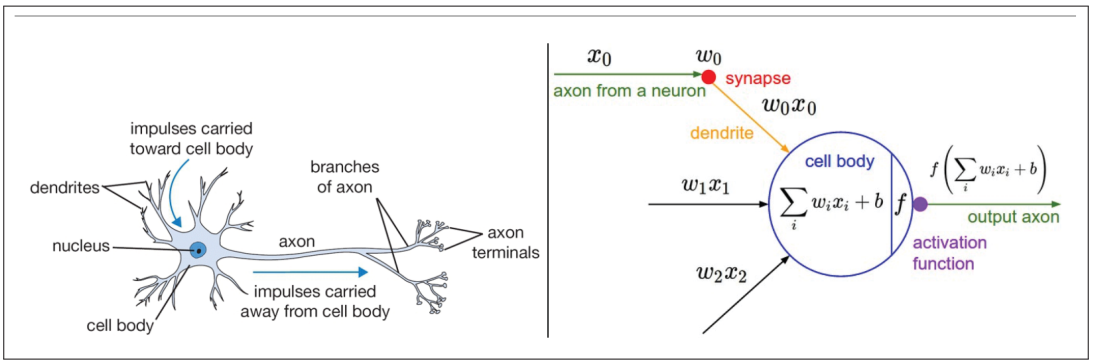

Figure 1.9 Illustration de l'analogie faite entre neurone biologique et neurone artificielle
Tirée de Karpathy (2015) [2]

Les topologies classiques de réseaux de neurones consistent à empiler des couches de neurones densément connectées à leurs entrées et sorties. L'apprentissage est alors opéré en rétro-propageant l'erreur calculée par une fonction de coût (e.g l'erreur quadratique moyenne ou l'entropie croisée), les poids et biais sont alors mis à jour par une méthode d'optimisation (descente stochastique du gradient de l'erreur, « Adamax », « Adam »). La capacité de généralisation de ces systèmes réside dans le potentiel statistique des données par lesquelles ils pourront extraire l'expérience nécessaire lors du processus d'apprentissage.

1.6.4 Réseaux de Neurones à Convolutions (CNN)

Les réseaux de neurones classiques de type Perceptron à couches multiples sont massivement interconnectés entre eux et rend le processus d'apprentissage long et lourd en calcul lorsqu'il convient de l'appliquer à des images. C'est ainsi qu'interviennent les architectures de type CNN. Non seulement les CNNs permettent d'alléger le calcul grâce au principe d'opérations en

2. https://cs231n.github.io/neural-networks-1/

convolutions mais permettent de détecter des formes particulières de l'image, insensiblement à la translation ou la rotation, ce qui les rend particulièrement robuste.

Le principe de base du CNN, similaire à la vision oculaire humaine, est d'apprendre un certain nombre de filtres sur l'image d'entrée. La dimension du noyau du filtre est ce qu'on appelle le champ réceptif de la couche de convolution. Techniquement, chaque filtre va opérer le même ensemble de paramètres aux pixels de l'image afin de transformer le volume d'entrée (i.e une image) en un volume de sortie réduit et dont l'information a été compactée. Le nombre de paramètres à modifier étant réduit, ces architectures sont allégées en calcul. De plus, à la différence de réseaux de neurones densément connectés, où chacun des pixels d'entrée est « vu » par l'ensemble des neurones d'une couche, la connectivité locale de chacun des neurones à une certaine région de l'image permet de relever certaines formes et objets contenus dans l'image.

Un réseau à convolution va typiquement être composé de plusieurs couches de convolution et de sous-échantillonnage (« pooling ») permettant de réduire les dimensions du volume d'entrée jusqu'à obtenir une prédiction finale quant au problème d'apprentissage machine. La Figure 1.10 illustre les différentes sorties d'une image filtrée traversant une succession de plusieurs couches de convolutions, activation « ReLu » et « pooling » avant la classification de l'objet suivant différentes catégories (voiture, camion, avion, …). Les activations « ReLu » permettent d'apporter une non-linéarité aux sorties de chacun des neurones. Quand la couche de « pooling », elle, réduit la taille du volume de sortie en calculant le maximum ou la moyenne de la valeur des pixels sur une certaine région. Pour un problème de classification, des couches densément connectées placées en fin d'architecture permettent de donner la probabilité d'appartenance à chacune des catégories.

L'architecture de la Figure 1.11 résume de manière simplifiée la conception d'un CNN pour une tâche de classification. Une image RGB de dimension 224×224 est insérée en entrée. S'en suit une série de couches de convolution et « max pooling » (blocs noirs et rouges) chacune traitant séquentiellement les sorties des couches précédentes jusqu'à classification par des couches

3. http://cs231n.github.io/neural-networks-1/
4. http://vision.gel.ulaval.ca/$\sim$cgagne/enseignement/apprentissage/A2018/presentations/iam-sem11-reseauconv.pdf

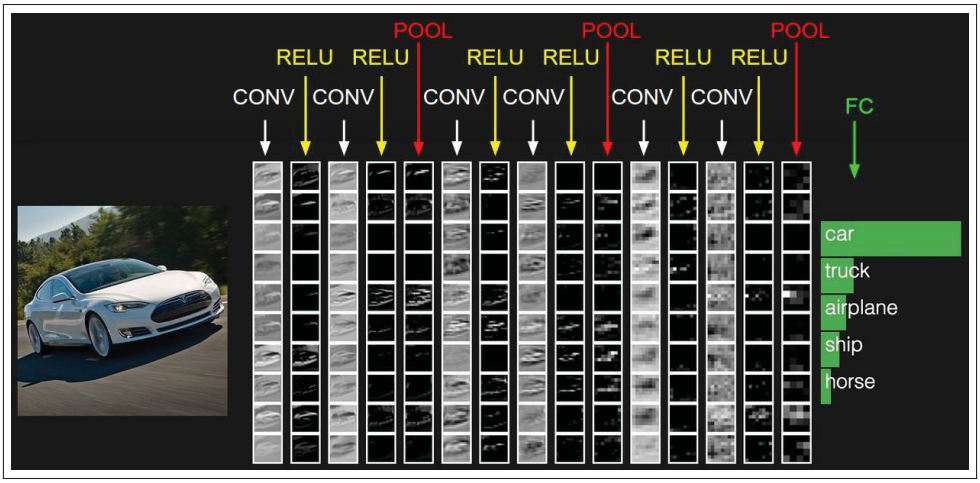

Figure 1.10 Illustration d'une succession d'opérations à convolution pour la prédiction d'objet
Tirée de Karpathy (2015) [3]

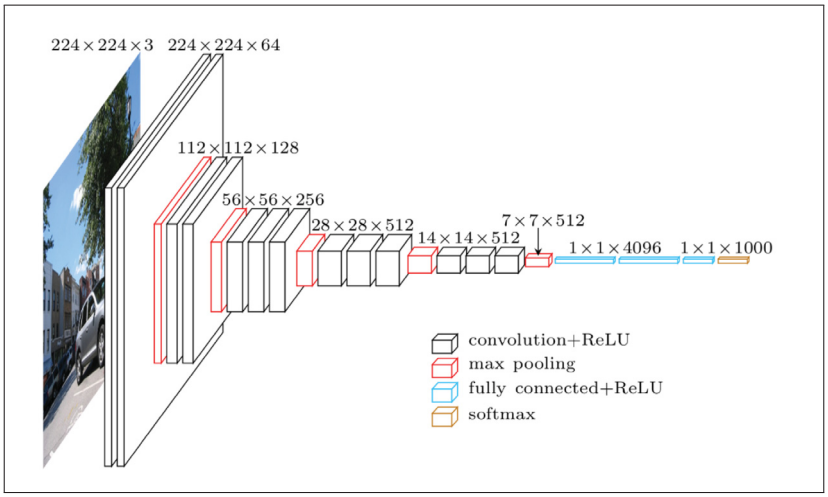

Figure 1.11 Illustration d'un réseau de neurones à convolution type VGG-16 [4]

de neurones densément connectés (bleu) pour la classification de l'objet contenu dans l'image. Les couches « ReLU » sont utilisées comme un type de non linéarité pour l'activation des neurones artificiels. Communément la fonction $f(x) = \max(0, x)$. La couche softmax de couleur or correspond à une couche densément connectée qui a pour fonction d'activation softmax. Chacun des neurones de la couche attribuera une probabilité d'appartenance à chacune des classes d'objets (ici 1 000 classes possibles). Par exemple Liu *et al.* (2018a) ont utilisé des modèles multi-modaux basés sur CNNs avec « late fusion » et une classification finale avec SVM pour

la prédiction d'expressions en catégories. L'idée était de combiner plusieurs CNNs basés soit sur des architectures de type Inception, DenseNet ou VGG-16, en prenant à la fois les images et les données audio comme source d'entrée. De même, Liu *et al.* (2015a), ont extrait le flux optique des images et ont alimenté un CNN-2D pour classifier les micro-expressions.

1.6.5 Architecture de type VGG

L'architecture de type VGG a été introduit par Simonyan & Zisserman (2014b) et qui a connu beaucoup de succès lors de son application sur ImageNet (Deng *et al.*, 2009) (classification d'objet de 1,000 classes) pour l'ISLVRC-2014 [5]. Ce modèle a montré des performances satisfaisantes pour de nombreuses applications. Typiquement, cette architecture est divisée en cinq blocs de convolutions alternés par des couches de « max pooling » et aboutit par un bloc de classification composé de couches densément connectées. Plusieurs modèles VGG sont apparus, variant dans leur nombre de couches de convolutions, ainsi on dénombre six modèles classiques de VGG à 11, 13, 16 et 19 couches. Le détail de chacune des architectures VGG sont illustrées par la Figure 1.12. La plus commune étant celle à 16 couches (VGG-16),illustrée par la Figure 1.13. On y retrouve de gauche à droite : la couche d'entrée en bleu, les couches convolutionnelles associées au nombre de filtres en orange, des couches de « max pooling » en jaune et les couches densément connectées en vert. Une activation softmax est usuellement utilisée en bout d'architecture pour des problèmes de classification.

1.6.6 Réseaux Multimodaux

Comme montré par ces dernières études (Liu *et al.*, 2018a, 2015a), diversifier les sources possibles d'information de l'émotion peut être bénéfique pour la qualité de la prédiction du système. On a vu précédemment que des sources couramment utilisées dans notre application sont les expressions faciales (l'image) et la voix des sujets ou le bruit contextuel (les données au-

5. http ://www.image-net.org/challenges/LSVRC/2014/
7. https://sefiks.com/2018/08/06/deep-face-recognition-with-keras/

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 1.12 Différentes configurations de modèle VGG à 11, 13 ,16 ou 19 couches paramétrables [6]

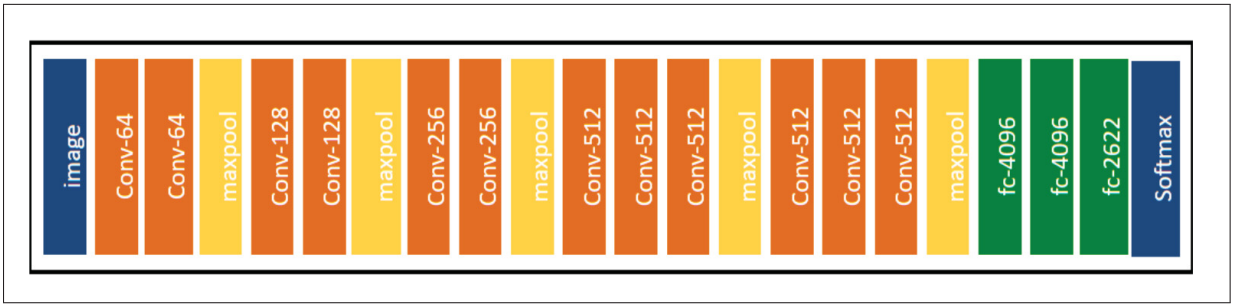

Figure 1.13 Architecture de type VGG-16 [7]

dio). De ce fait, la combinaison de différentes sources peut s'avérer complémentaire et faire gagner les systèmes en précision.

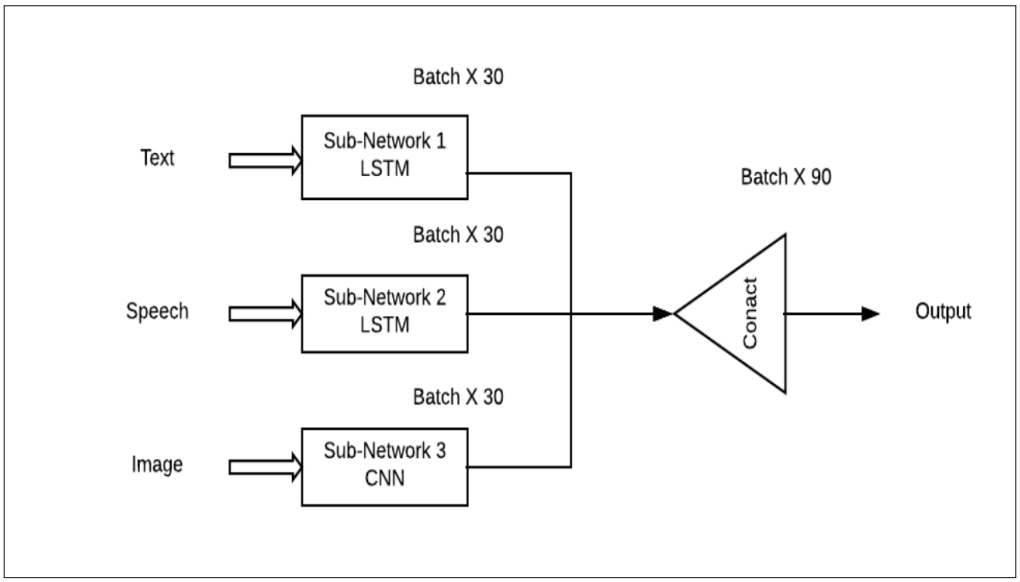

Figure 1.14 Exemple d'architecture multi-modale [8]

Comme montré dans la Figure 1.14, trois sources d'information (texte, voie, et image sont injectés respectivement dans trois réseaux différents, leurs sorties sont enfin concaténées et traitées par un module pour la prédiction finale. En général, la contribution de chacune des sources peut varier, et un réseau en bout d'architecture, avant la prédiction, a pour objectif de balancer la contribution des différentes modalités au cours de l'apprentissage.

1.7 Méthodes de Description Spatio-Temporelle de l'information

Après avoir introduit les méthodes artisanales et d'apprentissage machine permettant la description d'éléments texturaux des images mais aussi la prédiction finale de l'émotion, on souhaite décrire plus en avant les techniques de description de l'information temporelle des données. On citera dans un premier temps l'ensemble de ces méthodes décrites par la littérature puis en ferons une analyse plus détaillée.

8. https://towardsdatascience.com/multimodal-deep-learning-ce7d1d994f4

1.7.1 Généralités

En plus de la transition opérée par le domaine entre des caractéristiques extraites artisanalement à des composantes plus profondes apprises par la machine, les systèmes REF s'appliquent de plus en plus sur des séquences d'images dynamiques plutôt que sur des images statiques (Li & Deng, 2018). En d'autres termes, les précédentes études ont premièrement exploité des bases de données dont les sujets étaient associés à une unique catégorie d'émotion pour une seule image de visage. Les résultats faisant office d'état de l'art pour ces bases de données ont notamment utilisés des CNNs (e.g Kim *et al.* (2015); Liu *et al.* (2017); Zhang *et al.* (2015); Guo *et al.* (2016); Kim *et al.* (2016); Pramerdorfer & Kampel (2016). Cependant il convient d'adopter une tout autre approche pour l'analyse de séquences d'images. Une approche classique est d'agréger les caractéristiques de chacune des trames en un vecteur compact sur l'ensemble d'un mini-clip avant la prédiction de l'émotion finale, comme décrit par Ding *et al.* (2016b). En plus de l'agrégation de caractéristiques, Bargal *et al.* (2016) ont concaténé la moyenne, la variance, et les minimum et maximum sur le vecteur caractéristique de la séquence vidéo de manière à apporter de l'information statistique complémentaire.

Cependant du moment que l'agrégation de caractéristiques ne peut exploiter l'inter-corrélation existant entre chacune des images d'une séquence et ne peut expliquer la dépendance temporelle entre celles-ci, cette approche reste fortement limitée. Pour répondre à ce problème, les réseaux récurrents (RNNs, recurrent neural networks) et les architectures 3D (e.g 3D-CNN) peuvent intégrées en entrée une série de données et extraire l'information temporelle par un système de mémoire qui se traduit soit séquentiellement par une forme de récurrence, soit spatialement par extension de ses champs récepteurs au niveau de l'architecture du système. Nous verrons par la suite comment ces architectures opèrent ces principes. Néanmoins, il est important de s'assurer pour ces types d'architecture que les données d'entrées soient ordonnées séquentiellement et que les transitions entre chaque étape aient un potentiel considérable d'information. Alors que les modèles améliorés de RNNs, les Long Short Term Memory (LSTM) peuvent traiter des données séquentielles de taille variable dans les deux directions temporelles,

les 3D-CNN exploitent les variations de textures de séquences d'images en étendant leur noyau de convolutions à la troisième dimension.

Voilà pourquoi les 3D-CNNs sont tout à fait appropriés pour les applications de vision par ordinateur. Les premières architectures 3D-CNNs ont été évaluée par Tran *et al.* (2014) pour des tâches de reconnaissance d'actions notamment sur la base de donnée UCF101, un ensemble de vidéos hiérarchisées en 101 catégories d'actions. Tran *et al.* (2014) ont finalement démontré que le 3D-CNN surpassait les modèles 2D-CNNs sur des applications similaires d'analyse vidéo, et ainsi pouvaient apporter des vecteurs caractéristiques plus compact et efficace dans la description visuelle d'images et vidéos. Le transfert de 3D-CNNs à des applications REF a pu s'opérer et récemment plusieurs études se sont basées sur ce type d'architecture (Abbasnejad *et al.*, 2017; Fan *et al.*, 2016; Nguyen *et al.*, 2017; Liu *et al.*, 2018b; Barros & Wermter, 2016; Zhao *et al.*, 2018; Ouyang *et al.*, 2017).

1.7.2 Étude séquentielle de trames par fusion des caractéristiques

On a vu qu'il était possible de fusionner plusieurs sources différentes afin de compléter l'information émotionnelle. Cependant à la différence de cette fusion de type modale, lorsqu'il s'agit d'analyser un ensemble de séquences vidéos, l'information temporelle disponible est réparti au sein de plusieurs trames contigues. Pour ce faire plusieurs catégories de fusion temporelle sont possibles afin de caractériser la connectivité entre ces trames. Karpathy *et al.* (2014) ont caractérisé trois types de fusion : « late fusion », « early fusion » et « slow fusion ». Comme décrit par la Figure 1.15 différentes méthodes s'opposent pour l'obtention de l'information temporelle par l'aggrégation de l'information spatiale. «Late fusion» extrait les caractéristiques de plusieurs trames à partir de plusieurs séquences de convolutions avant de les concaténer pour la classification ou la régression, ceci implique l'utilisation de différents réseaux à convolutions. «Early fusion» aggrège les trames pour les insérer dans un unique réseau à convolution avant classification ou régression. «Slow fusion» résulte de la combinaison des deux différentes techniques en intégrant à la fois la fusion des trames en entrée et la fusion successive des caractéristiques extraites à différents niveaux de convolutions.

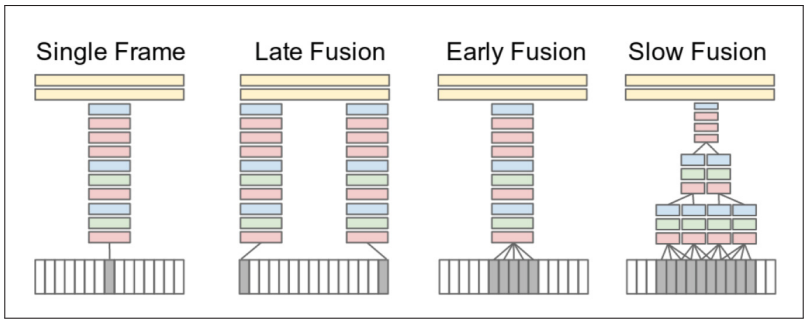

Figure 1.15 Différents types de fusion pour l'analyse séquentielle de trames
Tirée de Karpathy *et al.* (2014)

En gris sont indiqués les trames sélectionnées pour la fusion, en rouge, vert et bleu sont représentés les couches de convolution, normalisation et pooling. Enfin, en jaune, on retrouve, les couches densément connectées pour la prédiction finale.

1.7.3 Réseaux de Neurones Récurrents (RNN)

Les Réseaux Récurrents (ou Recurrent Neural Networks (RNNs)) permettent de récupérer l'information temporelle ou séquentielle d'une série de données. L'architecture type du RNN consiste en un assemblage d'unité de mémoire basée sur une typologie particulière de réseaux de neurones. Le principe de récurrence, appliqué à ce type de modèle, implique qu'à partir d'une entrée x prise à un instant t, et en lui injectant la sortie précédente y à l'instant t-1, on soit capable de prédire la sortie y à l'instant t. Ceci défini le fonctionnement d'une simple unité de récurrence ou plus communément appelée cellule. La Figure 1.16 montre le schéma architectural d'une cellule RNN. Celle-ci représente schématiquement et de manière différentes, la même cellule RNN. Une entrée X_t est introduite dans la cellule pour produire une sortie Y_t, dans cet exemple, un vecteur à trois nombres est transformé en un vecteur à quatre nombres. La cellule est composée de trois réseaux de neurones (représenté par des matrices de poids (U, V, W) : V et U sont composés de cinq neurones (A1-A5 et B1-B5) et W de quatre neurones (C1-C4). La cellule retient à chaque instant t un état représenté par la fonction h_t pour le réintroduire à l'instant suivant t+1. Ainsi le principe de mémoire est appliqué en injectant à l'entrée un état calculé à l'itération précédente. De cette manière l'information temporelle est transmise aux éléments successifs d'une séquence. On obtient ainsi, une vue d'ensemble

sur tous les éléments passés à un instant t. Enfin, la cellule est composée de deux fonctions d'activations g_h pour l'état intermédiaire et g_y correspondant à l'activation de la sortie. Par formulation mathématique, la cellule RNN est définie par deux équations. L'état intermédiaire (ou état caché) h_t et la prédiction de la cellule Y_t sont définis par :

$$h_t = g_h(Vx_t + Uh_{t-1} + b') \tag{1.5}$$

$$\widehat{y}_t = g_y(Wg_h h_t) + b \tag{1.6}$$

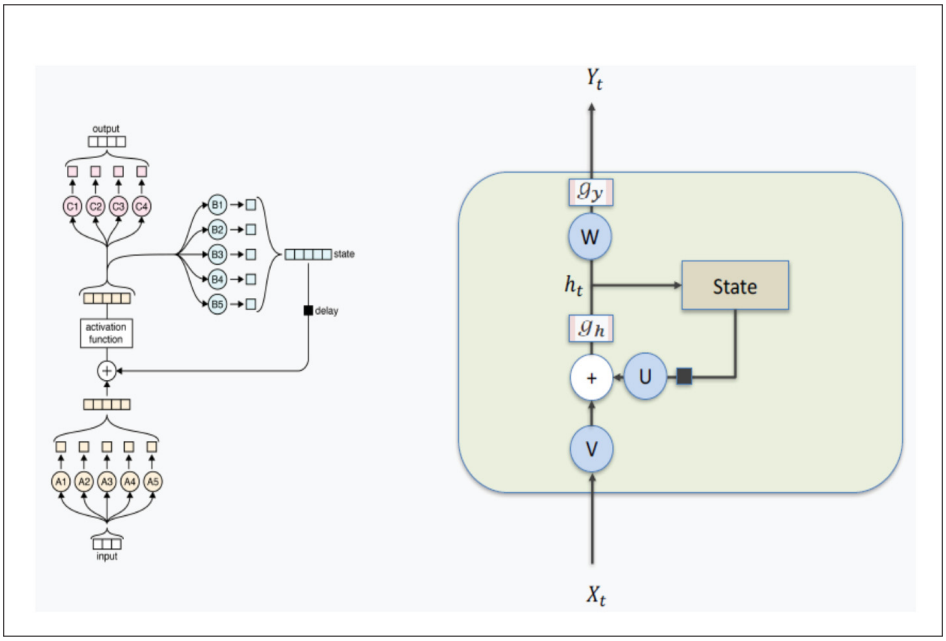

Figure 1.16 Représentation d'une cellule RNN [9]

Long-Short Term Memory (LSTMs)

L'un des principaux problèmes des RNNs est lors de l'apprentissage de produire des gradients de plus en plus faibles à mesure que la longueur des séquences s'accroît. C'est ce qu'on ap-

9. https://towardsdatascience.com/recurrent-neural-networks-explained-ffb9f94c5e09, P.Protopapas, CS109b, Harvard FAS

pelle l'évanouissement du gradient. Les paramètres des cellules les plus profondes sont plus lentement modifiées que ceux en bout d'architecture car la modification des paramètres est proportionnelle aux gradients de la fonction de coût d'après le principe de rétropropagation de l'erreur. Les LSTMs (Hochreiter & Schmidhuber, 1997) sont une amélioration du RNN permettant, grâce à l'introduction d'un système de porte, de sélectionner l'information à conserver, et à oublier lorsque l'information à transmettre est particulièrement longue. On note la présence de trois portes constituant la cellule LSTM : Entrée (Input Gate), Sortie (Output Gate), Sélection de l'information (Forget Gate).

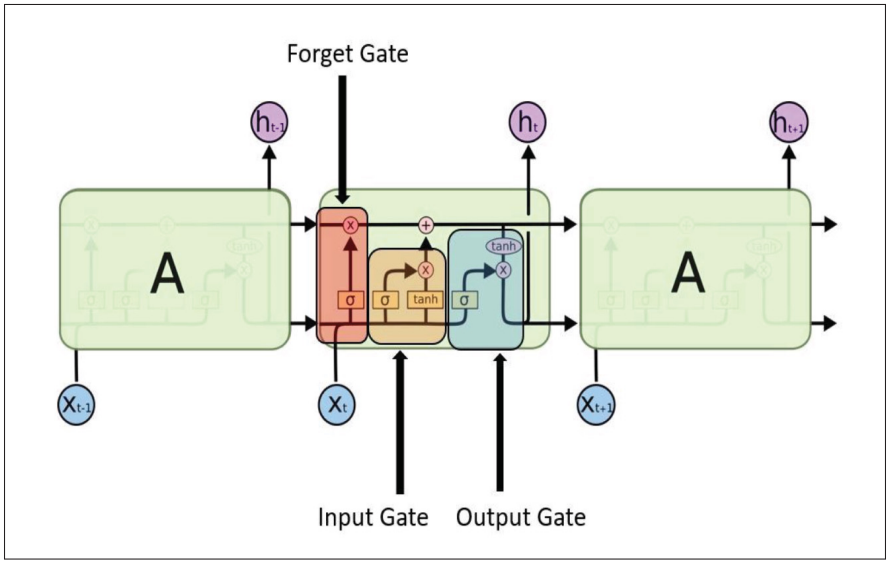

Figure 1.17 Représentation d'une cellule LSTM
Tirée de Mittal (2019) [10]

1.7.4 Réseaux en Cascade

Une approche populaire pour la description spatio-temporelle de l'information est le réseau en cascade. Les architectures sont empilées les unes sur les autres formant ainsi une succession de blocs, permettant d'extraire successivement plusieurs types de caractéristiques, jusqu'à la prédiction finale. En particulier, combiner les architectures, CNN et LSTM a montré une efficacité satisfaisante pour obtenir des représentations spatio-temporelles d'images.

10. https://towardsdatascience.com/understanding-rnn-and-lstm-f7cdf6dfc14e

Ouyang *et al.* (2017) ont par exemple utilisé une architecture CNN basée sur VGG-16 pour extraire les caractéristiques de séquences de 16 images puis alimenté un LSTM pour prédire une catégorie d'émotions parmi six. Les images ont été prétraité par MTCNN (Multi-Task Cascaded Networks) afin de détecter les visages puis chacune des vidéos sont découpés en fenêtre de 16 images. De manière similaire, Vielzeuf *et al.* (2017), ont utilisé des architectures VGG-16-LSTM comme sous partie d'un ensemble de réseau comprenant également un 3D-CNN-LSTM et un réseau audio. En particulier, les chercheurs ont utilisé la méthode MIL (Multi-Instance Learning ou Apprentissage par Instance Multiple), qui consiste à créer un ensemble de mini-clips pour chacune des séquences vidéo avec un certain taux de recouvrement. Chaque mini-clip (fenêtre) est décrit par un unique label et contribue à la prédiction du label de la vidéo dans sa globalité.

1.7.5 3D-CNN / i3D

De manière similaire au 2D-CNN développé précédemment, qui opère des convolutions sur des images en deux dimensions. Le 3D-CNN applique les convolutions en étendant son champ réceptif à une 3ème dimension (la dimension temporelle). C'est-à-dire que le noyau des filtres de chacune des couches de convolutions sont à trois dimensions. Ainsi, ce type de réseau est capable de traiter en entrée des données spatio-temporelles comme des séquences vidéos en alignant leurs images constitutives selon un 3ème axe. Les 3D-CNNs sont alors capables d'apprendre à la fois des textures des images et de leur inter-corrélation temporelle. Le revers de cette architecture est de s'avérer complexe quant au nombre de paramètres à modifier lors de l'apprentissage et de se montrer exigent quant à la quantité et distributivité des données afin d'obtenir une bonne capacité de généralisation.

La Figure 1.18 confronte les architectures CNN-RNN au 3D-CNN. Avec CNN-RNN, on extrait dans un premier temps les caractéristiques de chacune des trames avec CNN puis la concaténation de l'ensemble est vu comme un unique vecteur d'entrée par le RNN qui établira la relation temporelle entre chaque trame pour la prédiction du label de la séquence. Avec 3D-CNN, les caractéristiques spatio-temporelles de chaque séquence sont extraites en une seule étape par

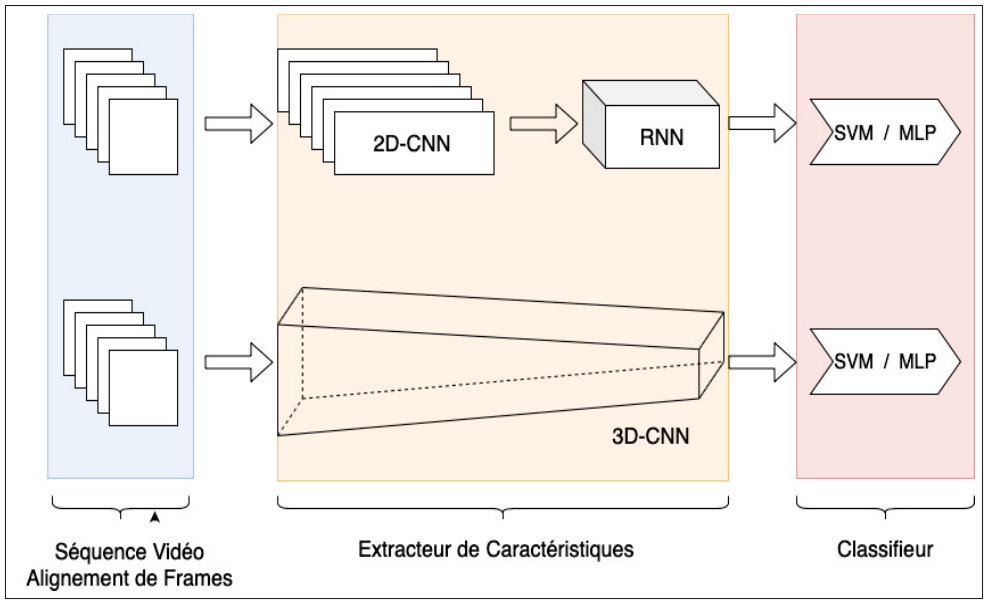

Figure 1.18 Comparaison schématique de l'analyse de séquences vidéo par CNN-RNN et 3D-CNN

le modèle, prenant directement en entrée la séquence d'image. Un vecteur caractéristique est ensuite produit afin d'établir la prédiction finale du label.

Initialement, de premiers modèles ont été développé pour des tâches de reconnaissance d'action (Ji *et al.*, 2013; Tran *et al.*, 2014; Liu *et al.*, 2018b). Des 3D-CNN pré-entraîné pour ces tâches ont ensuite été rendu disponible et transféré à la recherche en informatique affective (Fan *et al.*, 2016; Nguyen *et al.*, 2017). A notre connaissance aucun modèle 3D-CNN n'ont encore été utilisés pour des systèmes REF avec le modèle dimensionnel, et ont plutôt fait l'objet d'expérimentations pour le modèle en catégorie. Ceci est principalement dû actuellement à une meilleure disponibilité de données pour la classification d'émotions sur des images plutôt que pour la reconnaissance d'expressions au format vidéo et annotées sur une base de valeurs réelles continues exploitant adéquatement la dimension temporelle. En effet la reconnaissance d'expressions faciales sur des images et un pré-entraînement sur 2D-CNN sont actuellement encore un meilleur choix. Afin de pallier à ce problème, (Carreira & Zisserman, 2017) ont développé un modèle i3D («inflated 3D-CNN »), capable d'apprendre des représentations 3D basées sur des ensembles de données 2D (images). Ils ont pour cela utilisés un modèle 2D-

CNN de type Inception dont les poids ont été étendu à la troisième dimension. Autrement dit les noyaux de convolutions 2D appris à l'aide d'ensemble de données 2D, ont été élargi à des noyaux de convolutions 3D par copie des poids sur la troisième dimension. De cette manière, les chercheurs ont pu développer plusieurs réseaux pré-entraînés basé sur une même architecture de type Inception à l'aide des ensembles de données ImageNet (reconnaissance d'objets dans l'image) et Kinetics (reconnaissance d'actions pour la vidéo). Leur technique faisait également intervenir une combinaison (aggrégation) du flux optique avec les images en entrée des réseaux.

1.7.6 Réseaux de neurones à convolutions temporelles (TCN)

Une variante des réseaux CNNs a été proposée par Lea *et al.* (2016) ; les (TCN - Temporal Convolution Network). Jusqu'ici les réseaux récurrents, RNNs, et ses variantes LSTM et GRU, capables d'enregistrer l'information temporelle, étaient les méthodes les plus populaires afin d'extraire les caractéristiques et dépendances internes de longues séquences. Par ailleurs, on a vu que les modèles 3D-CNN étaient capables d'étendre les convolutions 2D à une 3ème dimension par extension de ses noyaux de convolutions afin de capter à la fois l'information spatiale et temporelle. Les séquences d'images 2D, pouvaient donc être analysées en opérant des convolutions 3D.

Cependant, une équipe de Google DeepMind (Van den Oord *et al.*, 2016) a eu l'idée d'utiliser les CNN comme modélisateur de séquences. En considérant que l'entrée soit une séquence, les convolutions d'une couche de neurones sont opérées sur le pas de temps courant et précédents en une seule étape, via la couche de convolutions précédente. A noter d'une part que DeepMind opérait dans ce cas de figure des convolutions 1D, soit des vecteurs à une dimension comme source d'entrée (et non des images 2D), et d'autre part chacune des couches de convolutions était insensible à la longueur de la séquence d'entrée et rendait en sortie une séquence dont la longueur était conservée.

À la différence d'un RNN qui ne peut calculer la prédiction courante que séquentiellement, à partir d'une version compactée des pas de temps précédents, le TCN, en une seule étape, est capable d'opérer des convolutions sur une fenêtre entière de la séquence. Par ailleurs, le principe du TCN a permis d'introduire le principe de dilution de convolutions, (Bai *et al.*, 2018). Cette méthode consiste en une connexion particulière des neurones des couches de convolutions permettant d'opérer des convolutions en étirant le champ réceptif de chacun des neurones de la couche. Avec un niveau de dilution approprié à chaque couche de convolutions, il est possible de transmettre l'information temporelle de la séquence au travers des différentes couches du CNN.

On représente via la Figure 1.19 le principe de dilution. Une séquence de longueur T est présentée comme source d'entrée à un CNN de trois couches de neurones (deux couches cachées et une couche de sortie). Les éléments en jaune et bleu représentent la séquence d'entrée ainsi que ces modifications successives par convolutions. Tandis que les lignes bleues sont les opérations de convolutions.

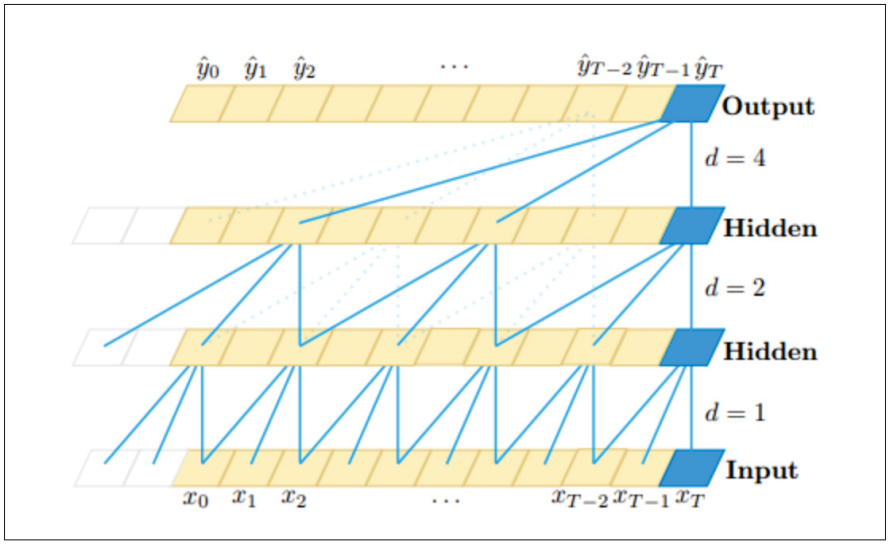

Figure 1.19 Représentation du principe de dilution de convolutions à l'aide d'un CNN à trois couches
Tirée de Roy (2019) [11]

11. https://medium.com/@raushan2807/temporal-convolutional-networks-bfea16e6d7d2

Afin de faire suivre l'information temporelle au travers du modèle, on attribue à chaque couche du réseau de neurones un niveau de dilution (ici d=1, d=2 et d=4). De cette manière on observe que le vecteur de la couche de sortie représentant l'instant T (en bleu sur la figure), résulte d'opérations de convolutions sur l'ensemble des éléments de la séquence d'entrée. On peut alors imaginer intégrer ce principe à un modèle de convolutions 3D, pour lequel la dilution serait opérer sur la dimension temporelle du noyau de convolution 3D. Il est alors possible de faire suivre l'information temporelle au travers des couches d'un 3D-CNN prenant en entrée des séquences d'images.

1.8 Transfert d'apprentissage

1.8.1 Transfert d'apprentissage : concept essentiel de l'apprentissage profond

Lorsque nous voulons apprendre une nouvelle tâche, le premier réflexe humain est d'utiliser ce que nous avons appris afin d'apprendre plus efficacement. Nous utilisons nos connaissances dans un domaine afin de l'appliquer à une tâche d'un domaine similaire. Plus ces deux domaines sont reliés et plus il est facile de maîtriser cette nouvelle tâche. L'apprentissage profond fonctionne en effet de la même manière. Cependant pour résoudre des tâches complexes et souvent particulièrement spécifiques, l'apprentissage profond nécessite une quantité de données conséquente et présentant une distribution suffisante pour se confronter à un grand nombre de cas possible, i.e être capable de généraliser. D'autant plus, qu'en mode supervisé les données doivent être annotées. Obtenir ce genre d'ensemble de données pour chacun des domaines associés à une tâche spécifique est bien souvent complexe.

Le principe de transfert d'apprentissage part du postulat que les connaissances acquises (i.e paramètres d'un modèle) pour une tâche particulière peuvent servir de meilleur point de départ pour un apprentissage sur une tâche plus ou moins similaire qu'une initialisation aléatoire des paramètres d'un nouveau modèle lors d'apprentissage machine traditionnel. La Figure 1.20 opposent schématiquement ces deux derniers concepts.

12. https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a

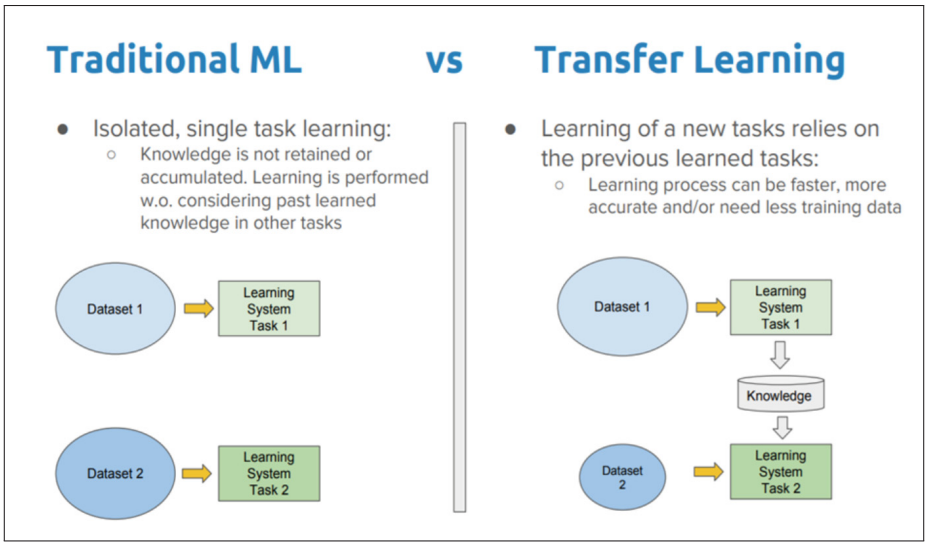

Figure 1.20 Apprentissage machine traditionnel vs transfert d'apprentissage
Tirée de Sarkar (2018) [12]

A gauche, l'apprentissage machine traditionnel associe un domaine particulier de données pour résoudre une tâche spécifique. Les ensembles de données doivent être de taille conséquente afin de garantir la robustesse des systèmes à de nouvelles données. A droite, l'apprentissage d'une tâche via un ensemble de données conséquent et un modèle particulier permettent l'apprentissage d'une nouvelle tâche plus ou moins similaire à la première à l'aide d'un nouvel ensemble de données. Le transfert d'apprentissage permet d'apprendre une nouvelle tâche avec moins de données disponibles qu'avec l'apprentissage machine traditionnel.

1.8.2 Formalisation mathématique du transfert d'apprentissage

L'apprentissage automatique peut être défini par deux composantes principales, un domaine D et une tâche T. Un domaine est caractérisé par un espace de caractéristique X et une distribution $P(X)$ où $X = \{x_1, \ldots, x_n\}$, $x_i \varepsilon\ X$. En résumé $D = \{X, P(X)\}$. Une tâche a pour objectif de faire correspondre l'espace de caractéristiques X avec un espace de valeurs cibles Y via une fonction objectif $P(Y|X)$ où $Y = \{y_1, \ldots, y_n\}$, $y_i \varepsilon\ Y$. Cette fonction devra être trouvée lors du processus d'apprentissage.

Le transfert d'apprentissage, quant à lui, fait intervenir un domaine source D_S associé à une tâche source T_S, ainsi qu'un domaine cible D_T associé à une tâche cible T_T, dont l'objectif sera d'apprendre une distribution $P(Y_T| X_T)$ à partir des connaissances apprises de D_S et T_S. En tenant compte que $D_S \neq D_T$ et $T_S \neq T_T$. Plusieurs stratégies de transfert d'apprentissage sont possibles suivant la disponibilité des valeurs cibles Y_S et Y_T des domaines D_S et D_T. (Pan & Yang, 2010a). On différencie trois types de transfert d'apprentissage, le transfert inductif, le transfert transductif et le transfert non-supervisé :

- **transfert inductif** : Les valeurs cibles de D_T sont disponibles.
- **transfert transductif** : Seules les valeurs cibles de D_S sont disponibles
- **transfert non-supervisé** : Aucune donnée cible n'est disponible. Il existe différentes techniques comme le regroupement ou la réduction de dimension pour opérer ce type de transfert d'apprentissage.

Dans le cas du transfert inductif d'apprentissage, il est possible de différencier deux cas suivant la disponibilité des données cible de D_S :

- si les données sont disponibles, en supposant que les domaines sources et cibles sont les mêmes, et leurs tâches associées similaires, il est possible de faire de l'apprentissage Multi-Tâche, i.e les deux tâches sont apprises simultanément.
- si les données ne sont pas disponibles, en supposant que les domaines sources et cibles sont les mêmes, et leurs tâches associées similaires, l'apprentissage est induit automatiquement.

Dans le cas du transfert transductif, on différencie deux autres cas suivant la relation entre les domaines source et cible :

- si les domaines sont différents, mais s'appliquent à la même tâche, il est possible d'opérer l'adaptation de domaine (Kouw & Loog, 2018)
- si les domaines sont fortement reliés et associés à une seule tâche, il est possible d'opérer ce qu'on appelle la sélection biaisée d'échantillons, les données sont réarrangées à partir de

certains postulats, ou le changement de covariance où la distribution de donnée est directement modifiée.

Généralement, pour le transfert transductif et inductif, les tâches classiquement appliquées ont pour but la classification et la régression.

1.8.3 VGG-Face pour le transfert d'apprentissage

Construire des architectures pour une tâche aussi spécialisée que la reconnaissance d'émotion représente un véritable challenge depuis que des données pour ce type d'application sont disponibles. De plus, entraîner des modèles intégralement, c'est à dire sans une initialisation adéquate des paramètres de l'architecture et donc avec une initialisation aléatoire, n'est pas envisageable si nous voulons extraire des caractéristiques sur des images avec peu de pré-traitement. Ainsi, pré-entraîner les réseaux de neurones est un atout essentiel pour ne pas conduire les architectures au sur-apprentissage. De ce fait, des modèles faisant actuellement office d'état de l'art ont été développé et partagés à des fins de recherche tel que VGG-Face (Parkhi *et al.*, 2015) ou inflated Inception3D (Carreira & Zisserman, 2017).

VGG-Face est basé sur une architecture de type VGG-16 développée par (Parkhi *et al.*, 2015). Ce modèle avait pour objectif de pallier le manque de données disponible en proposant une architecture pour l'identification et la vérification de visages ainsi qu'un ensemble de données conséquent pour la reconnaissance faciale. Ainsi les chercheurs ont collecté environ trois millions d'images à partir de 2 622 sujets différents, rendant ainsi cette architecture particulièrement adaptée à la fois pour des tâches de reconnaissance de visages et d'émotions.

De récents travaux utilisant VGG-Face, ont atteint de bonnes performances au sein de challenges internationaux REF tels que EmotiW ou AVEC en se plaçant en tête de ces compétitions. Ceux-ci ont prouvés une efficacité particulière à extraire des caractéristiques discriminantes du visage. Wan *et al.* (2017) ont combiné LDA et WPCA (méthodes de réduction de dimension des vecteurs caractéristiques) avec VGG-Face pour l'extraction de caractéristiques ainsi que des techniques de réduction de dimension pour des applications de reconnaissance faciale.

Knyazev *et al.* (2017) pour EmotiW challenge ont perfectionné (fine-tune) un modèle VGG-Face avec l'ensemble de donnée FER2013, puis ont agrégé les caractéristiques de trames pour la classification d'émotions avec SVM linéaire. Ouyang *et al.* (2017) ont combiné VGG-Face avec LSTM au sein d'un ensemble de réseau utilisant d'autres type de CNN-RNN et 3D-CNN. Ceci dans le but de fusionner plusieurs modalités d'émotions pour la prédiction de catégories d'émotions. Enfin, Ding *et al.* (2016a) ont proposé une technique particulière de perfectionnement de réseau (« fine-tuning ») avec VGG-Face. Ces derniers ont contraint leur propre réseau à « agir » comme VGG-Face en transférant la distribution des couches de neurones en sorties plutôt qu'en transférant ses poids. Deux CNNs (VGG-Face et un CNN initialisé aléatoirement) sont placés en parallèle puis connectés ensemble à un classifieur en sortie pour la prédiction finale de l'émotion. Ainsi, lors de l'apprentissage, la distribution du VGG-Face est transmise au CNN par rétro-propagation de l'erreur. Ce faisant, la distribution finale des poids de chacune des architectures étant plus similaire l'une à l'autre.

Sur-apprentissage

Durant la période d'entraînement de modèles, l'ensemble des paramètres du modèle sont modifiés par utilisation de données d'entraînement. Inversement, en mode de test afin d'observer les capacités de généralisation du modèle, ces paramètres sont fixés en observant les prédictions à partir de données non connues du modèle. Dans le cas où le modèle performe suffisamment sur les données d'entraînement mais pauvrement sur les données test, le modèle ne peut généraliser les paramètres appris à de nouvelles données. On dit alors que le modèle a sur-appris des données d'entraînement. Ceci survient généralement lorsque le modèle est trop complexe pour l'ensemble de données.

1.9 Apprentissage par instance multiples (MIL)

Les séquences vidéo analysées par les modèles d'apprentissage sont parfois suffisamment longues pour pouvoir être découpées en mini-clips. Souvent l'instant où l'information la plus discriminante à la détection d'émotions dans la vidéo est difficile à identifier, il est possible

de séparer la vidéo en plusieurs fenêtres de trames, analysées séparément. Ainsi, chaque vidéo peut être considérée comme un ensemble de fenêtres possédant toutes le même label (classification d'émotion). Comme proposé par Vielzeuf *et al.* (2017), après extraction des caractéristiques sur un batch de mini-clips avec 3D-CNN, un classifieur a alors pour tâche de détecter la fenêtre, produisant le score le plus élevé, pour la classification de l'émotion.

Comme nous allons le voir par la suite, notre méthode s'inspire fortement de cette technique mais appliqué à la régression. Dans le cas de modèles dimensionnels, des valeurs de valence et excitation sont appliquées à chacune des trames d'une vidéo, et non plus une seule catégorie d'émotion. Il est aussi parfois difficile en pratique d'appliquer un seul label à la vidéo du fait de sa longueur, plusieurs unités d'émotions apparaissent. Une solution est donc de diviser la vidéo en plusieurs fenêtres de trames avec un certain taux de recouvrement et de leur attribuer un unique label. Ainsi les mini-clips créés forment des échantillons à part entière et ne desservent plus la prédiction de la vidéo dans son ensemble.

1.10 Reconnaissance de l'émotion avec représentations dimensionnelles

Bien que le modèle en catégorie soit particulièrement exploité pour la reconnaissance d'émotions et ait prouvé de bonnes performances à la fois sur l'image et la vidéo, la recherche se tourne sensiblement vers la représentation dimensionnelle de l'émotion (i.e modèle du circumplex). Du moment que les modèles discrets constituent une représentation simplifiée de l'émotion et ne peuvent être généralisable à différentes cultures. Par exemple, l'expression d'un sourire peut à la fois être attribué à la joie mais aussi à la peur ou au dégôut dépendamment du contexte, ce qui rend plus complexe l'interprétation de l'expression. Cependant, les modèles dimensionnels peuvent différencier l'émotion en utilisant une meilleure base vectorielle constituée des axes de valence et excitation. Ces deux valeurs, largement utilisées dans le domaine de la psychologie peuvent assigner une plus grande étendue de valeurs associées à un état émotionnel et facilite la convergence de réseaux de neurones profond par l'apprentissage de meilleures caractéristiques. Néanmoins, pour les méthodes d'apprentissage supervisées, modifier la nature de la sortie avec des modèles plus précis, redéfinie les techniques d'apprentissage

ainsi que la façon dont les architectures se structure elles-mêmes par l'intermédiaire des données. Ceci nous conduit à opérer de nouvelles approches.

Comme premièrement suggéré par Simonyan & Zisserman (2014a), Kim *et al.* (2017) ont développé des réseaux de neurones en combinant différentes caractéristiques de haut niveau, les textures (LBP), les formes (SIFT) et les objets. Les chercheurs ont montré que les caractéristiques bas et haut niveau pouvaient être complémentaire et leur combinaison pouvait diminuer ce qu'ils appellent le fossé affectif (i.e la concordance entre les propriétés du signal émotionnel ou ses caractéristiques et les prédictions désirées). On entend par haut niveau les caractéristiques visuelles qui ressortent directement de l'analyse des pixels des images, les objets ou le contexte de l'image, et bas niveau les caractéristiques issues de l'apprentissage profond. D'autres travaux se sont focalisés sur la reconnaissance d'émotions au sein de groupes en étudiant les expressions faciales (plusieurs visages sont présents simultanément) mais également la posture corporelle ou le contexte (i.e l'environnement) (Mou *et al.*, 2015) ou en explorant différents signaux physiologiques tels que l'électro-cardiogramme et le volume respiratoire (Ben Henia & Lachiri, 2017), dans ce dernier cas une variante de l'échelle de valence et excitation a été utilisée. Kollias & Zafeiriou (2018), ont quant à eux, comparé et utilisé plusieurs variations de modèles CNN-RNN pour la prédiction de valence-excitation sur l'ensemble de données Aff-Wild (Kollias *et al.*, 2019) en étudiant les expressions faciales. Leur étude a notamment atteint un niveau de performance en terme de CCC de 0.491 en valence et de 0.311 pour l'excitation.

1.10.1 Résumé

Les expressions faciales sont un medium particulièrement efficace pour évaluer les émotions humaines. Nous avons vu que pour détecter les émotions, il était nécessaire d'une part de choisir un modèle de représentation ainsi qu'une méthode d'extraction de caractéristiques. Dans un premier temps, les modèles en catégorie d'émotions se sont imposés mais les modèles multi-dimensionnels plus précis, comme celui du circumplex, font de plus en plus l'objet de recherche. De plus l'analyse spatio-temporelle de séquences d'images à partir de sources vi-

déos se révèle une plus grande source d'information émotionnelle qu'une seule image. D'autre part le relevé de caractéristiques a grandement évolué depuis ces dernières décennies, notamment les méthodes de descriptions artisanales (basées sur des analyses géométriques du visage ou psychologiques) ont laissé place à des systèmes d'apprentissage automatique de plus en plus performants. L'apprentissage profond de caractéristiques associé à des ensembles de données annotées ont prouvé que les systèmes avaient un grand potentiel de généralisation. Ainsi de nombreux modèles à base de CNN ont été développé de manière à extraire l'information spatio-temporelle de séquences vidéos. Parmi l'ensemble des approches développées dans la littérature deux méthodes font aujourd'hui référence pour le relevé de caractéristiques spatio-temporelle : les réseaux en cascade ainsi que les modèles à convolutions 3D. Les réseaux en cascade de type CNN-LSTM sont aujourd'hui très exploités pour notre type d'application et se révèle particulièrement simple d'implémentation et modulaire. Dans ce type d'architecture les informations spatiale et temporelle sont relevées séquentiellement, ce qui apporte une grande visibilité et un certain contrôle sur la représentation des caractéristiques du visage. Enfin, avec le développement récent de réseaux à convolutions 3D ainsi qu'avec l'apparition de données annotées sur un modèle continu de l'émotion, notre méthode consistera donc à évaluer dans ce cadre des modèles d'apprentissage profond également performants pour d'autres tâches de reconnaissance. Nous détaillerons dans le prochain chapitre notre méthode de conception de tels modèles associés à des tâches spécifiques de pré-traitement ainsi que de transfert d'apprentissage.

CHAPITRE 2

MÉTHODOLOGIE

Dans la première partie, nous avons pu étudier les approches de travaux concernant la détection de l'émotion au travers de différents modèles de représentations mais nous avons surtout pu constater la transition et l'évolution des méthodes artisanales vers les méthodes d'apprentissage machine toujours plus efficaces, notamment par l'intermédiaire d'architectures de réseaux de neurones particulièrement adaptée à la détection de l'information spatio-temporelle. L'apprentissage automatique a d'une part permis de trouver des espaces de représentations de l'émotion directement par la distribution des données mais d'autres part le transfert d'apprentissage rendu possible grâce aux architecture de réseaux de neurones permet d'optimiser des modèles existants et ainsi de gagner en temps d'apprentissage et précision. Les modèles exploitant l'information spatio-temporelle pour la détection de valeurs de valence et excitation pour la vidéo étant encore à ses débuts, nous montrerons les performances obtenues selon les récentes approches de la littérature. Dans cette partie, d'une part nous implémenterons en partie les approches présentées et d'autres part à partir de celles-ci nous proposerons de nouvelles méthodes s'en inspirant.

Dans un premier temps, nous présenterons le transfert d'apprentissage si essentiel à une convergence efficace de nos architectures via des ensembles de données adaptés à la reconnaissance d'émotions à la fois pour des modèles en catégorie et dimensionnels. S'en suivront les méthodes de pré-traitement des données, les approches de représentation spatio-temporelles de l'expression ainsi que certains ajustement pour la prédiction par régression linéaire, et enfin les dernières étapes de post-traitement afin d'améliorer la qualité de la prédiction.

2.1 Présentation de l'approche

Dans cette étude nous nous sommes spécialement intéressés à deux types d'architectures pour la représentation spatio-temporelle de l'émotion : le réseau en cascade de type CNN-RNN et

le 3D-CNN. La Figure 2.1 développe la structure globale de notre chaîne de traitement des données.

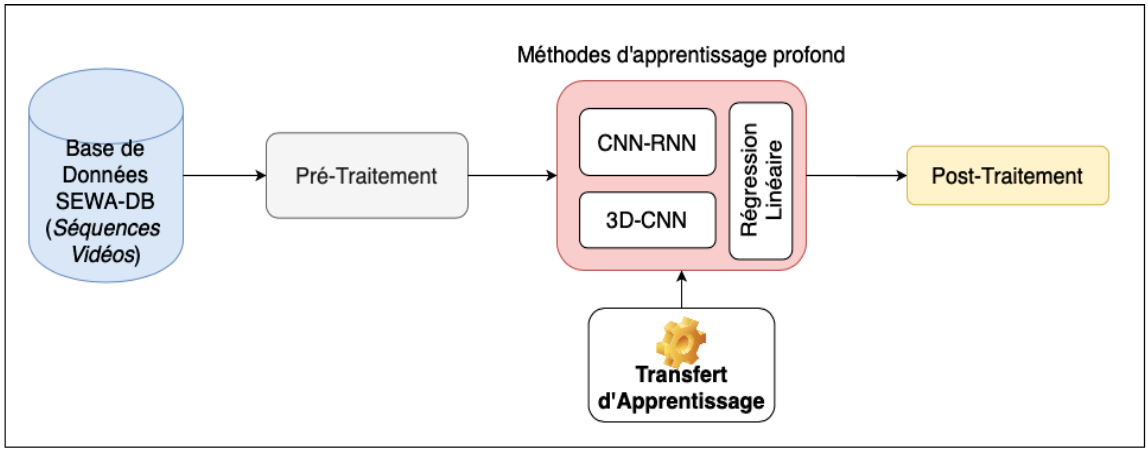

Figure 2.1 Étapes du processus d'apprentissage développé pour l'étude

Premièrement, nous transférons les connaissances acquises par l'intermédiaire de domaines plus ou moins proches à notre tâche cible (VGG-Face, VGG-Face + RAF-DB, ImagNet). Ensuite, lors de la deuxième étape, les données pour notre tâche cible sont arrangées dans un nouveau format, de manière à obtenir des prédictions souhaitables pour nos modèles (alignement des visages, normalisation, augmentation de données, sélection de mini-clips). Nous détaillerons troisièmement, le point central de l'approche, qui est le développement de modèle d'apprentissage profond basé sur des réseaux de neurones à convolutions (CNN-RNN et 3D-CNN). Nous travaillons, lors de cette étude en mode supervisé. C'est-à-dire que les données utilisées sont labellisées de telle manière à servir de guide à l'apprentissage d'architectures de nos réseaux de neurones. Nous comparerons l'effet de différents pré-entraînement mais aussi l'effet de la variation de différents paramètres internes aux architectures. Enfin, lors de la dernière étape nous expliquerons les méthodes de post-traitement nécessaires à l'optimisation des résultats.

2.2 Transfert d'apprentissage avec VGG-Face, RAF-DB et ImageNet

2.2.1 Notions importantes du transfert d'apprentissage

Comme montré précédemment, les architectures de réseaux de neurones sont capables d'apprendre des données, et si celles-ci sont présentes en quantité, les modèles peuvent se montrer très robustes à la présentation de nouvelles données, c'est-à-dire que les modèles généralisent suffisamment pour rester précis sur leur tâche cible. En mode supervisé, grâce aux données, les modèles peuvent modéliser une fonction de transfert établissant la corrélation entre données d'entrées et valeurs cibles. Cependant la modélisation de cette fonction (i.e l'apprentissage de paramètres de réseaux de neurones) peut s'avérer complexifiée non seulement dû au nombre insuffisant d'exemples disponibles mais aussi de la différence trop importante entre la distribution de départ de l'architecture avec sa distribution finale idéale.

L'art de l'apprentissage machine réside dans la capacité de faire converger les architectures vers une précision optimale avec le minimum de données. Cependant, bien souvent la nature des données ainsi que leur récolte n'est pas aisée, particulièrement pour le domaine de la détection de l'émotion. Pour ce faire, il est alors nécessaire de fournir aux architectures une distribution de départ la plus similaire possible à la distribution cible afin d'assurer la convergence des architectures (autrement dit il est nécessaire d'initialiser adéquatement les paramètres des architectures afin d'obtenir une augmentation sensible de la précision à mesure que sont présentés les exemples). Pour un entraînement intégral sans pré-entraînement, en règle générale les distributions de départ sont générées de manière aléatoire avec une distribution uniforme.

En résumé, nous venons de démontrer le besoin d'initialiser adéquatement nos architectures avec des ensembles de données dites « sources » récoltées pour des tâches « sources » les plus similaires possibles à notre tâche « cible ». De plus deux notions importantes sont à intégrer lorsqu'on envisage le transfert d'apprentissage :

- idéalement **la distribution et la nature des données sources doivent être le plus proche des données cibles**. En particulier pour des images présentées à des CNNs, le contenu et

la dimension des images doivent être le plus similaire possible. Par exemple les images sont définies par le format : $(hauteur \times largeur \times nombre\ de\ canaux)$. Les images que nous utilisons sont de nature RGB (trois canaux de couleurs), il est donc important de conserver des images de type RGB. Ceci est dû à la dimension des noyaux des filtres utilisés pour les CNNs. Les opérations de convolutions sur les vecteurs d'entrées et les matrices de poids doivent respecter les règles de calculs vectoriels. De plus, les filtres des images sont initialisés avec des images sources d'une certaine hauteur et largeur, conserver ces caractéristiques permet de conserver l'échelle d'entrée. Dans cette étude, nous choisissons d'utiliser des images de dimension fixe ($100\times80\times3$). Ceci a été défini par la dimension moyenne des trames de nos données cibles.

- **les données sources doivent être en quantité supérieure par rapport au données cibles**. Dans le cas contraire, ceci aurait pour effet d'annuler le bénéfice du transfert d'apprentissage auprès de notre architecture qui aurait tendance à exercer du sur-apprentissage sur nos données sources et généraliserait difficilement sur nos données cibles.

2.2.2 Protocole de pré-entraînement de modèles à convolutions 2D

Nous présentons ici les données sources envisagées pour le transfert d'apprentissage sur nos modèles de référence : VGG-Face, RAF-DB et ImageNet. Nous avons à notre disposition des architectures de type CNN (VGG-11, VGG-16 et Resnet50) pré-entraînées par des tiers sur VGG-Face ou ImageNet. Notre travail a consisté ici à de nouveau entraîner chacune de ces architectures (initialisées avec VGG-Face ou ImageNet) avec RAF-DB afin d'orienter nos architectures vers la détection d'émotions. Les tâches de classification d'émotions et de régression sur les valeurs de valence et excitation étant a priori très similaires.

Le Tableau 2.1 donne un aperçu des ensembles de données utilisés. Y sont détaillés les tâches d'application respectives, le nombre de classes et sujets différents, le nombre d'images récoltées et enfin le taux de précision obtenus en phase de test, notamment par apprentissage profond de réseaux de neurones à convolutions.

Tableau 2.1 Brève description des ensembles de données utilisés pour l'étude

Ensemble de données	**Tâche**	**Nombre de sujets / classes**	**Nombre d'images**	**État de l'art (Précision)**
VGG-Face	Vérification d'identité	2 622 sujets	2.6 M	97.2%
RAF-DB	Classification d'émotion	7 classes	29 672	78.23% (7 Classes) 85.77% (6 Classes)
ImageNet	Classification d'objets	1 000 classes	1.3 M	88.4%

Le pré-entraînement de modèles avec RAF-DB consiste à réaliser une tâche de classification pour sept catégories d'émotions (Joie, Peur, Tristesse, Colère, Surprise, Dégoût, Neutre). Plusieurs problèmes inhérents à la classification se sont posés et ont nécessités des étapes de pré-traitement :

1. **Le contre-balancement des données :** Le nombre d'exemples pour chacune des classes d'émotions varie sensiblement. Le modèle sera capable de mieux généraliser sur les classes présentant le plus grand nombre d'exemples, les autres classes seront plus difficilement détectables et se montrent préjudiciables pour l'apprentissage des autres classes. Plusieurs solutions s'offre à ce cas de figures :

    a. donner le même nombre d'exemples par classe. On donne N_c le nombre d'exemples de la classe c et N le nombre d'exemples à atteindre.

        - par sur-échantillonnage, c'est à dire ajouter des exemples par augmentation de données aux classes en déficit d'exemples jusqu'à atteindre le même nombre d'exemples que la classe en présentant le plus grand nombre. ($N = \max(N_c)$)
        - par sous-échantillonnage, on fixe le nombre d'exemples par classe par rapport à la classe en présentant le plus petit nombre. ($N = \min(N_c)$)

b. attribuer un poids à chacune des classes afin que les modèles donnent plus ou moins d'importance à certaines classes. Le poids W_c attribué à chacune des classes est défini par :

$$W_c = \frac{N_{Tot}}{N_C \times C} \tag{2.1}$$

où N_{Tot} est le nombre total d'exemples, N_C est le nombre d'exemples de la classe, et C est le nombre de classes. On choisit pour cette étude d'attribuer un poids à chacune des classes lors de l'entraînement.

2. **L'ajout d'exemples par augmentation de données :** Il a été observé que pour un ensemble de données de la taille de RAF-DB, l'augmentation de données est bénéfique à l'entraînement pour améliorer les performances. Plusieurs niveaux d'augmentation de données ont été appliqués tels que rotations, décalage horizontal et vertical, effet miroir, augmentation du contraste, augmentation et diminution des hautes lumières.
3. **Le redimensionnement des images :** Les modèles d'apprentissages profond comme les CNNs ne prennent en entrée que des batchs d'images de taille fixe. La taille des images est définie par notre tâche cible de régression sur valence et excitation après transfert d'apprentissage. Il est donné que la taille moyenne des images pour l'ensemble de données cible SEWA-DB, est de 100×80. Les images de RAF-DB et SEWA-DB sont donc toutes redimensionnées à cette dimension.

Il est important de noter que les étapes essentielles d'isolation et d'alignement des visages ont déjà été opérées au préalable et n'ont donc pas nécessité de pré-traitement supplémentaire.

Le type de pré-entraînement utilisé avec RAF-DB fait référence au transfert inductif mentionné précédemment au travers des différents modes de transfert d'apprentissage. La Figure 2.2 schématise le processus de pré-entraînement de modèles avec RAF-DB pour le transfert d'apprentissage.

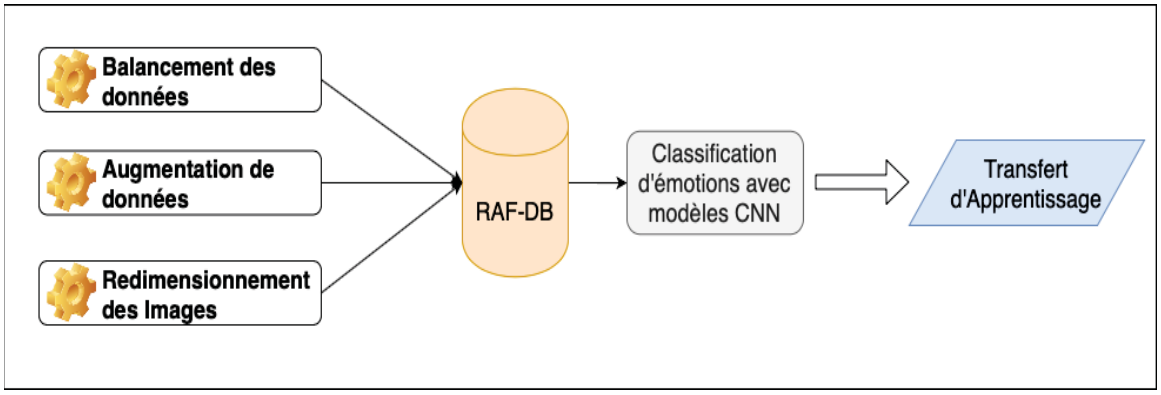

Figure 2.2 Schéma explicatif du processus de pré-entraînement de modèles CNN avec RAF-DB

2.3 Pré-traitement

On décrit ici toutes les étapes de pré-traitement des séquences vidéos de notre base de données. La Figure 2.3 schématise en deux parties principales le pré-traitement des séquences vidéos : l'extraction des trames et visages des vidéos et l'alignement de trames en mini-clips.

Le domaine cible de notre étude repose sur l'ensemble de données SEWA-DB, un ensemble de clips vidéos annotés dans le temps selon les valeurs de valence et excitation, et représentant une unique personne parlant face caméra. La reconnaissance d'expressions faciales avec cet ensemble de données est une tâche particulièrement complexe du moment qu'il existe une faible variété de sujets pour l'entraînement de modèles d'apprentissage automatique et la longueur de chacune des vidéos implique la présence de plusieurs unités émotions. Ainsi d'une part nos modèles peuvent facilement être sujets au sur-apprentissage mais la nature de chacune des vidéos doit être redéfinie afin d'obtenir des prédictions adéquats. En effet une forte variabilité d'émotions pour chaque clip vidéos ne permet pas en l'état de prédire un unique état émotionnel sur l'ensemble d'une vidéo. Par ailleurs, les sujets présentent temporellement différentes poses de tête face à la caméra et différentes occlusions comme les cheveux ou le micro (pour l'enregistrement audio), ceci empêchant dans certains cas la visualisation complète du visage. De ce fait, la structure globale de l'ensemble de données doit être redéfini par des étapes de pré-traitement afin de redéfinir la structure mais aussi la nature de nos données.

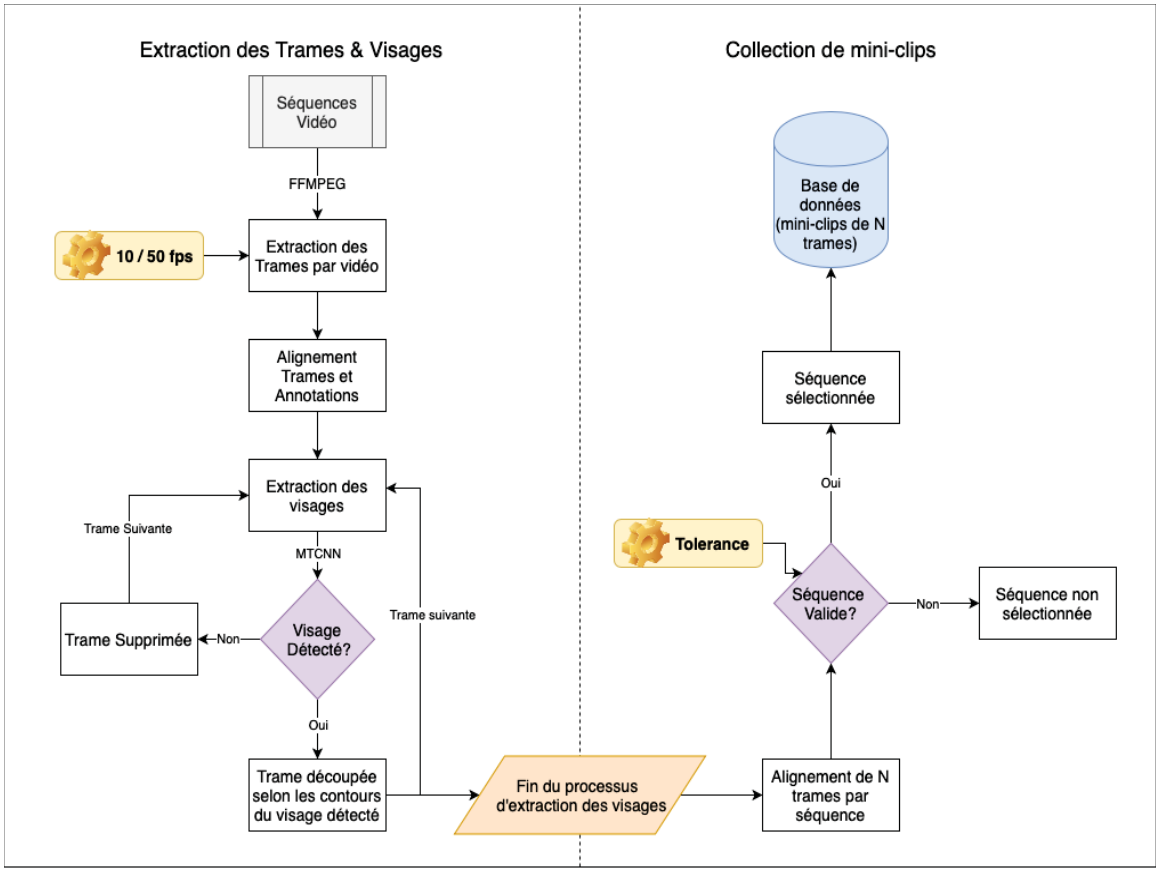

Figure 2.3 Détail du Processus de Prétraitement

Une particularité de cet ensemble de données est de placer les sujets dans un contexte similaire, c'est à dire, face caméra à l'aide d'un ordinateur dans une pièce fermée lors de l'enregistrement vidéo, ce qui limite les mouvements du visage et uniformise l'environnement des sujets. Ainsi les variations de la pose de la tête et de l'illumination des visages restent limitées.

2.3.1 Extraction d'images à partir de séquences vidéos

Dans un premier temps, nos modèles d'apprentissage prenant en entrée des lots de séquences d'images, la première étape consiste à transformer les vidéos en une succession de trames exploitables indépendamment. Afin de faire correspondre les annotations de valence et excitation à chacune des images, il est nécessaire d'extraire les trames à une fréquence correspondant au pas de temps des annotations. Les annotations ont une fréquence de 10 fps (trame par seconde).

Cependant les vidéos ont été enregistrées à une fréquence de 50 fps. Ce qui potentiellement fait perdre une quantité considérable d'information en terme de nombre d'image par vidéo. Notre stratégie est donc de relever à la fois les trames de chacune des vidéos à 10 et 50 fps. Comme plus d'images sont présentes à 50 fps, exactement cinq fois plus, un pas de temps à 10 fps correspondant à 1 trame, sera de cinq trames pour ce même pas de temps à 50 fps. Les annotations manquantes pour 50 fps entre deux pas de temps (t et t+1) sont comblées par recopie de l'annotation du pas de temps t jusqu'au pas de temps t+1. En pratique l'extraction des trames a été réalisé avec le tramework ffmpeg [1].

Une fois les trames de chacune des vidéos extraites, chacune des trames sont alignées avec les annotations correspondantes, de cette manière nous nous assurons que le nombre de trames extraites correspondent au nombre d'annotations. Il est possible que quelques trames supplémentaires soient détectées en fin de vidéo, celles-ci sont donc supprimées.

2.3.2 Extraction des visages

Une fois le travail de transformation de nos données dans un format adéquat, il est nécessaire d'isoler l'information émotionnelle des images par l'extraction des visages. Les trames de chacune des vidéos sont insérées dans un modèle MTCNN, comme décrit dans la Section 1.5.1, afin de détecter la présence d'un visage. Si un visage est détecté, le modèle retourne à la fois les coordonnées du cadre contenant le visage ainsi les points de repères du visage tels que les yeux, la bouche et le nez. Deux cas se présentent alors :

- **un visage est détecté** : l'image est découpée selon les coordonnées du cadre fourni par le modèle MTCNN
- **aucun visage n'est détecté** : l'image est supprimée

1. https ://www.ffmpeg.org

2.3.3 Découpage séquentiel des vidéos et fusion des annotations

Du fait de la longueur des vidéos, celles-ci présentent une succession de plusieurs états émotionnels, il est alors nécessaire de découper la vidéo en mini-clips (séquence de trames contigues) dont la meilleure longueur est à définir. Après la définition d'une longueur de séquence et afin de minimiser la perte d'information, les séquences ne sont pas adjacentes les unes aux autres mais se recouvrent suivant un certain ratio de la longueur. La Figure 2.4 illustre le principe de découpage vidéo en mini-clips avec recouvrement. Une vidéo de 13 trames est fragmentée en mini-clips de cinq trames avec un recouvrement de une trame.

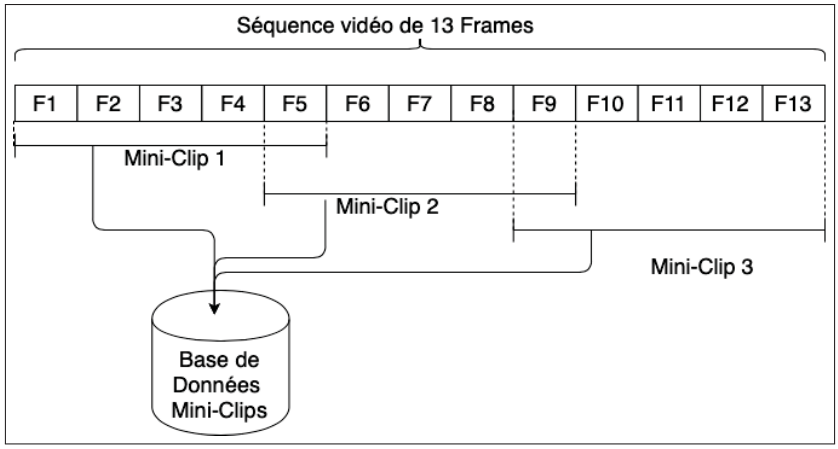

Figure 2.4 Méthode de découpage vidéo en mini-clips

Typiquement dans cette étude, nous avons comparés plusieurs choix de longueurs de séquences avec plusieurs ratios de recouvrement :

- longueurs de séquence : 16 et 64 ;
- ratio de recouvrement (en pourcentage) : 20, 50, et 80.

En résumé, nous avons restructuré notre ensemble de données suivant trois paramètres essentiels, que nous nommerons par la suite les paramètre MIL : (i) le taux fps ; (ii) la longueur de séquence, et (iii) le taux de recouvrement. Suivant ces trois paramètres, le nombre d'exemples ainsi que la valeur de l'information émotionnelle contenue dans les exemples varient fortement.

Ainsi les exemples présentés à nos modèles sont constitués de mini-clips pouvant provenir d'une même vidéo ou d'une autre. La Figure 2.5 montre l'évolution du nombre de séquences suivant la combinaison de ces trois paramètres (taux fps-longueur de séquence-ratio de recouvrement). On donne les courbes pour les ensembles d'entraînement (bleu), de validation (jaune) et de test (vert). Le nombre de séquence croît suivant l'augmentation du taux fps, la diminution de la longueur de séquence et l'augmentation du taux de recouvrement.

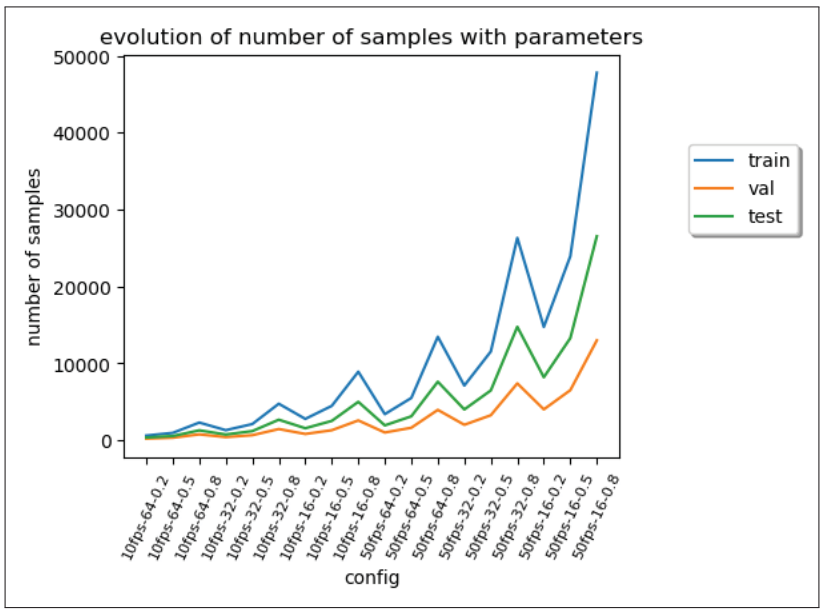

Figure 2.5 Évolution du nombre de séquences pour les 3 ensembles de données : entraînement, validation et test

Mode de fusion des annotations (moyenne, extremum, maximum)

Pour rappel, afin d'entraîner nos modèles nous utilisons le mode supervisé. C'est-à-dire que le gradient d'erreur est calculé à partir de la différence entre les prédictions du modèle avec les annotations. Et le modèle va établir ces prédictions à partir de lots de séquences d'entrées. Cependant du moment qu'une séquence de trames contient plusieurs annotations, il est nécessaire de les fusionner pour n'obtenir qu'une seule valeur cible. Pour ce faire, nous avons expérimenté plusieurs modes de fusion des annotations : le maximum, l'extremum et la moyenne.

La Figure 2.6 montre l'histogramme 2D typique de la distribution des valeurs de valence et excitation de l'ensemble de données SEWA-DB. En réalité, on reproduit ici le modèle du circumplex, en utilisant une base orthonormée pour représenter les valeurs de valence et excitation selon deux axes horizontal et vertical. Un code couleur est indiqué pour représenter la fréquence d'exemples suivant la valeur de l'émotion. Pour l'exemple, nous avons représenté une fusion des annotations avec la moyenne, le taux fps est de 10, la longueur de séquence de 64 trames et le taux de recouvrement de 0.8 sur l'ensemble d'entraînement des modèles. On observe que la majeure partie des séquences sont concentrées autour de l'origine du circumplex et s'étendent vers les valeurs positives de valence et excitation. Ainsi la majeure partie du temps, les sujets ont exprimés une attitude plutôt neutre voire sensiblement optimiste. La variabilité des annotations étant faible, la détection de l'émotion se montre d'autant plus complexe. Les autres modes de fusion, tels que maximum et extremum ne modifie pas particulièrement les histogrammes que ce soit pour les ensembles d'entraînement, de validation, et de test. Prendre l'extremum a tendance à sensiblement mieux répartir les annotations, mais la tendance reste similaire.

Sélection de séquences de trames continues (application d'une tolérance)

Après restructuration de nos données sources ainsi que leurs annotations, une autre difficulté importante est à prendre en compte à propos de la détection des visages. La suppression de certaines trames ne contenant pas de visage a créé pour chacune des vidéos plusieurs discontinuités dans la succession des trames. Autrement dit, deux trames peuvent avoir un pas de temps allongé et proportionnel au nombre de trames les séparant initialement. Si cet écart est trop important, une même séquence pourrait contenir des trames appartenant à l'expression de deux émotions différentes lors de la sélection des séquences au sein d'une même vidéo. C'est pourquoi il est important d'observer l'impact qu'a eu la suppression de trames sur chacune des vidéos. La Figure 2.7 montre les impacts extrêmes qu'a pu avoir la suppression de trames sur deux vidéos.

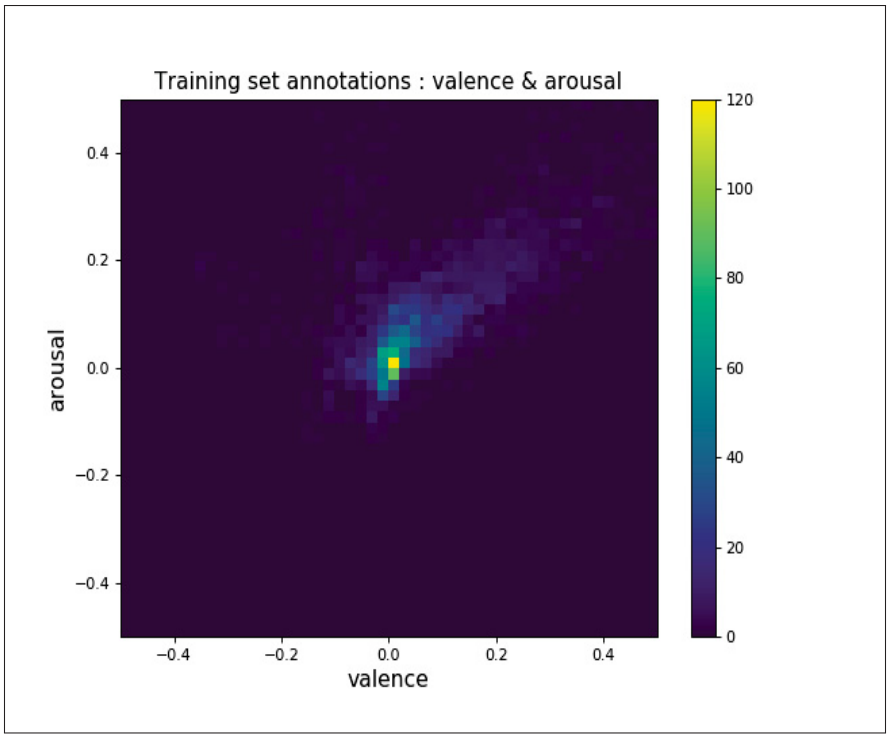

Figure 2.6 Exemple de distribution des valeurs de valence et excitation sur un sous ensemble de données de SEWA-DB

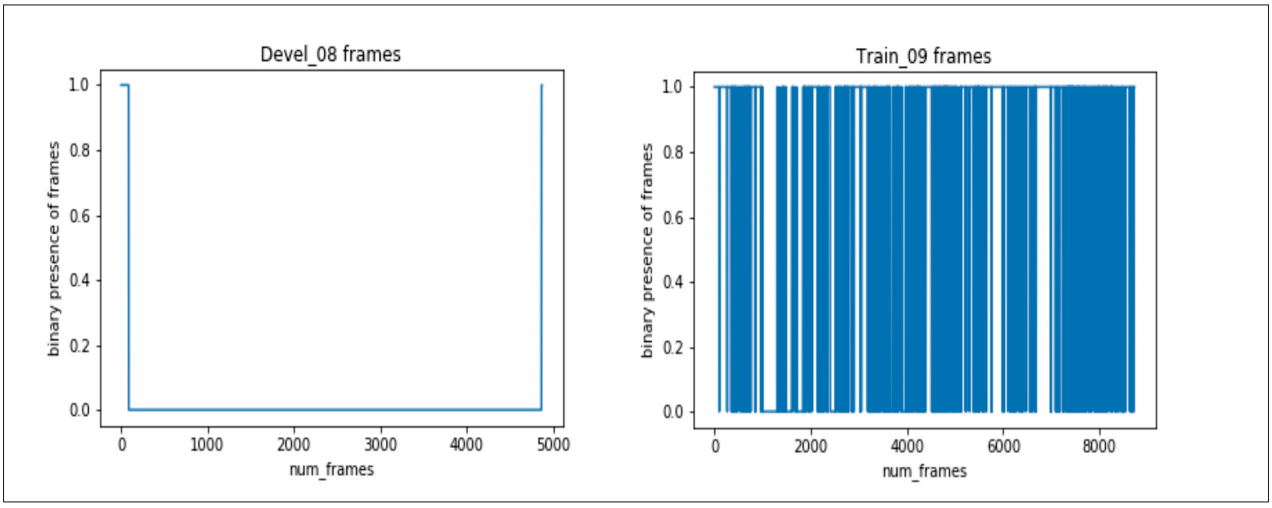

Figure 2.7 Suppression de trames pour deux cas extrêmes de vidéos

On représente la présence binaire de trames en fonction du numéro de la trame, le numéro de la trame correspondant à un ordre chronologique. Le niveau un signifie que celle-ci a été supprimée, le niveau zéro signifie qu'elle a été conservée. À gauche, dans le premier cas seule quelques trames en début et fin de vidéo ont été supprimées. Ceci est dû au fait que l'écran est souvent noir à ces instants. La détection de visages a été particulièrement efficace pour cette vidéo. A droite, dans le second cas, les quelques espaces blancs montrent que peu de trames sont en réalité continues et la vidéo a subi un fort élagage de ces trames. Prélever des séquences à partir des trames restantes de cette vidéo sans prendre en compte leur degré de continuité serait néfaste pour obtenir des prédictions sensées. Ainsi la solution est d'appliquer un niveau de tolérance pour l'écart temporelle existant entre chacune des trames d'une séquence. Si le niveau de tolérance est dépassé la séquence n'est pas sélectionnée pour être ingérée par nos modèles.

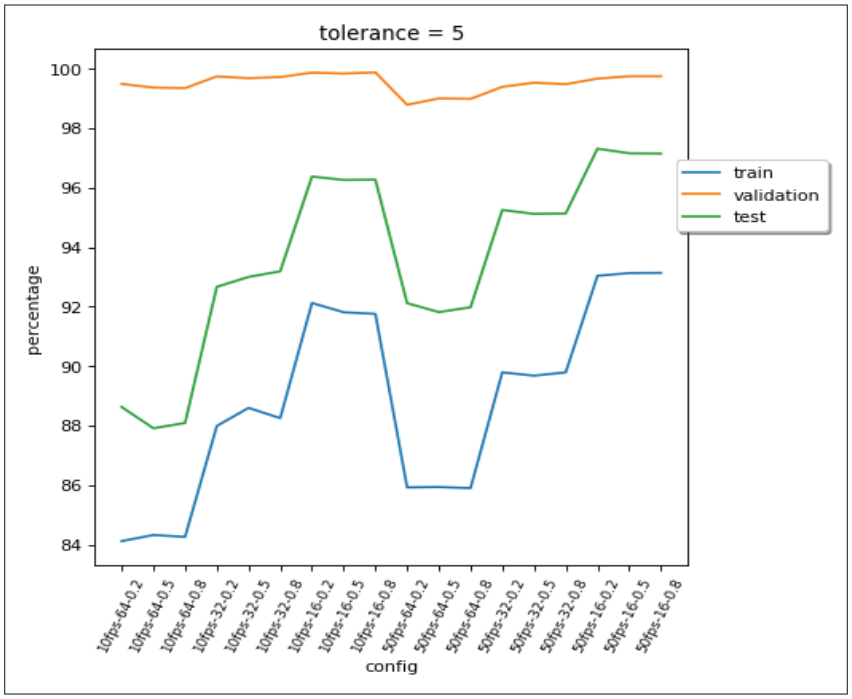

Figure 2.8 Pourcentage de trames totales restantes et les paramètres MIL suivant l'ensemble de données : entraînement (bleu), validation (orange), et test (vert)

La Figure 2.8 développe le pourcentage restant de séquence après sélection suivant le niveau de tolérance, en fonction des paramètres MIL. Avec des séquences de petite longueur, il existe moins de risque de supprimer des exemples, cependant ce risque augmente en augmentant la longueur de séquence ainsi qu'en diminuant le ratio de recouvrement. De plus, une tolérance de 5 signifie que deux trames peuvent au plus être séparées par cinq pas de temps après suppression des trames. Par lecture du graphique, à ce niveau de tolérance, on observe que dans le pire des cas 84% des séquences extraites pour l'entraînement sont exploitables, au mieux 92%. L'ensemble de validation est très peu modifié, et l'ensemble de test se comporte de manière similaire à l'ensemble d'entraînement avec plus de séquences valables. On considère ainsi qu'un niveau de tolérance de 5 est acceptable. A ce niveau, en moyenne, 7% du nombre total de séquences extraites ne sont pas sélectionnables. Ceci prouve que globalement les visages ont pu être détectés sans trop de difficultés.

2.3.4 Augmentation de données

La sélection de séquences d'après un découpage avec recouvrement de chacune des vidéos, implique que un certain nombre de séquences sont très similaires en terme de texture puisque celle-ci contiennent en partie les mêmes trames. De ce fait afin d'apporter plus de variabilité aux textures des séquences nous introduisons de l'augmentation de données. Les trames d'une même séquence sont modifiées selon des paramètres similaires. Les critères retenus pour ce processus sont une modification du niveau de contraste, de luminosité, un décalage vertical et horizontal et un effet miroir. Chacun de ces paramètres sont sélectionnés aléatoirement suivant un certain champ de valeur.

2.4 Représentations spatio-temporelles de l'émotion

De manière similaire à la littérature récente et afin de fournir un modèle de référence sur lequel nous pouvons apporter une certaine modularité, notre étude se base sur différentes déclinaisons d'une architecture CNN. À partir de ce standard notre objectif était de valider certains paramètres internes à nos modèles permettant d'améliorer ou non la description spatio-temporelle

de séquences vidéo pour la régression. Dans un premier temps nous montrerons l'implémentation d'une architecture en cascade de type 2D-CNN-LSTM. Ensuite nous détaillerons la modélisation d'une architecture 3D-CNN issue de l'expansion d'une architecture 2D-CNN suivant différents paramètres et techniques.

2.4.1 Architecture CNN-LSTM

Notre premier modèle consiste en une juxtaposition d'un modèle d'extraction de caractéristiques spatiales ; le CNN, avec un descripteur de l'information temporelle ; le LSTM. Les images de chacune des séquences vidéos sont représentées successivement par les couches de convolutions du CNN sous la forme de vecteurs caractéristiques. Ceux-ci sont accumulés en sortie du CNN afin d'alimenter un LSTM pour la description temporelle globale de la séquence vidéo. Le vecteur de caractéristiques spatio-temporel résultant de ce dernier bloc sert enfin à la prédiction des valeurs de valence et excitation pour la détection de l'émotion. La Figure 2.9 schématise l'architecture en cascade de type CNN-LSTM développée.

On peut décrire succinctement la méthode d'implémentation de cette architecture en deux étapes distinctes :

1. **Extraction des caractéristiques du visage par CNN** : À partir de différents pré-entraînement d'un CNN, les caractéristiques de l'ensemble des trames des vidéos de SEWA-DB sont extraites et sauvées sous forme d'un vecteur caractéristique à une dimension.
2. **Apprentissage de caractéristiques temporelles par LSTM** : Le LSTM est alimenté en entrée par une concaténation des vecteurs caractéristiques de la séquence de trames et modélise l'aspect temporelle du mini-clip sous forme d'un vecteur à une dimension permettant la régression des valeurs de valence et excitation. Notre étude s'est porté sur deux types de CNN pré-entraîné pour cette architecture :
    a. VGG-16 pré-entraîné avec VGG-Face
    b. VGG-16 pré-entraîné avec VGG-Face et optimisé avec RAF-DB.

De cette manière, nous pourrons étudier les bénéfices d'une spécialisation de nos modèles à une tâche de reconnaissance d'émotion avant apprentissage de l'application finale.

«Fine-tuning » de modèles pré-entraînés

L'optimisation avec RAF-DB a été effectuée en « gelant » plus ou moins de couches de convolutions de l'architecture VGG-16 afin d'améliorer les performances. Geler les couches signifie que les paramètres de ces couches ne sont pas modifiables. En effet il est connu que les couches plus profondes des architectures ont de meilleures capacités de généralisation que les couches en sortie d'architecture, ainsi ne modifier que les paramètres des couches de convolutions en sortie de modèles peut s'avérer bénéfique pour le transfert d'apprentissage. Dans notre cas, les couches de convolutions plus profondes sont orientées pour une tâche générale d'identification des visages (initialisation avec VGG-Face) alors que les couches de sortie sont plus spécialisées pour la reconnaissance d'émotions (optimisation sur RAF-DB). Le nombre de couches à optimiser avec RAF-DB restant à déterminer.

Description détaillée de l'architecture

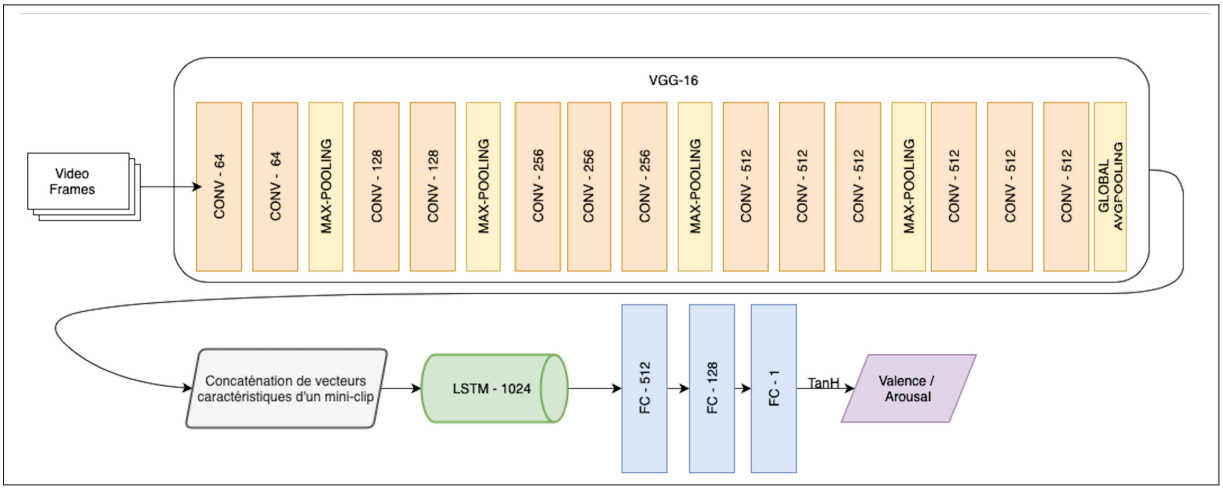

Figure 2.9 Architecture VGG-16-LSTM

Dans la Figure 2.9, les couches de convolutions sont indiquées en orange ainsi qu'avec leur nombre de canaux de sorties (a.k.a. nombres de filtres). Les couches en jaunes sont les couches

de «max pooling» et «global average pooling» permettant de réduire la dimension des vecteurs en sortie des couches de convolutions. Après transformation de chacune des trames d'une séquence en un vecteur caractéristiques, l'ensemble de ces vecteurs sont concaténés afin de représenter l'ensemble de la séquence. Les vecteurs caractéristiques sont long de 512 unités, ceci signifie par exemple que pour des séquences de 64 trames, le vecteur d'entrée du LSTM a pour dimensions (64, 512).

Chacune des séquences sont enfin transformées pour obtenir des vecteurs de 1 024 unités. La régression est enfin opérée par un ensemble de trois couches densément connectées (respectivement de 512, 128 et 1 unités). La fonction d'activation tangente hyperbolique est utilisée comme activation finale pour la prédiction de valence et excitation. A l'exception de la dernière FC («fully connected» ou densément donnectée) l'ensemble des fonctions d'activation des couches de convolutions, LSTM et FC du modèle sont de type «ReLU».

2.4.2 Architecture 3D-CNN

Contrairement aux architectures en cascade, le modèle 3D-CNN permet de représenter l'information spatio-temporel de séquences vidéos simultanément, en intégrant une dimension temporelle aux noyaux de convolutions. Le revers de cette architecture est le nombre important de paramètres à optimiser lors de l'apprentissage, ce qui en fait un modèle particulièrement sujet au sur apprentissage et complexifie d'autant plus notre tâche. Il est donc essentiel d'obtenir une initialisation adéquate de nos modèles à partir de tâches similaires ou offrant une bonne capacité de généralisation à notre tâche cible.

Un autre problème relatif à la reconnaissance vidéos d'émotions est que peu d'ensemble de données au format vidéo de taille conséquente sont disponibles pour ce type de tâche et préentraîner sur ces ensembles de données particulièrement spécialisés ne nous permettent pas de généraliser à de nouvelles tâches. Actuellement, les ensembles de données pour la reconnaissance d'émotions font plutôt état d'images 2D annotées suivant les modèles en catégorie ainsi que les modèles dimensionnels de représentations de l'émotion plutôt que pour des sé-

quences vidéo. Cependant, une solution proposée par Carreira & Zisserman (2017) a été de bénéficier de l'entraînement de modèles à convolutions à partir de ces ensembles de données 2D et de les transposer au domaine 3D par expansion de leur architecture. Ainsi les paramètres appris dans un domaine à deux dimensions ont pu être élargi au domaine à trois dimensions. En d'autres termes, la représentation de l'information spatiale d'images peut être transmis à une représentation spatio-temporelle de données vidéos. On rappelle que la vidéo n'est autre que l'alignement de trames dans le domaine temporelle, ainsi seule une dimension sépare le domaine vidéo de l'image.

L'objectif de notre étude est alors de transposer les modèles pré-entraînés avec des ensembles de données 2D au domaine 3D par expansion des poids des architectures développées. Puis enfin de spécialiser nos modèles sur notre tâche cible pour la régression de l'émotion. À partir de ces modèles 3D étendus, nous verrons qu'il existe plusieurs méthodes d'extension des paramètres mais aussi plusieurs techniques d'apprentissage de la dimension temporelle encore inexploitée. La Figure 2.10 explicite notre modèle de prédiction de l'émotion par régression linéaire avec une architecture VGG-16 étendue en 3D. Voici la liste exhaustive des architectures étendues développées et comparées dans cette étude :

- VGG-11-BN
- VGG-16
- VGG-16-BN (architectures avec batch normalisation)
- Resnet50

2.4.3 Complexité des architectures CNN-LSTM et 3D-CNN

Afin de bien mesurer la faisabilité technique de notre étude et la possibilité de déploiement des modèles REF en temps réel, il convient de comparer le niveau de complexité des architectures développées, notamment en terme de coût mémoire et temps de calcul. En général pour les modèles à réseaux de neurones, deux critères interviennent dans nos estimations : le nombre

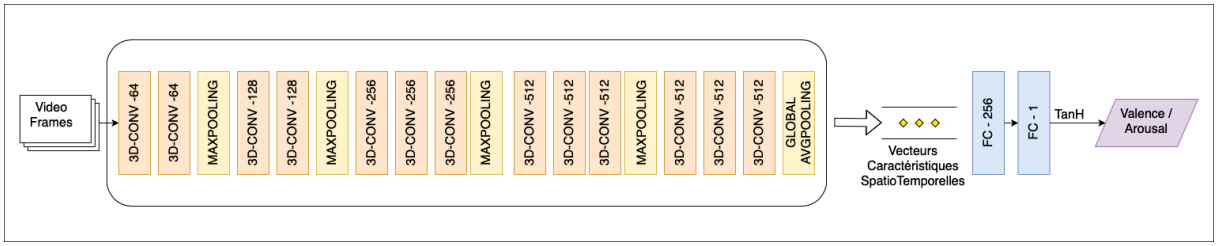

Figure 2.10 Architecture VGG-16-3D

de paramètres à optimiser ainsi que la durée d'une epoch pour l'apprentissage (une epoch correspondant à l'optimisation de nos modèles sur l'ensemble des données d'entraînement). Le Tableau 2.2 présente en exemple ces critères pour chacune des architectures développées. Les valeurs données représente l'entraînement des modèles sur l'ensemble de données SEWA-DB avec une taille de batch de 8 séquences vidéos enregistrées à 10 fps. Les valeurs données sont une moyenne obtenue sur le type d'architecture. Premièrement les modèles en cascade s'avère peu gourmands, à la fois en mémoire et en temps de calcul. Nous avons fait le choix d'entraîner seulement le bloc LSTM à partir des caractéristiques extraites d'un CNN, seules les relations temporelles entre caractéristiques de frames sont apprises. Ceci limite donc les besoins en mémoire et le temps de calcul comparé à un entraînement intégral de ce type de d'architecture. Au contraire les modèles 3D-CNN apprenant conjointement les domaines spatio-temporels, les filtres de convolutions sont appris en même temps que les relations temporelles entre trames. Le nombre de paramètres optimisables est donc plus conséquent et la durée d'entraînement des modèles croit proportionnellement. Dans notre étude les modèles 3D-CNN sont donc environ 5 fois plus volumineux en capacité mémoire et en moyenne 5 fois plus long à opérer un visionnage complet des données d'entraînement.

Tableau 2.2 Exemples de valeurs pour les critères de complexité des modèles en cascade CNN-LSTM et 3D-CNN développés

	Nombre de paramètres	Durée d'une epoch
CNN-LSTM	8 M	5 min
3D-CNN	45 M	60 min

2.4.4 Modèles 2D-CNN de référence utilisé pour l'expansion 3D

Notre méthode a été d'initialiser différents modèles 2D-CNN avec RAF-DB afin d'étudier l'amélioration de performances potentielles de modèles 2D-CNN et 3D-CNN sur des séquences vidéos. Pour ce faire, nous avons entraîner trois types d'architectures à convolutions : VGG-11, VGG-16 et ResNet50. On présente ci-après les architectures utilisées dans cette étude. Les architectures de type VGG et ResNet sont divisées typiquement en blocs successifs de convolutions et pooling. Le nombre de couches densément connectées utilisées pour l'analyse des représentations formées en bout d'architecture sont variables suivant la tâche finale.

2.4.4.1 Architecture type VGG

Les Tableaux 2.3 et 2.4 développent respectivement les architectures de VGG-16 et VGG-11. Les architectures sont typiquement une séquence de blocs de convolutions, chacun des blocs comportant un à trois couches de convolutions et séparés les uns des autres par des couches de MaxPooling. Pour chacune des architectures VGG-11 et VGG-16 nous avons différencié les versions avec et sans normalisation des lots d'entrée. Ceci consiste à ajouter des couches de normalisation de lots à la suite de chacune des couches de convolutions afin de faciliter la convergence d'apprentissage des architectures (Ioffe & Szegedy, 2015). En effet, l'ajout d'une couche de «Batch Normalisation» a pour but de centrer la distribution sur un petit ensemble de données et évite ainsi aux modèles de diverger lors de la recherche de paramètres optimaux.

2.4.4.2 Architecture type ResNet

ResNet est une architecture bien établie dans la communauté de l'apprentissage profond, et connue pour avoir remporté la compétition ImageNet (ILSVRC, 2015) [2] pour des tâches de classification. Ce modèle a solutionné en son temps le problème d'évanouissement du gradient pour des modèles d'apprentissage avec de nombreuses couches de convolutions. De manière similaire au RNN avec le LSTM. À mesure que l'architecture présente un grand nombre de

2. http ://image-net.org/challenges/LSVRC/2015/

Tableau 2.3 Modèle VGG-16 pré-entraîné avec VGG-Face

Blocs	**Type de Couche**	**Nombre de Filtres, Noyaux de Conv., Stride**
Bloc 1	2 × Conv. 2D 1 × Max Pooling	64, 3×3, 1×1 2×2, 2×2
Bloc 2	2 × Conv. 2D 1 × Max Pooling	128, 3×3, 1×1 2×2, 2×2
Bloc 3	3 × Conv. 2D 1 × Max Pooling	256, 3×3, 1×1 2×2, 2×2
Bloc 4	3 × Conv. 2D 1 × Max Pooling	512, 3×3, 1×1 2×2, 2×2
Bloc 5	3 × Conv. 2D 1 × Max Pooling	512, 3×3, 1×1 2×2, 2×2
Bloc 6	Fully Connected	4 096
Bloc 7	Fully Connected	4 096
Bloc 8	Fully Connected	2 622
	Softmax	

couches de réseaux de neurones, la rétro-propagation du gradient d'erreur tend à l'annuler, et l'optimisation des poids de l'architecture est de plus en plus complexe. La particularité de l'architecture est de régulièrement implémenter des connexions résiduelles afin d'assurer que l'information ne se perd pas lors de la rétro-propagation du gradient de l'erreur. La structure du modèle s'organise alors en plusieurs blocs typiquement composés chacun de trois couches de convolutions, chacune associée à une couche de batch normalisation et une activation ReLU. Ainsi l'empilement de ces blocs permet de construire des modèles d'apprentissage à convolutions plus profond en amenuisant l'effet de l'évanouissement du gradient. Plusieurs architectures resnet ont été proposées en faisant varier l'organisation de ces blocs (e.g ResNet50, ResNet101, ResNet152). La Figure 2.11 schématise la structure typique d'un bloc resnet. Chaque bloc Resnet utilise un bloc à deux convolutions (bloc de gauche) pour les resnet moins profond comme ResNet18 ou ResNet34, ou trois convolutions (bloc de droite), pour ResNet50,

Tableau 2.4 Modèle VGG-11 pré-entraîné avec ImageNet

Blocs	**Type de Couche**	**Nombre de Filtres, Noyaux de Conv., Stride**
Bloc 1	1 × Conv. 2D 1 × Max Pooling	64, 3×3, 1×1 2×2, 2×2
Bloc 2	1 × Conv. 2D 1 × Max Pooling	128, 3×3, 1×1 2×2, 2×2
Bloc 3	2 × Conv. 2D 1 × Max Pooling	256, 3×3, 1×1 2×2, 2×2
Bloc 4	2 × Conv. 2D 1 × Max Pooling	512, 3×3, 1×1 2×2, 2×2
Bloc 5	2 × Conv. 2D 1 × Max Pooling	512, 3×3, 1×1 2×2, 2×2
Bloc 6	Fully Connected	4 096
Bloc 7	Fully Connected	4 096
Bloc 8	Fully Connected	1 000
	Softmax	

ResNet101, ResNet152. Le Tableau 2.5 développe la structure d'un ResNet50 que nous avons utilisé pour l'étude et qui comporte 50 couches de convolutions.

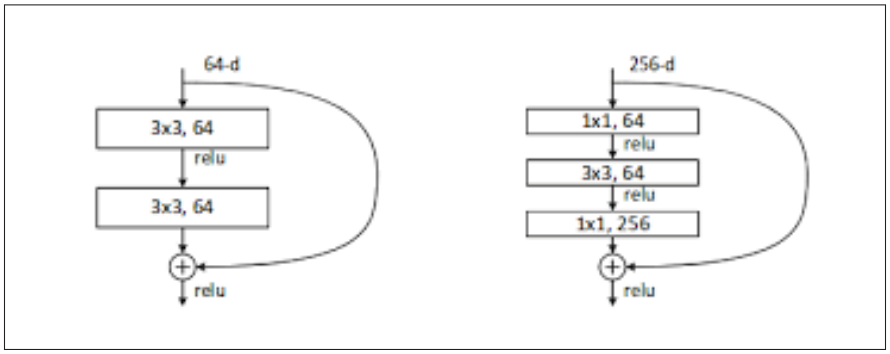

Figure 2.11 Structure typique d'un bloc Resnet [3]

3. https://medium.com/@14prakash/understanding-and-implementing-architectures-of-resnet-and-resnext-for-state-of-the-art-image-cf51669e1624

Tableau 2.5 Modèle ResNet50 pré-entraîné avec VGG-Face

Blocs	**Types de couches**	**Noyaux de Conv., Nombre de Filtres, (Stride)**	**Dimension de sortie**
Conv1	1 x Conv2D	7 x 7, 64, (Stride 2)	112 x 112
Conv2_x (x3)	1 x MaxPooling2D	3 x 3, - , (Stride 2)	56 x 56
	1 x Conv2D 1 x Conv2D 1 x Conv2D	$\begin{bmatrix} 1\times1,64 \\ 3\times3,64 \\ 1\times1,256 \end{bmatrix} \times 3$	
Conv3_x (x4)	1 x Conv2D 1 x Conv2D 1 x Conv2D	$\begin{bmatrix} 1\times1,128 \\ 3\times3,128 \\ 1\times1,512 \end{bmatrix} \times 4$	28 x 28
Conv4_x (x6)	1 x Conv2D 1 x Conv2D 1 x Conv2D	$\begin{bmatrix} 1\times1,256 \\ 3\times3,256 \\ 1\times1,1\,024 \end{bmatrix} \times 6$	14 x 14
Conv5_x (x3)	1 x Conv2D 1 x Conv2D 1 x Conv2D	$\begin{bmatrix} 1\times1,512 \\ 3\times3,512 \\ 1\times1,2\,048 \end{bmatrix} \times 3$	7 x 7
	1 x 1 AveragePooling2D		
FC1	4096		
FC2	4096		
FC3	2622		
	Softmax		

2.4.5 Expansion de modèles 2D à 3D

La technique d'expansion des poids 2D-CNN à 3D-CNN consiste à ajouter une 3$^{\text{ème}}$ dimension aux noyaux de convolutions. On rappelle que chacun des noyaux de convolutions, constitue un filtre qui transformera notre séquence d'images d'entrée en un vecteur de caractéristiques plus compact. Afin de mieux comprendre ce procédé, nous détaillons l'aspect tensoriel définissant les architectures de réseaux de neurones. Typiquement, les couches de convolutions sont définies par trois paramètres dimensionnels : (canaux de sortie, canaux d'entrée, noyaux de filtres). Le nombre de canaux d'entrée et sortie correspondent au nombre de filtres de convolutions tandis que les noyaux de filtres correspondent à la taille des filtres. Pour des convolutions 2D, les noyaux de filtres sont à deux dimensions (hauteur, largeur). Dans le cas de convolutions 3D, les noyaux des filtres sont à trois dimensions (pas de temps, hauteur, largeur). Ainsi les tenseurs de poids de couches de convolutions ont pour format :

- 2D : (canaux de sortie, canaux d'entrée, hauteur, largeur), tenseur à quatre dimensions
- 3D : (canaux de sortie, canaux d'entrée, pas de temps, hauteur, largeur), tenseur à cinq dimensions

La Figure 2.12 schématise la méthode d'inflation. Un noyau de convolution 2D est recopié n fois le long d'une nouvelle dimension de taille n. Comme nous le voyons ensuite, cette méthode nous permet d'appliquer une initialisation particulière des noyaux 3D des filtres de convolutions ainsi que de définir le principe d'ancrage (masking).

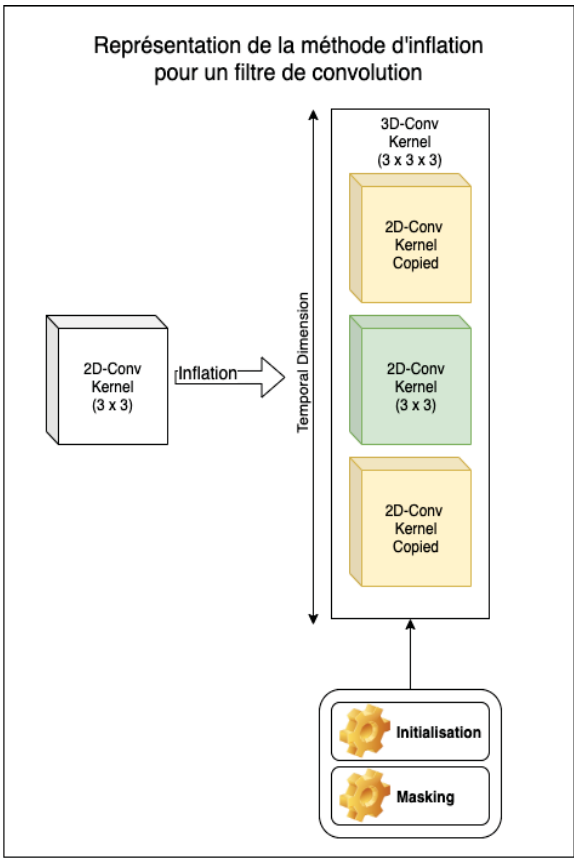

Figure 2.12 Représentation de la méthode d'inflation pour un filtre de convolution

2.4.5.1 Centrage vs recopiage des poids

Vient alors plusieurs possibilités d'initialisation des poids en 3D à partir des noyaux de convolutions 2D :

- soit les poids sont **recopiés** dans chacune des dimensions temporelles. Dans ce cas-ci les poids sont normalisés à la dimension temporelle. Exemple : Pour une architecture de type VGG, les noyaux de convolutions 2D ont pour dimension (3×3). L'expansion des poids rend cubique le noyau de convolution, et a pour dimension (3×3×3), ainsi les poids 2D sont divisés par trois avant d'être recopiés sur la dimension temporelle.
- soit les poids sont **centrés**, les poids sont recopiés seulement sur la dimension temporelle centrale, les autres dimensions temporelles peuvent être initialisées à zéro ou de manière aléatoire et uniforme.

Les Figures 2.13 et 2.14 donnent des exemples de tenseurs de poids suivant ces différents cas de figure. On a représenté à partir d'un modèle VGG-16-3D, un des noyaux de filtres de la première couche de convolution de dimension (3×3×3). Dans le premier cas, on observe que les poids 2D sont recopiés sur chacune des dimensions temporelles et normalisés. Dans le second cas les poids sont centrés sur la 2ème dimension temporelle, les poids restants initialisés à zéro.

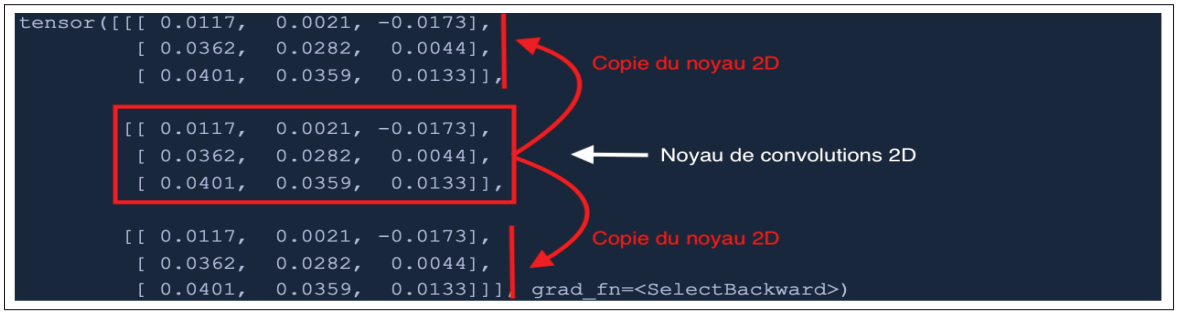

Figure 2.13 Cas 1 : Recopiage des poids normalisés

```
tensor([[[ 0.0000,  0.0000,  0.0000],
         [ 0.0000,  0.0000,  0.0000],
         [ 0.0000,  0.0000,  0.0000]],

        [[ 0.0350,  0.0064, -0.0520],
         [ 0.1087,  0.0846,  0.0133],
         [ 0.1204,  0.1077,  0.0400]],

        [[ 0.0000,  0.0000,  0.0000],
         [ 0.0000,  0.0000,  0.0000],
         [ 0.0000,  0.0000,  0.0000]]], grad_fn=<SelectBackward>)
```

Figure 2.14 Cas 2 : Centrage des poids

2.4.6 Ancrage des poids 2D (masking)

De manière similaire à l'optimisation d'architecture 2D-CNN avec RAF-DB, on choisit de ne modifier que certains paramètres de poids lors de l'apprentissage. En effet lors de l'expansion des poids, les poids 2D ont soit été copiés ou centrés mais dans aucun des cas, la dimension temporelle n'a réellement bénéficié du transfert d'apprentissage. Ainsi nous partons du principe qu'il est possible de conserver la distribution de la dimension spatiale initiale afin de transférer les connaissances à la dimension temporelle en fixant les poids de la dimension spatiale transférée. Ainsi pour chacun des noyaux des filtres de convolutions, toutes les dimensions temporelles ont des paramètres optimisables sauf la dimension temporelle centrale vue comme la base des connaissances de l'information spatiale. Nous appelons cette méthode, le «masking» des poids 2D ou «spatial-freezing». Par conséquent avec ce procédé chacune des couches de convolutions du modèle possèdent un noyau de poids 2D non modifiables transférant leur connaissance à la dimension temporelle au cours de l'apprentissage.

2.4.7 Dilution temporelle

Comme montré par la Figure 1.19, il est possible d'élargir les champs récepteurs des noyaux de convolution et ainsi de couvrir de plus larges zones sur les tenseurs de caractéristiques intermédiaires aux couches de convolutions. Ce principe est particulièrement intéressant pour établir des corrélations temporelles sur de plus grand pas de temps. Par cette méthode (Yu & Koltun, 2015) ont montré notamment que cela pouvait améliorer la segmentation d'images 2D. Ainsi

nous cherchons à reproduire ce procédé en appliquant une dilution temporelle aux couches de convolutions. Seule la dimension temporelle dans cette étude a vu ses champs récepteurs étendus. Pour ce faire nous avons appliqué différents niveaux de dilutions suivant différents blocs de convolutions des architectures. Typiquement, nous avons séparés chaque architecture en blocs de convolutions et appliqué un niveau décroissant de dilution par puissance de deux dans la direction de la profondeur du réseau. La Figure 2.15 explicite ce dernier point à partir d'une architecture de type VGG-16. Dans cette configuration le niveau maximum de dilution est de huit et l'architecture est divisée en quatre blocs de convolutions. Le même procédé est appliqué aux architectures VGG-11 et ResNet50.

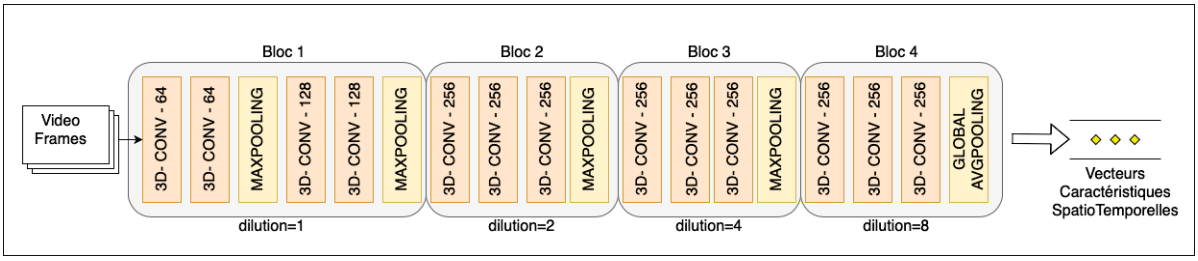

Figure 2.15 Application de la dilution temporelle à une architecture VGG-16-3D

2.5 Prédiction de l'émotion par régression

Grâce aux représentations spatio-temporelles obtenues par les architectures présentées précédemment, nous avons pu opérer la régression linéaire des valeurs de valence et excitation. Pour ce faire nous avons choisi d'utiliser un modèle de réseaux de neurones de type perceptron à deux ou trois couches placé comme dernier bloc de nos architectures. Ainsi l'apprentissage des modèles développés se fait intégralement par rétro propagation de l'erreur entre les prédictions et annotations des données.

2.5.1 Multiplication des annotations

Il a été observé que les annotations ainsi que les paramètres des modèles possédaient initialement des valeurs semblables proche de zéro. Ainsi il est plus difficile lors de l'entraînement de trouver une distribution de paramètres optimale pour nos architectures et le choix d'hyperparamètres adéquats pour assurer la convergence des modèles s'avère plus complexe. De ce fait, modifier les valeurs cibles lors de l'entraînement de manière à obtenir un domaine de distribution des annotations avec des valeurs plus élevées permettrait a priori aux architectures d'accroître les domaines de distribution possibles et d'avoir plus de marge de manœuvre dans leur entraînement. Nous avons donc choisi dans certains cas de multiplier les valeurs réelles cibles par 100. Ceci ne modifie en rien la nature des annotations qui ont simplement changer d'échelle de valeurs. Nous comparerons donc l'effet de ce changement d'échelle sur l'entraînement de nos modèles.

2.5.2 Méthode d'optimisation de l'apprentissage machine

Afin d'entraîner nos modèles de réseaux de neurones par rétro-propagation de l'erreur, il est nécessaire de choisir deux fonctions essentielles adaptées à notre application de classification pour le pré-entraînement avec RAF-DB et de régression linéaire pour l'application cible avec SEWA-DB (une fonction de coût ainsi qu'une fonction d'activation pour la prédiction finale). Pour la classification : la fonction de coût typiquement adoptée est la fonction d'entropie croisée et la fonction d'activation log-softmax. La fonction de coût entropie croisée associée à la matrice des poids W est donné par :

$$J(W) = -\frac{1}{N}\sum_{n=1}^{N}\left[y_n \log\ (\widehat{y}_n) + (1-y_n)\log(1-\widehat{y}_n)\right] \tag{2.2}$$

où y_n est l'annotation de l'échantillon n, $\widehat{y}_n$ la prédiction associée et N le nombre d'échantillons.

Pour la régression : la fonction de coût utilisée est l'erreur quadratique moyenne (MSE) et la fonction d'activation tangente hyperbolique. Cette dernière convient parfaitement au do-

maine de valeur de notre tâche cible, soit [-1 ; 1] ou [-100 ; 100] dans le cas où nous appliquons un multiplicateur d'annotation. Enfin les modèles sont optimisés avec la méthode Adam (Kingma & Ba, 2014).

2.6 Post-Traitement

Afin d'améliorer les performances après entraînement des modèles, il est possible d'appliquer un ensemble de biais et techniques sur les prédictions. En effet, comme la phase d'apprentissage peut apporter un certain nombre de biais propres aux données d'entraînement, il est nécessaire de traiter les nouvelles prédictions des modèles de manière à prendre en compte ces décalages. Pour ce faire nous appliquons séquentiellement des étapes de normalisation d'échelle, et de décalage verticaux et horizontaux des prédictions. La Figure 2.16 résume les étapes de post-traitement opérées. À partir des trois sous ensembles de données de SEWA-DB (entraînement, validation et test), sont transformées successivement selon les trois étapes de post-traitement.

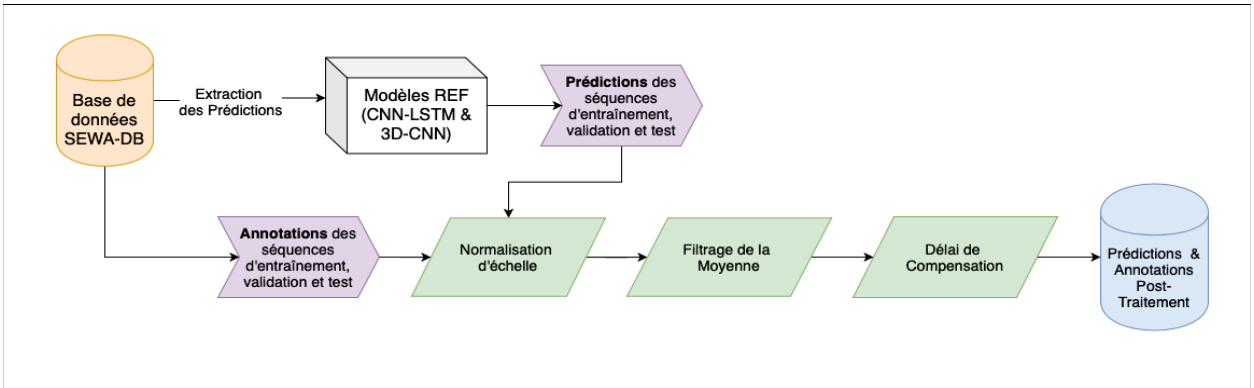

Figure 2.16 Chaîne de processus de Post-Traitement

2.6.1 Normalisation d'échelle

Lors de l'inférence, i.e l'évaluation de nos modèles avec des ensembles de données test, il vient de manière courante que les distributions des valeurs cibles de test soient sensiblement

différentes à celles des valeurs cibles d'entraînement. De ce fait, il convient d'assimiler les deux distributions de manière à améliorer les performances sur des données test (non «connues») de nos modèles. Les prédictions et annotations de l'ensemble de test sont recalculées à partir de la distribution de l'ensemble d'entraînement. L'équation 2.3 défini la formule de normalisation d'échelle.

$$y_{norm} = \frac{y - \overline{y_t}}{std\,(y_t)} \tag{2.3}$$

Les nouvelles prédictions y sont normalisées à l'aide de la moyenne des exemples d'entraînement $\overline{y_t}$ et de leur déviation standard $std\,(y_t)$

2.6.2 Filtrage par la moyenne

On retire la moyenne des prédictions de l'ensemble d'entraînement aux prédictions test, ce qui a pour effet d'aligner horizontalement dans un plan 2D les projections des annotations avec les prédictions. L'équation 2.4 défini la formule du délai de compensation. On donne $\widehat{y}_n$ la n$^{\text{ième}}$ prédiction, $\widehat{\overline{y}}_t$, la moyenne des prédictions des données d'entraînement et $\overline{y}_t$, la moyenne des annotations des données d'entraînement.

$$\widehat{y}_{n(new)} = \widehat{y}_n + \widehat{\overline{y}}_t - \overline{y}_t \tag{2.4}$$

2.6.3 Délai de compensation

Les prédictions sont décalées sur un nombre n de pas de temps antécédents ou futurs afin de mieux aligner les prédictions temporellement. Ceci a pour effet d'aligner verticalement dans un plan 2D les projections des annotations avec les prédictions. L'équation 2.5 défini la formule du délai de compensation. On donne $\widehat{y}_n$ la n$^{\text{ième}}$ prédiction et d le nombre de pas de temps

associé au délai. Afin d'optimiser les métriques de performances on fait varier l'indice d sur un intervalle relativement étendu (e.g [-10 ;10]).

$$\widehat{y}_n = \widehat{y}_{n-d} \tag{2.5}$$

2.7 Résumé

Dans cette partie, nous avons pu développer notre approche de représentation et détection spatio-temporelle d'émotions pour des séquences vidéos. Nous avons vu qu'il était essentiel pour nos architectures d'avoir une initialisation adéquate grâce au transfert d'apprentissage d'après des tâches de classification d'émotion similaires à notre application de régression des valeurs de valence et excitation. De plus, il a été nécessaire de restructurer l'ensemble de nos données en mini-clips afin d'alimenter nos modèles et par conséquent de revoir l'agencement des annotations associées à chacune des vidéos. A partir de modèles et techniques existant dans la littérature nous avons développé plusieurs architectures tirant profit de méthodes nouvelles et a priori prometteuse pour la recherche en représentation de l'affect humain. D'une part nous utilisons les modèles en cascade apprenant séquentiellement les domaine spatial et temporel de la dynamique d'une expression faciale. Dans un deuxième temps, nous développons des modèles 3D-CNN modélisant simultanément le domaine spatio-temporel. Grâce à l'inflation nous arrivons à isoler relativement ces deux domaines afin d'analyser l'impact dominant ou non que peut avoir l'un sur l'autre. Enfin et dans le but d'optimiser nos résultats au vu de certains biais intrinsèques aux exemples d'entraînement, il est nécessaire d'extraire ces biais statistiques de nos modèles et de les injecter en post-traitement afin d'optimiser les performances. Dans le prochaine chapitre nous allons d'une part étudier plus en détail les ensemble de données utiliser pour la détection de l'émotion mais également développer les résultats expérimentaux obtenus avec les modèles présentés précédemment.

CHAPITRE 3

RÉSULTATS EXPÉRIMENTAUX

Dans cette partie, nous développons dans un premier temps les ensembles de données utilisés pour le pré-entraînement de nos modèles tels que VGG-FACE, RAF-DB et ImageNet ainsi que pour notre application finale, SEWA-DB. Nous détaillons ensuite les métriques de performances utilisées pour évaluer nos modèles. Nous opérons un «fine-tuning» avec l'ensemble de données RAF-DB sur différents modèles CNNs pré-entraînés avec VGG-FACE pour lesquels nous comparerons les performances. Ensuite, seront présentés en détail les conditions d'expérimentations et les performances des modèles CNN-LSTM et 3D-CNN développés. Enfin nous ferons l'analyse critique des résultats ainsi que leur comparaison avec la littérature.

3.1 Bases de données

3.1.1 Ensembles pour le pré-entraînement

Nous présentons ici les trois ensembles de données utilisés pour l'initialisation de nos modèles avant entraînement sur SEWA-DB.

- **VGG-Face :** L'ensemble de données récolté par Parkhi *et al.* (2015) a permis de développer des modèles adapté à la reconnaissance de visages mais de rendre également possible leur spécialisation à des tâches similaires comme la reconnaissance d'émotions. L'ensemble de données compte près de 2.6 millions d'images de 2 622 sujets différents. Grâce à la base de donnée Internet Movie Data Base (IMDB), une liste de noms de célébrités a été établie, et des images de visages ont été téléchargés à partir de Google Image Search. Quatre groupes de 50 annotateurs ont enfin labellisés chacune des images. L'annotation consistait à éliminer les images présentant une homonymie (même identité associée à deux personnes différentes) ou trop peu d'images correspondantes à une seule identité. Ainsi plusieurs mo-

dèles ont été entraînés à partir de ces données. Deux architectures type ont été retenues pour notre étude : VGG-16 (Parkhi *et al.*, 2015) et ResNet50 (Cao *et al.*, 2017) [1]

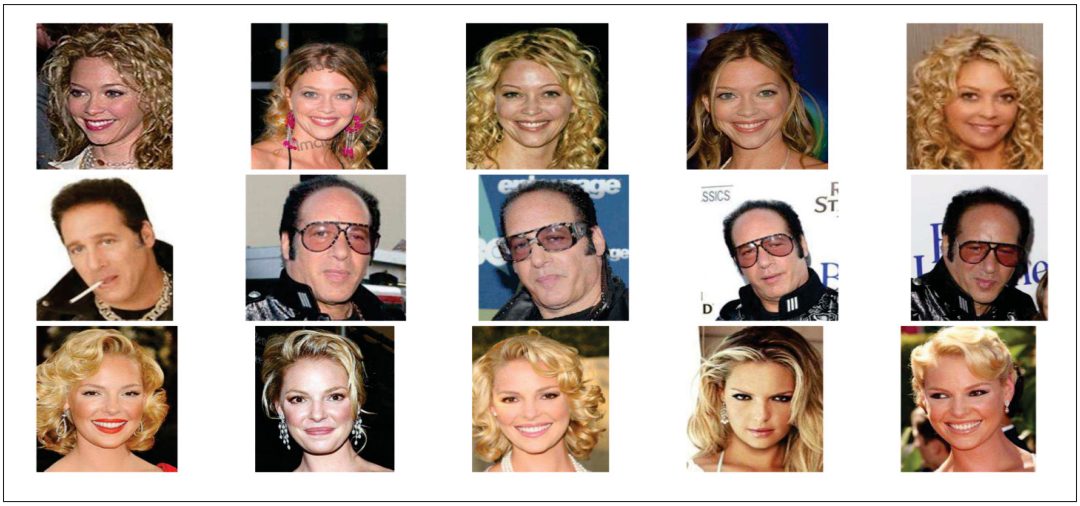

Figure 3.1 Exemples d'images de VGG-Face avec trois identités
Tirée de Parkhi *et al.* (2015)

- **ImageNet :** L'ensemble de données ImageNet (Deng *et al.*, 2009) réunit un million d'images d'objets de 1 000 catégories différentes. Cet ensemble de données est à l'origine d'une compétition (ImageNet Challenge, ILSVRC 2012) qui a permis le développement de modèles d'apprentissages automatique particulièrement performants. Ces modèles présentent une bonne capacité de généralisation pour des domaines variés, et aujourd'hui servent de bases d'apprentissage pour un grand nombre de tâches, grâce au transfert d'apprentissage. Notamment le premier vainqueur de la compétition, le modèle AlexNet (Krizhevsky *et al.*, 2017) a été le précurseur de l'apprentissage profond en démontrant pour la première fois les performances des CNNs (LeCun *et al.*, 1989), qui sont actuellement les modèles de références pour l'apprentissage profond.
- **RAF-DB :** RAF-DB (Li *et al.*, 2017) est une base de données créée pour la classification d'émotions et que nous utiliserons pour pré-entraîner nos modèles sur une tâche qui se rapporte le plus à notre application finale, la régression d'émotions sur le modèle du circumplex. RAF-DB est constitué de deux sous-ensembles de données. Celui que nous

1. http://www.robots.ox.ac.uk/~albanie/pytorch-models.html

utiliserons est annoté selon sept catégories basiques d'émotions (Joie, Tristesse, Surprise, Peur, Colère, Dégoût, Neutre), le second sous ensemble est quant à lui annoté selon 12 catégories d'émotions composées (chaque image est associée à deux catégories d'émotions). Au total la base de données compte 29 672 images, et le sous ensemble de sept catégories utilisé, 15 360 images. La Figure 3.2 donne quelques exemples d'images de l'ensemble de données. La première ligne correspond au sous ensemble d'émotions en catégorie quand les deux dernières lignes correspondent au sous ensemble d'émotions composées.

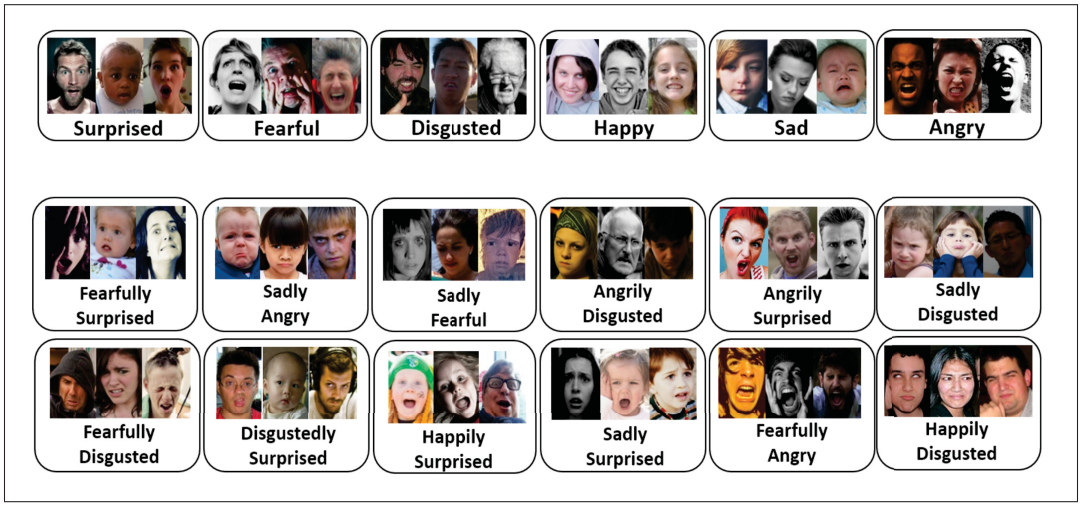

Figure 3.2 Exemples d'images de RAF-DB selon les deux sous-ensembles
Tirée de Li *et al.* (2017)

3.1.2 Ensemble de données SEWA-DB

La récolte de données de SEWA-DB a été réalisée dans le cadre d'une expérience impliquant 392 sujets. Afin de solliciter les émotions des candidats dans des conditions naturelles, chacun devait au préalable visionner quatre publicités suscitant différents caractères (violent, amusant, ennuyant, plaisir, intérêt). Après cette phase de préparation, les sujets regroupés par paire ont du discuter de chaque visionnage au travers d'une communication à distance. Leurs échanges ont été enregistrés à partir de leur ordinateur et d'une webcam personnelle. Chacun des sujets a été réparti en cinq groupes de classes d'âge allant de 18 ans à 60 ans et plus. Au total les 392

candidats ont permis de récolter 1057 minutes de données audio-visuelles. Cependant nous n'avons eu accès pour notre étude qu'à seulement 46 vidéos de sujets annotés. Les annotations ont inclus à la fois les valeurs de valence et excitation du circumplex (valeurs continues comprises entre -1 et +1). Les valeurs de plaisir («liking») ont également été annotées mais dû au faible résultats obtenus par l'étude des auteurs, nous avons décidé de ne pas étudier ces valeurs.

D'après l'étude de Kossaifi *et al.* (2019), la régression des valeurs de valence et excitation a été réalisée à partir de deux types de caractéristiques vidéos : d'une part les caractéristiques artisanales, «Dense SIFT», et d'autres part des caractéristiques extraites par apprentissage profond avec une architecture de type ResNet18.

- «Dense SIFT» est une méthode d'extraction de caractéristiques artisanales qui étudie selon une certaine fenêtre les pixels environnants les points particuliers du visage (ou «facial landmarks»). À partir de ces caractéristiques trois types de modèles ont été utilisés pour la prédiction des séquences vidéos : SVR, Random Forest et LSTM. Leurs expériences ont consisté à étudier les performances des modèles à partir de différents groupes ethniques. En effet, les chercheurs ont entraîné les modèles en n'étudiant qu'une ethnie particulière parmi six ou en ne faisant aucune différence sur les ethnies (expérience multi-ethnique).
- dans le cas de l'architecture profonde ResNet18, les caractéristiques ont été relevé par apprentissage automatique selon deux critères d'optimisation (ou fonction de coût) : Root Mean Squared Error (RMSE) et Concordance Correlation Coefficient (CCC). En conclusion, la régression par apprentissage profond de caractéristiques s'est montrée plus performante qu'avec les caractéristiques artisanales. De plus, il s'est avéré que l'apprentissage avec CCC a eu des résultats plus satisfaisants qu'avec RMSE. Ainsi l'état de l'art retenu pour l'ensemble de données SEWA-DB correspond au modèle ResNet18 entraîné avec CCC comme fonction de coût.

C'est à ce dernier cas que nous avons comparé notre étude. Le Tableau 3.1 détaille les résultats de cette étude, constituant l'état de l'art relatif à l'ensemble de donnée SEWA-DB.

Tableau 3.1 État de l'art pour l'ensemble de données SEWA-DB à partir de caractéristiques vidéos d'expressions faciales
Tiré de Kossaifi *et al.* (2019)

Modèle	Valence		excitation	
	PCC	CCC	PCC	CCC
SVR	0.321	0.312	0.182	0.202
RF	0.268	0.207	0.181	0.123
LSTM	0.322	0.281	0.173	0.115
ResNet18 (RMSE)	0.29	0.27	0.13	0.11
ResNet18 (CCC)	**0.35**	**0.35**	**0.35**	**0.29**

3.2 Métriques de performance

À partir des prédictions de valence et excitation associées à chacune des séquences, plusieurs métriques de références nous permettront d'évaluer les performances et la pertinence de nos modèles.

3.2.1 Coefficient de corrélation de Pearson

Le coefficient de corrélation de Pearson (PCC - Pearson Correlation Coefficient) défini une mesure de la covariation linéaire existant entre deux distributions de données. En termes mathématiques, il est défini par la covariance de variables, divisé par le produit de leur déviation standard. Les valeurs de ce coefficient s'étendent sur l'ensemble [-1 ; 1]. Plusieurs interprétations peuvent suivre à partir de l'étendu de cet ensemble. En supposant deux ensembles de données x et y. Une valeur de +1 signifie que les valeurs des deux distributions sont en phase, à mesure que les valeurs de y augmentent, les valeurs de x augmentent, et parfaitement égales. Une valeur de -1 signifie que les valeurs des deux distributions sont en opposition de phase, à mesure que les valeurs de y diminue, les valeurs de x augmentent. Une valeur de 0 signifie qu'aucune corrélation n'existe entre les deux distributions

Le PCC pour deux ensembles de données x et y est usuellement donné par :

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}} \tag{3.1}$$

où n est le nombre d'échantillons, x_i sont les échantillons individuels, y_i sont les annotations, et $\overline{x}$ est la moyenne des échantillons, de même pour $\overline{y}$.

3.2.2 Coefficient de corrélation de Lin

Le coefficient de corrélation de Lin (CCC - Concordance Correlation Coefficient) est similaire au PCC, et leurs valeurs sont mathématiquement liées. Cependant le CCC permet de définir un degré de correspondance entre deux distributions. Le CCC à la différence du PCC établi une relation entre les deux distributions à la fois en se basant sur la covariance et la correspondance, ce qui en fait une mesure plus précise sur la corrélation entre deux distributions.

Le CCC entre deux ensembles de données x et y est donné par :

$$\rho_{xy} = \frac{2s_{xy}}{s_x^2 + s_y^2 + (\overline{x} - \overline{y})^2} \tag{3.2}$$

où s_{xy} est la covariance entre x et y donné par :

$$\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y}) \tag{3.3}$$

et s_x^2 est la variance de x, donné par :

$$\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2 \tag{3.4}$$

de même pour s_y^2.

3.2.3 Erreur Absolue Moyenne (MAE) & Pourcentage d'Erreur Absolue Moyenne (PEAM)

Nous nous servons de l'erreur moyenne absolue pour évaluer la distance moyenne entre les prédictions et les annotations. De plus, comme nous nous servons de différentes échelles de valeurs de valence et excitation, notamment par multiplication des annotations, mais aussi par normalisation lors du post-traitement, il est intéressant d'observer le pourcentage d'erreur absolue moyenne entre prédictions et annotations. De cette manière, nous compensons la perte d'interprétation de l'erreur moyenne absolue.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i| \tag{3.5}$$

$$PEAM = \frac{100\%}{n}\sum_{i=1}^{n}|\frac{y_i - x_i}{y_i}| \tag{3.6}$$

3.3 Performances des modèles 2D sur RAF-DB

Le Tableau 3.2 détaille les performances des modèles pré-entraînés avec l'ensemble de données RAF-DB. Dans chacun des cas nous indiquons si le modèle a été entraîné sur l'intégralité de ses couches de convolutions ou si une partie de ces couches ont été fixées pour ne pas être modifiées et conserver leurs paramètres initialisés soit avec ImageNet ou VGG-Face. Nous avons alors indiqué quel bloc de convolution a servi de limite à l'optimisation des couches. Les couches précédant ce bloc sont non-optimisables. Nous avons variés de nombreux paramètres dans chacun des cas :

- **taille de batch** : 16, 32
- **utilisation ou non d'augmentation de données**

- **type de pooling avant régression** : moyenne, maximum, ou flatening. Choisir adéquatement la méthode de réduction des vecteurs caractéristiques avant classification peut avoir un impact significatif sur les performances des modèles. Nous avons donc évaluer trois méthodes traditionnelles de pooling : «Average Pooling» et «Maximum Pooling», «Flattening» ne réduit pas la dimension des vecteurs mais applani la représentation en un vecteur à une dimension.

De plus nous avons indiqué comme référence pour l'état de l'art sur RAF-DB, les travaux de Jyoti & Dhall (2018). Les chercheurs ont utilisé plusieurs types de réseaux à convolutions ainsi que pour un nombre variable de classes d'émotions. Le modèle le plus performant utilisé pour sept classes d'émotions (dont les résultats sont indiqués dans le Tableau 3.1 est un modèle CNN composé de quatre couches de convolutions et trois couches densément connectées. Des couches de «Maximum Pooling» ont été insérées entre les couches de convolutions et de dropout entre les couches densément connectées afin d'éviter le sur-apprentissage. Globalement les modèles CNNs que nous avons développés performent de manière similaire au modèle des chercheurs néanmoins notre modèle VGG-16, pré-entraîné avec VGG-FACE surpasse de 1.7% leur meilleur modèle.

Plusieurs paramètres se sont montrés important afin d'améliorer les performances. L'augmentation de données a pu améliorer de 2% la précision moyenne des modèles, par ailleurs l'augmentation des données est d'autant plus efficace sur de petits batchs. En moyenne entraîner intégralement les modèles avec des batchs de dimension 16 avec augmentation de données et «AveragePooling» avant régression se sont montrés plus performants que les autres combinaisons.

Ainsi nous avons évalué plusieurs méthodes de spécialisation des modèles sur RAF-DB. Globalement les modèles performent de manière similaire à l'état de l'art et présentent un taux plutôt satisfaisant de précision. Étant donné que nous pratiquerons l'expansion des paramètres de ces architectures, nous avons préféré garder les modèles intégralement entraînés avec RAF-DB, de manière à ce que toutes les couches de convolutions soient spécialisées pour la clas-

sification d'émotions. Par ailleurs, nous avons suggéré a priori que les modèles performant le mieux sur RAF-DB permettraient d'obtenir de meilleurs résultats sur des tâches de régression de l'émotion.

Tableau 3.2 Pré-entraînement de différents modèles avec l'ensemble de données RAF-DB

Modèle (Initialisation)	**Bloc de Convolution**	**Précision**
(Jyoti & Dhall, 2018)	NA	78.2%
VGG-11 (ImageNet)	Intégral	75.6%
	Conv_2_1	75.9%
	Conv_3_1	75.9%
	Conv_4_1	75.9%
	Conv_5_1	70.3%
VGG-11-BN (ImageNet)	Intégral	77.8%
VGG-16 (VGG-Face)	Intégral	78.5%
	Conv_2_1	78.5%
	Conv_3_1	79.1%
	Conv_4_1	**79.9%**
	Conv_5_1	74.4%
VGG-16-BN (VGG-Face)	Intégral	78.4%
ResNet50 (VGG-Face)	Intégral	79.7%
	Conv_4_1	78.0%
	Conv_5_1	65.1%

NA : Non Acquis.

3.4 Modèles CNN-LSTM

3.4.1 Paramètres expérimentaux

D'après notre méthode, nous avons proposé en première approche de développer une architecture en cascade de type CNN-LSTM afin d'avoir un aperçu du potentiel de modèles de régressions avec l'ensemble de données SEWA-DB. Le protocole d'apprentissage des modèles se base sur une architecture VGG-16 associée à un LSTM à une couche pour la caractérisation spatio-temporelle des séquences vidéos. Plusieurs paramètres, inhérents à l'ensemble de

données ainsi qu'à l'architecture, nous ont fournis une grande variabilité d'expériences et de différencier leurs performances.

Paramètres variables inhérents aux données :

- **type d'initialisation** : Le CNN (VGG-16) est dans un premier cas initialisé uniquement avec VGG-Face (aka VGG-16 [VGG-Face]) et dans un second cas à la fois initialisé avec VGG-Face *et* optimisé avec RAF-DB (aka VGG-16 [RAF]).
- **taux fps** : Les séquences sont constituées de trames extraites à un taux de 10 ou 50 fps
- **longueur de Séquence** : Les séquences sont longues de 16 ou 64 trames
- **taux de recouvrement** : Les séquences consécutives d'une même vidéo se recouvrent avec un taux de recouvrement de 20% ou de 80%. On pourra trouver dans le Tableau 3.3 les meilleurs configurations suivant chacun des modèles pour lesquelles le taux de recouvrement est spécifié.
- **mode de fusion des annotations** : Comme les annotations de l'ensemble de données concernent initialement chacune des trames des vidéos. La restructuration de l'ensemble de données en séquences de trames nous conduit à fusionner les annotations d'une même séquence afin d'obtenir un unique label. Pour ce faire, les modes de fusion choisis sont le maximum, la moyenne, ou l'extremum.

Autres paramètres inhérents à l'architecture :

L'entraînement du modèle CNN-LSTM a nécessité des mini-batchs de huit séquences vidéos, le bloc LSTM a utilisé un dropout récurrent de 20% et un dropout en sortie de 10%, afin de prévenir le sur-apprentissage.

3.4.2 Performances des modèles

Le Tableau 3.3 décrit les meilleurs résultats obtenus pour chacun des modèles testés selon les paramètres évoqués précedemment et avec post-traitement, du moment que le post-traitement a eu un effet bénéfique sur nos résultats comme nous le verrons ci-après.

Tableau 3.3 Performances de l'architecture CNN-LSTM sur SEWA-DB

Modèle [Initialisation] (fps)	**Label**	**Configuration (L, TR, MF)**	**PCC**	**CCC**
VGG-16 [VGG-Face] (10)	Valence	16, 0.2, extremum	0.590	0.560
	excitation	16, 0.2, moyenne	0.549	0.542
VGG-16 [VGG-Face] (50)	Valence	64, 0.2, moyenne	0.541	0.511
	excitation	64, 0.2, extremum	0.495	0.492
VGG-16 [RAF] (10)	Valence	64, 0.8, moyenne	**0.631**	**0.625**
	excitation	64, 0.8, extremum	**0.558**	**0.557**
VGG-16 [RAF] (50)	Valence	64, 0.2, moyenne	0.582	0.568
	excitation	64, 0.2, extremum	0.517	0.517

L : Longueur de séquence en trames, TR : Taux de Recouvrement en %, MF : Mode de fusion des annotations

D'après les résultats obtenus, utiliser l'extremum ou la moyenne comme méthode de fusion des annotations peuvent apporter l'un comme l'autre de bonnes performances. En général le taux de recouvrement impact faiblement les résultats si ce n'est d'augmenter le nombre de séquences disponibles avec l'augmentation du taux de recouvrement. Ce qui est plutôt bénéfique car cela permet d'obtenir plus d'exemples d'entraînement pour les modèles. De plus utiliser des séquences suffisamment longues (64 trames) s'avère plus efficace. Utiliser 64 trames, revient avec un taux 10 fps à étudier des séquences de durée 6.4s. Par ailleurs, les meilleures performances ont été obtenues avec le modèle VGG-16 initialisé séquentiellement avec VGG-Face et RAF-DB. Ce qui prouve que la spécialisation avec une tâche de classification de l'émotion a été bénéfique pour la régression. Enfin, les valeurs en PCC et CCC obtenues sont satisfaisantes, les meilleures architectures atteignent un PCC moyen de 0.6, ce qui montre une bonne corrélation entre prédictions et annotations. En moyenne, le pourcentage d'erreur absolue moyenne (PEAM) associé à chaque prédiction est de 7.5%. Le PEAM représente, en d'autres termes la

marge d'erreur que les prédictions des modèles CNN-LSTM développé ont sur les annotations réelles.

Bénéfice du post-traitement

Nous avons comparé les résultats avant et après post-traitement afin d'évaluer les bénéfices sur les valeurs de PCC et CCC. Ainsi sur l'ensemble des modèles testés, en moyenne le PCC a évolué de +0.07, et le CCC a évolué de +0.1. En conséquences, la normalisation, le filtrage de la moyenne ainsi que l'ajout d'un délai ont permis séquentiellement d'obtenir une amélioration notable. Le Tableau 3.3 montre l'évolution en PCC et CCC suivant les valeurs de valence, excitation et chaque catégorie de modèles. On remarque que les meilleurs modèles (VGG-Face pré-entraîné avec RAF-DB en 10 fps) sont ceux qui ont le moins bénéficié du post-traitement. Ceci prouve que les distributions obtenues après entraînement ont pleinement exploités leur potentiel de détection de l'émotion. Les Figures 3.3 et 3.4 montrent graphiquement l'amélioration des valeurs de PCC et CCC sur l'ensemble des 128 modèles CNN-LSTM développés après post-traitement.

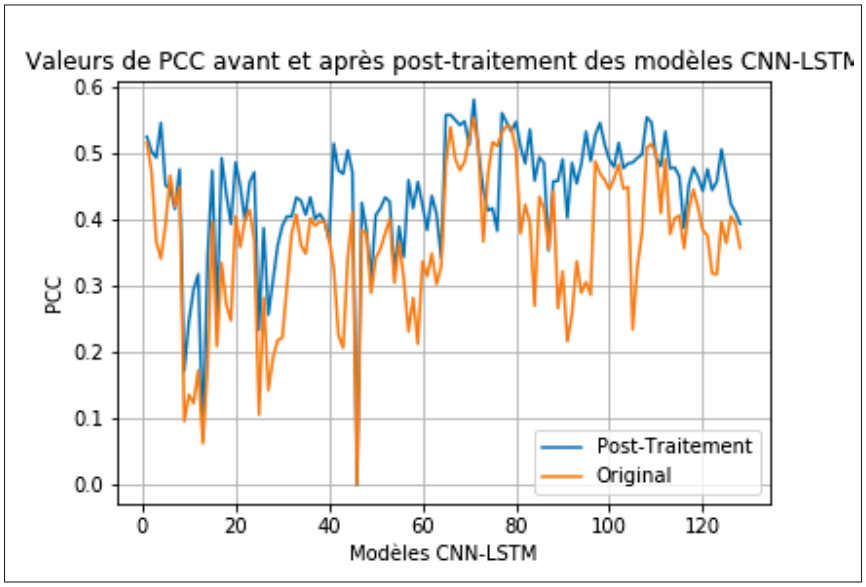

Figure 3.3 Valeurs de PCC avant et après post-traitement des modèles CNN-LSTM

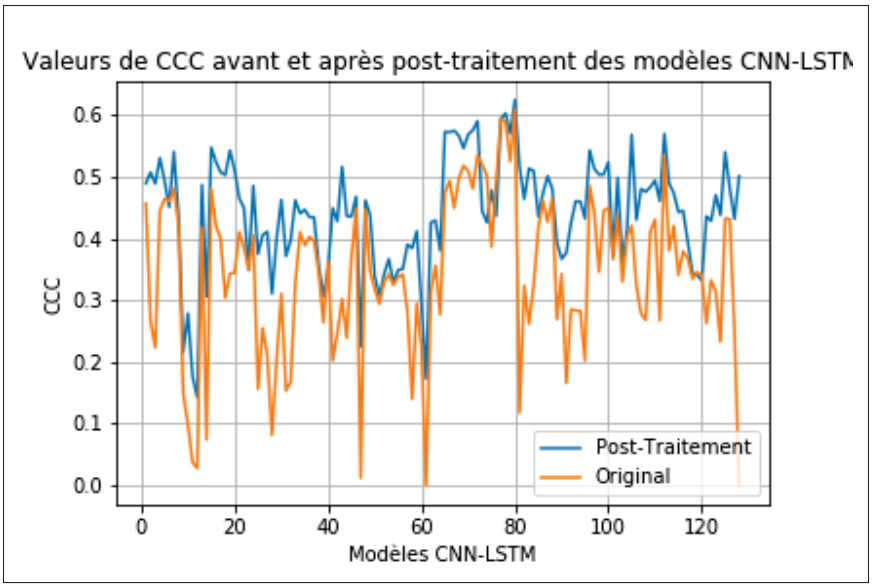

Figure 3.4 Valeurs de CCC avant et après post-traitement des modèles CNN-LSTM

Analyse graphique des prédictions et des annotations

Afin de vérifier la pertinence de nos modèles, il est essentiel d'observer les valeurs des prédictions et inspecter la cohérence des performances avec les prédictions. Les Figures 3.5 et 3.6 donnent graphiquement les prédictions et les annotations de valence sur un échantillon de 500 séquences sur le meilleur modèle développé (VGG-Face pré-entraîné avec RAF-DB en 10 fps).

3.4.3 Conclusion Préliminaire

En résumé, nous avons montré précédemment comment certains paramètres inhérent aux architectures et à la structure des sources de données peuvent influencer la performance de modèle CNN-LSTM. Dans un premier temps nous avons montré que l'initialisation des modèles VGG-16 avec RAF-DB ont été les plus performants. La fusion des annotations s'est avérée satisfaisante en choisissant les extremum ou les moyennes des annotations des trames sur une séquence. De plus, les séquences de trames doivent être suffisamment longues afin de pouvoir contenir le plus d'informations sur l'expression d'une unique émotion. À 10 fps, 64 trames se sont montrées plus performantes que 16. Et enfin les étapes de post-traitement ont eu un

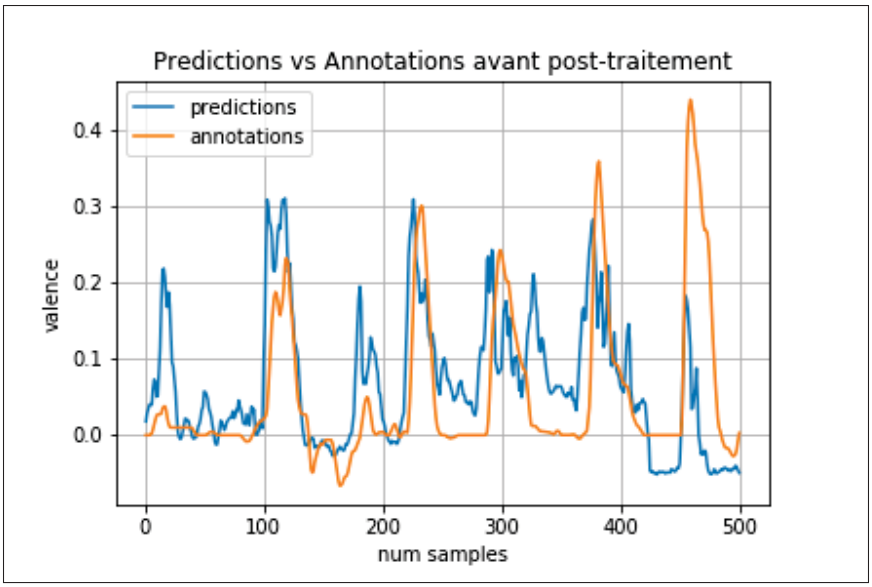

Figure 3.5 Exemple de prédictions et annotations sur un échantillon de 500 séquences sans post-traitement

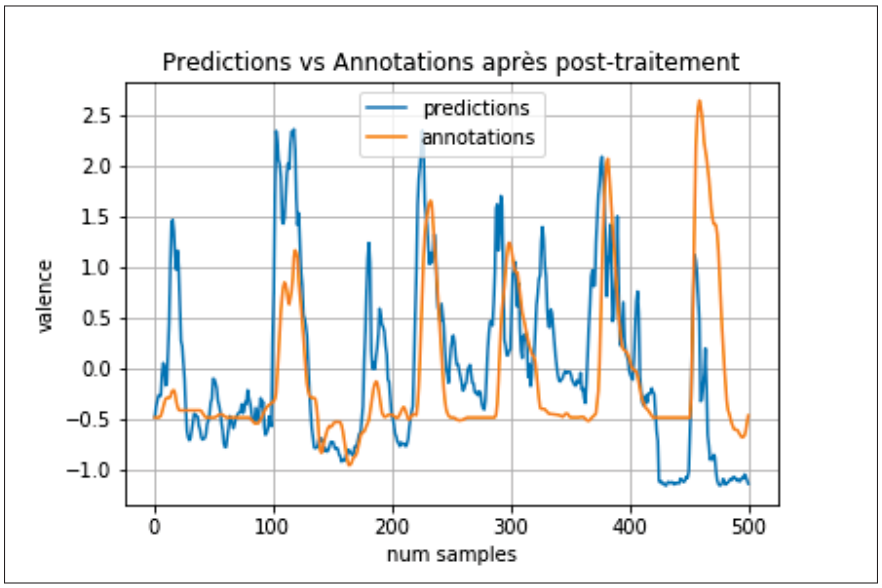

Figure 3.6 Exemple de prédictions et annotations sur un échantillon de 500 séquences avec post-traitement

bénéfice notable dans l'amélioration de la corrélation entre prédictions des modèles et annotations réelles. Il est important de noter que notre méthodologie nous a permis de surpasser l'état de l'art, en doublant les valeurs de PCC et CCC. Pour la valence, à partir d'un PCC et CCC

respectivement de 0.35 et 0.35 nous avons atteint des valeurs de 0.63 et 0.62. De même pour l'excitation, à partir d'un PCC et CCC respectivement de 0.35 et 0.29 nous avons atteint des valeurs de 0.558 et 0.557. Dans la prochaine section, nous étudierons les modèles basés sur 3D-CNN.

3.5 Modèles 3D-CNN

À partir des modèles 2D-CNN pré-entraînés avec VGG-Face, RAF-DB et ImageNet, et de la méthode d'expansion des poids (Carreira & Zisserman, 2017) nous avons étudié les performances de détection de modèles 3D-CNN. Nous allons tout d'abord donner les paramètres d'études, puis donner les meilleurs performances obtenus pour chaque type de modèles développés (VGG-11, VGG-16, ResNet50), et ce avec les mêmes étapes de post-traitement utilisées avec CNN-LSTM. Nous étudierons enfin plus en détail l'impact de la variation de certains paramètres. À titre d'exemples et afin de vérifier la pertinence d'entraînement des modèles, nous avons également étudié les activations de certains filtres de convolutions.

3.5.1 Paramètre d'études

À la différence des modèles CNN-LSTM, les modèles 3D-CNN présentent beaucoup plus de paramètres à optimiser au cours de l'apprentissage, trois fois supérieur en moyenne, de plus les données d'entrées sont des séquences d'images, ce qui complexifie les capacités mémoires des unités de calcul et allonge la durée d'entraînement des modèles. Il a donc fallu restreindre nos choix quant à certains paramètres d'études. Ainsi les prochains résultats sont associés aux paramètres fixes suivants : les trames de séquences sont relevées à 10 fps, les séquences sont constituées de 64 trames avec un taux de recouvrement de 80%, l'entraînement des modèles est effectué par batch de huit séquences, l'optimisation est réalisée avec Adam avec un taux d'apprentissage de $1e^{-5}$, la fonction de coût utilisée est l'erreur quadratique moyenne. Enfin la mesure de performance des modèles au cours de l'apprentissage est le CCC. Les paramètres variables sont décrits dans le Tableau 3.4.

Tableau 3.4 Paramètres variables des architectures 3D-CNN et valeurs possibles

Paramètres	**C1**	**C2**
Inflation	Centrage	Copie
Dilution	Bloc1 : 1 Bloc2 : 1 Bloc3 : 1 Bloc4 : 1	Bloc1 : 1 Bloc2 : 2 Bloc3 : 4 Bloc4 : 8
Masking	Sans	Avec
Initialisation des poids	Aléatoire	Zéro
Multiplicateur	x1	x100

Nous avons envisagé l'utilisation du CCC comme fonction de coût plutôt que le MSE, cependant comme l'optimisation des modèles avec cette fonction nécessite d'observer l'ensemble de données au complet avant la mise à jour des paramètres plutôt que par mini-batch, l'entraînement en serait d'autant plus difficile et rendrait nos expériences trop complexes. Ainsi il a été plus raisonnable d'optimiser avec le MSE tout en utilisant le CCC comme métrique d'apprentissage. L'avantage est que l'amélioration de le MSE en validation, assure également un meilleur PCC et CCC. En effet réduire la distance entre prédictions et annotations, amène les deux distributions à s'assimiler, autrement dit à rapprocher leur variance.

3.5.2 Performances générales des architectures

La Tableau 3.4 regroupe les meilleurs résultats obtenus pour chaque type de modèles. Enfin pour plus de clarté, les Figures 3.7 et 3.8 comparent graphiquement les valeurs de PCC et CCC suivant la valence et excitation de chacun des modèles.

Premièrement, il revient que les résultats sont bien moins performants que le modèle CNN-LSTM précédent. Au mieux les modèles ont des performances en PCC et CCC comprises entre 0.3 et 0.4, ce qui démontre une corrélation assez faible. Par comparaison graphique des performances des modèles, nous observons que les valeurs d'excitation ont globalement été mieux détectées que la valence. L'ajout de couches de batch normalisation a été particulièrement bénéfique, en effet, l'architecture de type VGG-16 présente les plus faibles résultats

comparés aux autres architectures de type VGG-11, VGG-16, et Resnet50 implémentant la batch-normalisation.

Concernant la variation du type d'initialisation de chacune des architectures, et contre toute attentes, pour les architectures de type VGG-16, le pré-entraînement avec ImageNet a montré de bien meilleurs résultats qu'avec RAF-DB et VGG-Face. Au contraire ResNet50 a plus bénéficié d'un pré-entraînement avec VGG-Face et VGG-11 avec RAF-DB. Ainsi, il en résulte que les architectures VGG-16-BN pré-entraînée avec ImageNet et ResNet50 pré-entraîné avec VGG-Face, ont été les modèles les plus performants.

3.5.3 Performances détaillées selon certains paramètres d'études

Dans ce dernier paragraphe, nous cherchons à montrer l'influence des paramètres d'études cités précédemment. Il est important de noter que considérant la faiblesse des résultats, il est difficile d'assurer l'effet de la variation de ces paramètres. Pour rappel nous avons chercher à faire varier cinq paramètres inhérents aux données d'entrée ainsi qu'aux architectures (inflation, dilution, masking, initialisation des poids centrés, multiplicateur). Il résulte d'après l'étude, qu'aucune variations de ces paramètres n'a pu entraîner une amélioration particulière des performances de chacun des modèles. Chaque architecture ayant montré des performances différentes suivant une même configuration de ces paramètres. Le Tableau 3.6 répertorie les configurations associées aux meilleures performances de chacune des architectures. D'après l'étude de la variation des configurations selon les différentes architectures, il apparaît que majoritairement les architectures ont plutôt bénéficié d'un centrage des poids lors de l'inflation, de plus dans le cas de poids centrés initialiser les poids des autres dimensions temporelles à zéro plutôt que les choisir aléatoire s'avère plus bénéfique statistiquement. Malheureusement, dans le cadre de cette étude, il n'est pas possible de statuer quant à la multiplication des annotations par 100, appliquer le principe de dilution, ou appliquer un masque pour l'optimisation partielle des couches de convolutions. La variation de tous ces paramètres a bénéficié indépendamment à chacune des architectures et a permis spécifiquement au modèle d'améliorer les performances au cours de l'apprentissage.

Tableau 3.5 Meilleures performances obtenues sur SEWA-DB selon les architectures testées et leur initialisation avec différents ensembles de données

Modèle	**Initialisation**	**Valence**		**excitation**	
		PCC	CCC	PCC	CCC
VGG-11-BN	RAF-DB	0.035	0.018	0.359	0.348
	ImageNet	0.04	0.025	0.342	0.203
VGG-16	VGG-Face	0.119	0.071	0.22	0.166
	RAF-DB	0.036	0.028	0.242	0.119
	ImageNet	0.209	0.19	0.391	0.189
VGG-16-BN	VGG-Face	0.203	0.105	0.347	0.304
	RAF-DB	0.123	0.101	0.284	0.165
	ImageNet	**0.346**	**0.304**	**0.382**	**0.326**
ResNet50	**VGG-Face**	**0.313**	**0.253**	**0.406**	**0.273**
	RAF-DB	0.113	0.063	0.262	0.207
	ImageNet	0.137	0.135	0.323	0.256

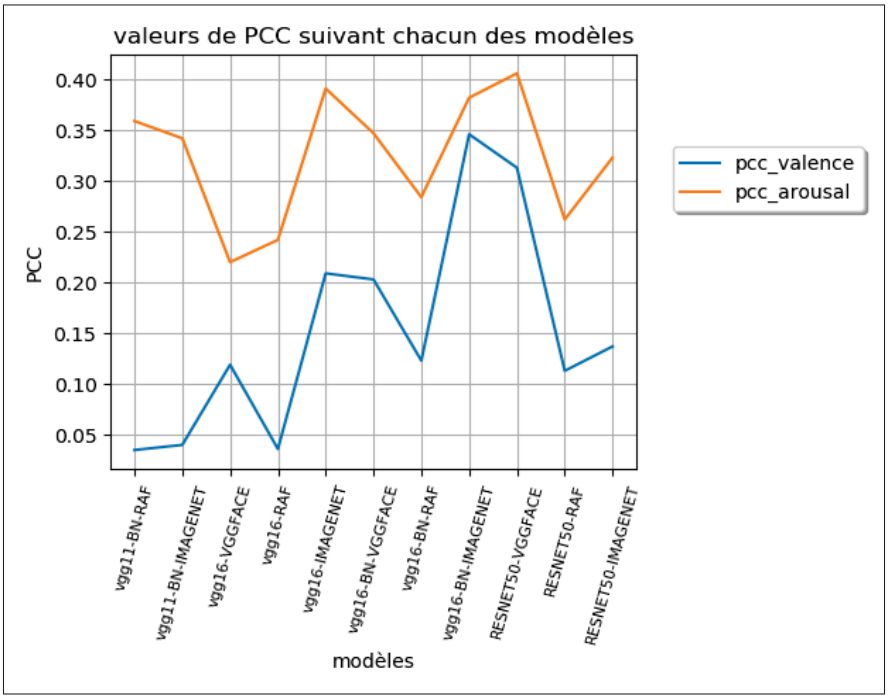

Figure 3.7 Évolution des valeurs de PCC sur les modèles les plus performants de chacune des architectures développées

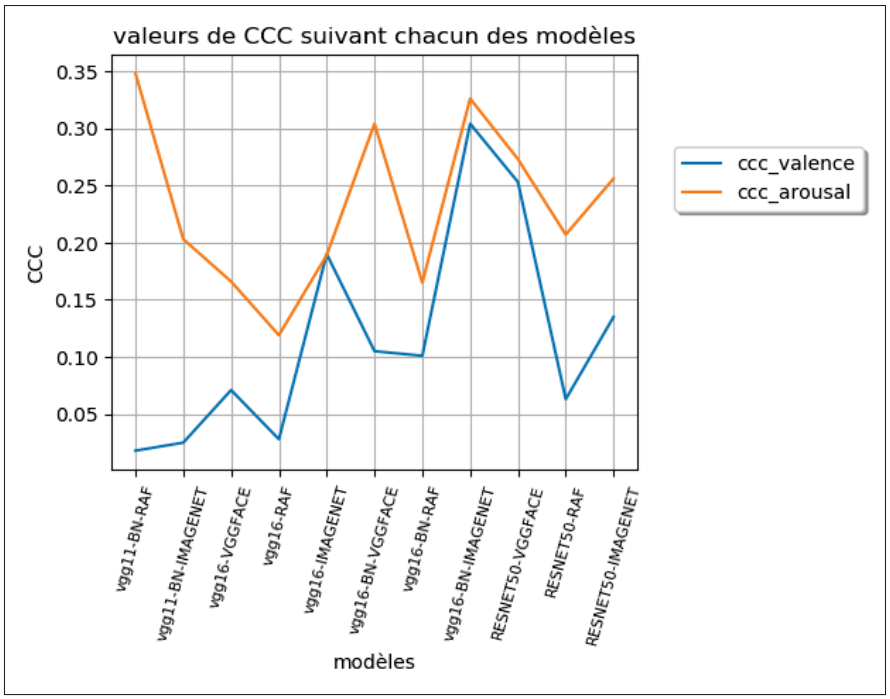

Figure 3.8 Évolution des valeurs de CCC sur les modèles les plus performants de chacune des architectures développées

Tableau 3.6 Configurations des modèles selon les meilleures performances obtenues sur valence

Modèles	**Initialisation**	**Paramètres**				
		Inflation	**Dilution**	**Masking**	**Init. des poids centrés**	**Mult.**
VGG-11-BN	RAF-DB	Centrés	I	Sans	zero	1
	ImagNet	Copiés	I	Sans	zero	100
VGG-16	VGG-Face	Centrés	I	Sans	aléatoire	1
	RAF-DB	Copiés	I	Sans	aléatoire	1
	ImagNet	Centrés	I	Sans	aléatoire	1
VGG-16-BN	VGG-Face	Centrés	I	Sans	aléatoire	1
	RAF-DB	Copiés	I	Avec	aléatoire	1
	ImagNet	Copiés	I	Sans	aléatoire	1
ResNet50	VGG-Face	Centrés	I	Avec	zero	100
	RAF-DB	Copiés	VIII	Sans	zero	1
	ImagNet	Centrés	VIII	Avec	zero	1

Tableau 3.7 Configurations des modèles selon les meilleures performances obtenues sur excitation

Modèles	Initialisation	Paramètres				
		Inflation	Dilution	Masking	Init. des poids centrés	Mult.
VGG-11-BN	RAF-DB	Centrés	I	Sans	aléatoire	1
	ImagNet	Centrés	VIII	Sans	zero	1
VGG-16	VGG-Face	Centrés	I	Sans	aléatoire	1
	RAF-DB	Copiés	I	Avec	aléatoire	100
	ImagNet	Centrés	I	Sans	aléatoire	1
VGG-16-BN	VGG-Face	Centrés	I	Avec	aléatoire	1
	RAF-DB	Copiés	I	Sans	aléatoire	1
	ImagNet	Centrés	I	Sans	zéro	1
ResNet50	VGG-Face	Copiés	I	Avec	zero	100
	RAF-DB	Centrés	VIII	Avec	zero	1
	ImagNet	Centrés	I	Sans	zero	1

3.5.4 Analyse graphique des filtres de convolutions

À titre d'exemples, nous avons collectés les activations des filtres en sortie des couches de convolutions pour les modèles 3D-CNN les plus performants. L'affichage des activations consiste à observer les différentes représentations filtrées des séquences d'images. Les activations modifient les images de manière à isoler certaines zones du visages à «forte température». Les zones les plus éclairées correspondent aux points d'intérêt de l'image filtrée. Les neurones artificiels associés à ces zones de «forte chaleur» auront ainsi plus tendance à être activés et se modifier pour donner de l'importance à ces caractéristiques du visage. Les Figures 3.9 et 3.10 montrent respectivement les activations pour une image de l'ensemble d'entraînement et une image de l'ensemble de test. Les activations représentées correspondent à l'ensemble des filtres de la couche de convolution «conv_2_2» d'un VGG-16-BN pré-entraîné avec ImageNet. À ce niveau de convolutions, les visages peuvent être encore distingués et afficher leurs points d'intérêts. Les zones d'activations, dites de «fortes chaleurs» auront tendance à tirer vers le jaune alors que les zones moins activées tourneront vers le violet. On a représenté pour l'exemple d'entraînement et de test, 121 filtres d'une seule image d'une séquence vidéo. Après observations des activations, à la fois pour l'image d'entraînement et de test, il revient que les parties les plus expressives du visage sont activées, c'est à dire les contours des yeux, les sourcils, le

nez et la bouche sont la plupart du temps particulièrement éclairés. Ceci prouve que globalement les modèles arrivent à extraire correctement l'aspect texturale des images et permet ainsi d'extraire des caractéristiques cohérentes de l'émotion.

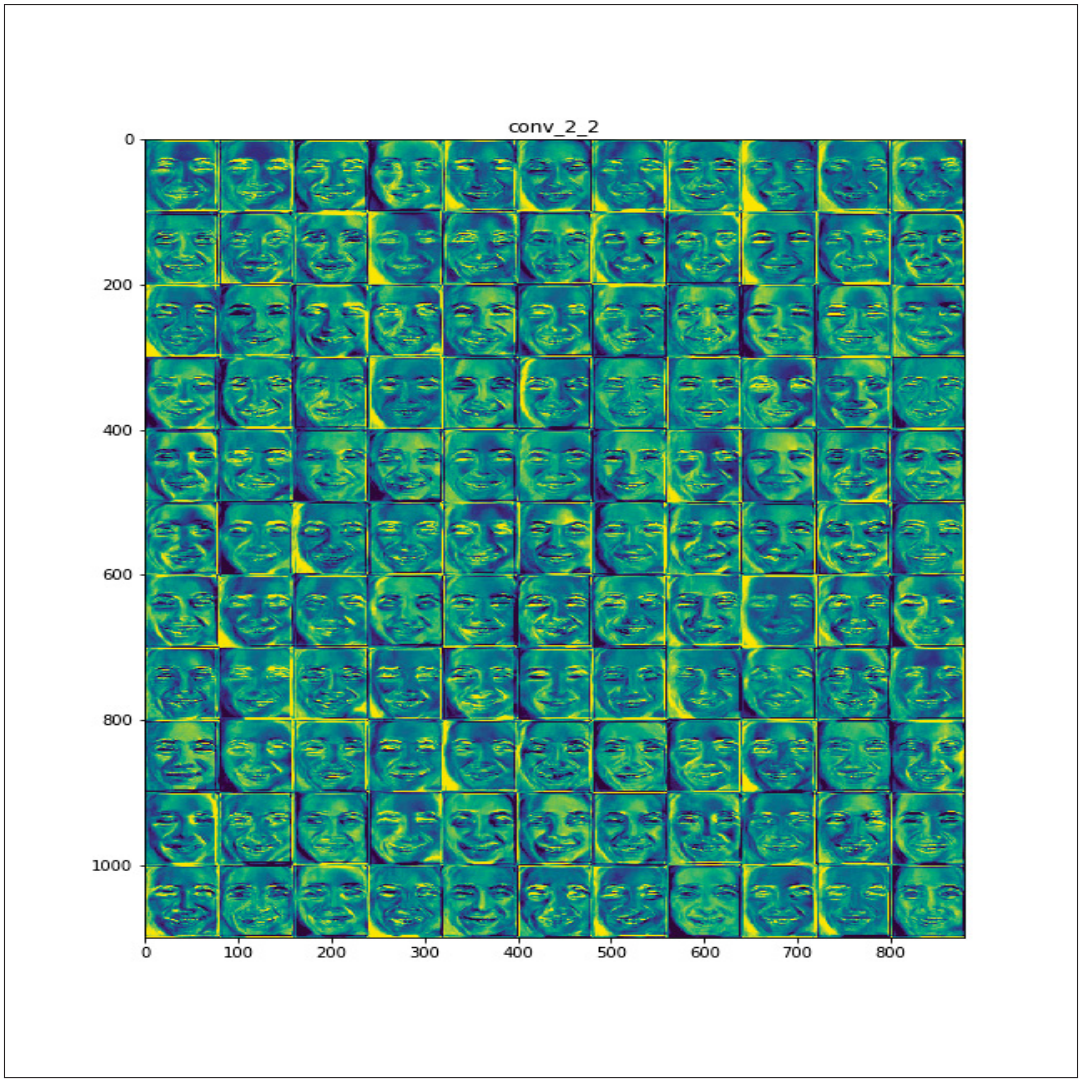

Figure 3.9 Activations d'une couche intermédiaire de convolutions sur une architecture de type VGG-16

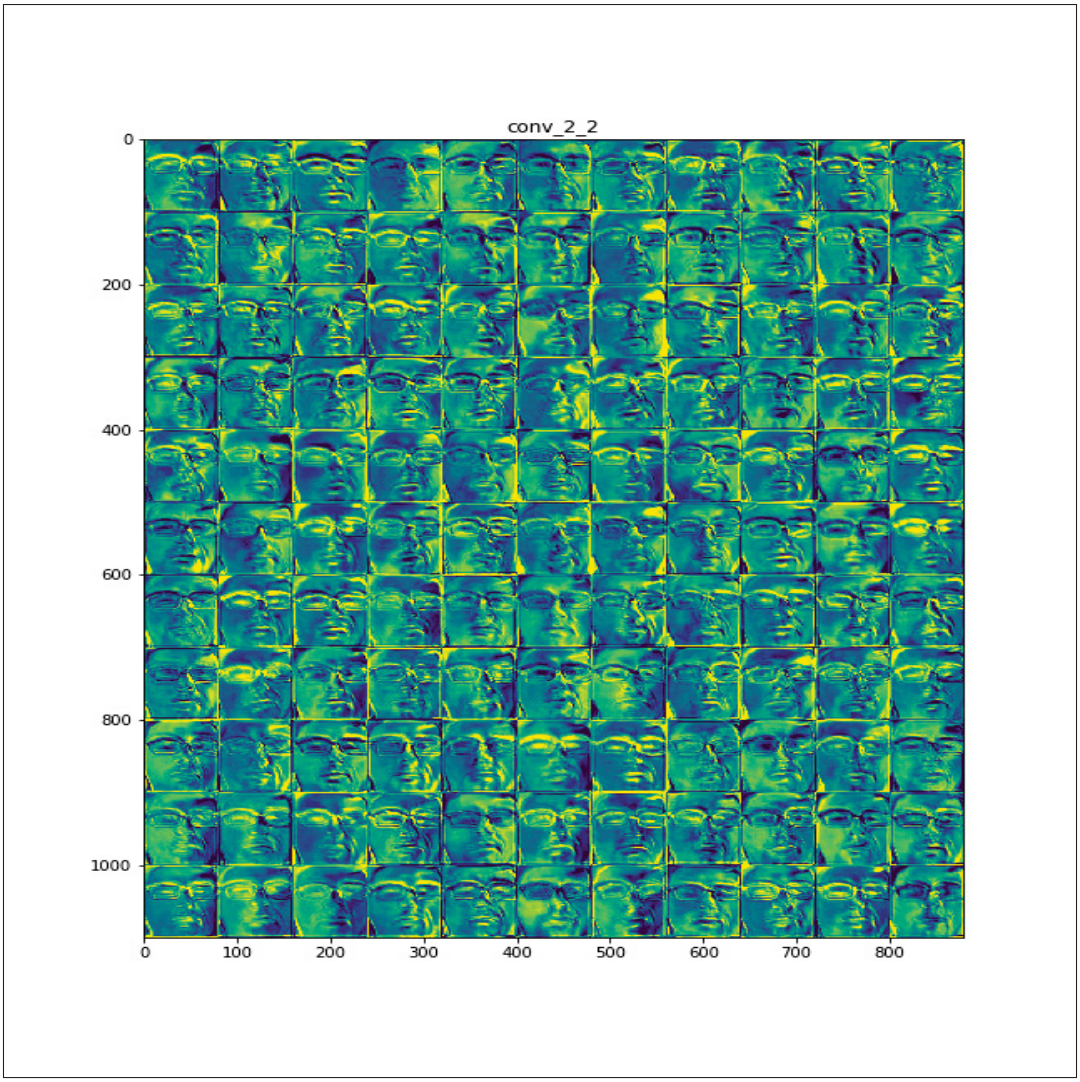

Figure 3.10 Activations d'une couche intermédiaire de convolutions sur une architecture de type VGG-16

3.6 Analyse critique des résultats et comparaison avec la littérature

Dans cette étude expérimentale, nous avons d'abord développer un modèle en cascade de type CNN-LSTM qui a montré de meilleurs résultats qu'avec les modèles 3D-CNN, quelqu'ait été la configuration des paramètres et l'initialisation de la distribution de départ des architectures. (Kossaifi *et al.*, 2019), dont l'étude avec l'ensemble de données SEWA-DB nous sert de référence, rappelle que dans la littérature les valeurs de valence sont plutôt bien détectée avec des caractéristiques de type vidéo, alors que l'excitation est mieux détecté par l'audio. Dans

notre étude, le CNN-LSTM a mieux performé sur valence qu'excitation, et confirme les travaux précédents. Cependant inversement les modèles 3D-CNN ont significativement mieux performés sur excitation que sur valence, et suggère ainsi que le relevé simultané de caractéristiques spatio-temporelles de l'émotion induit le même phénomène que les caractéristiques audio de séquences audio-visuelles. Par la variation de paramètres inhérents aux architectures 3D-CNN nous avons étudier l'impact de paramètres propres aux méthodes d'inflation et de dilution qui ne se sont finalement pas discriminés à partir de différentes configurations. Cependant la variation de l'initialisation des architectures avec différents ensembles de données (VGG-Face, RAF-DB et ImageNet) a eu un impact conséquent. Bien que ceci soit plus flou pour les valeurs d'excitation, pour les valeurs de valence les architectures de type VGG ont particulièrement bénéficiées d'un pré-entraînement avec ImageNet plutôt qu'avec RAF-DB et VGG-Face. Au contraire ResNet50 s'est montré plus performant avec un pré-entraînement sur VGG-Face à la fois sur les valeurs de valence et excitation. Contre-intuitivement, pour les architectures de type VGG, ImageNet, spécialisé dans la détection d'objet a permis de mieux détecter notre tâche cible de reconnaissance d'émotions que VGG-Face et RAF-DB plus spécialisés dans l'étude des visages. La notion de pré-entraînement est plus complexe quand intervient la méthode d'inflation puisque le transfert d'apprentissage ne prend du sens que pour les textures des données d'entrées, c'est à dire l'aspect 2D. L'aspect temporel des architectures 3D-CNN reste à être défini lors des étapes d'apprentissage suivantes. L'initialisation adéquat des dimensions temporelles est donc essentiel pour une bonne convergence des modèles à notre tâche cible. C'est ce que nous avons voulu démontrer par le biais de différentes méthodes, (e.g. centrage vs recopiage des poids 2D). Pour conclure, le degré de complexité des modèles 3D-CNN par rapport au domaine de distribution des données vidéos SEWA-DB, devait être important pour une tâche de régression de l'émotion sur le circumplex. Les modèles en cascade comme CNN-LSTM restent pour l'instant un meilleur choix. Le Tableau 3.8 résume enfin les résultats obtenus auprès des meilleures architectures.

Tableau 3.8 Résumé des meilleurs résultats comparés à la littérature

Modèle	**Initialisation**	**Valence**		**excitation**	
		PCC	CCC	PCC	CCC
ResNet18 (Kossaifi *et al.*, 2019)	N.A	0.35	0.35	0.35	0.29
VGG-16 + LSTM	**RAF**	**0.631**	**0.625**	**0.558**	**0.557**
VGG-16-3D-BN	ImageNet	0.346	0.304	0.382	0.326
ResNet50-3D	VGG-FACE	0.313	0.253	0.406	0.273

NA : Non Applicable.

CONCLUSION ET RECOMMANDATIONS

Au cours de notre étude, nous nous sommes intéressés à la détection de l'émotion grâce à la source (ou modalité) la plus abondante d'informations actuellement à notre disposition : les expressions faciales. De nombreux modèles de représentations ont été proposés, comme la séparation de l'affect en plusieurs classes d'émotions. Mais celui-ci laisse progressivement place à des modèles plus précis comme celui du circumplex. De même la recherche s'intéresse plus particulièrement aujourd'hui à des sujets exprimant leurs émotions dans un contexte naturel et dans le temps. Comme l'image d'une expression ne peut aisément se différencier parmi plusieurs états émotionnels possibles, l'étude temporelle de visages s'est avéré un élément essentiel de l'affect humain. Ainsi nous avons vu que l'étude de visages au sein de séquences vidéos ; sources riches d'informations ; permet de représenter l'émotion à la fois grâce à une quantité suffisante de données et à l'apprentissage profond de caractéristiques spatio-temporelles.

Notre principale contribution et méthode a alors consisté à se baser sur des modèles de référence ;les CNNs ; pour la détection automatique de l'émotion selon deux grands types d'architectures : modèles en cascade et modèles à convolutions 3D. Premièrement, l'utilisation du mode supervisé d'apprentissage de nos modèles, nous a incité à restructurer notre ensemble de données afin de faire correspondre séquences vidéos et annotations. De plus, plusieurs phases de pré-traitement nous ont permis de redimensionner les données mais aussi de les trier de manière à éliminer les échantillons pouvant être source de bruit à l'apprentissage. En appliquant le principe de transfert d'apprentissage nous avons pu d'une part spécialiser nos modèles CNN pour la reconnaissance d'émotions. Ensuite notre travail a consisté à développer une architecture de type CNN-LSTM et des architectures 3D-CNN.

Dans un premier temps l'architecture CNN-LSTM a particulièrement bien performée et surpasse l'état de l'art sur SEWA-DB en termes de PCC et CCC, de plus la marge d'erreur des prédictions sur les annotations de 7.5% s'avère satisfaisante. En règle général augmenter la lon-

gueur des séquences de trames permet d'apporter plus d'informations et de gagner en précision et performances, augmenter le taux de recouvrement permet de gagner en nombre d'exemples pour l'apprentissage et l'utilisation de la moyenne et des extremum comme méthode de fusion des annotations s'est montré dans les deux cas efficace. Enfin les méthodes de post-traitement (normalisation, filtrage de la moyenne, délai de compensation) ont également permis d'améliorer les performances générales du CNN-LSTM de manière notable. Comme montré par de précédents travaux, les valeurs de valence sont classiquement mieux détectées avec l'information visuelle alors que les valeurs d'excitation le sont mieux avec les données audio. Ce qui est intuitif, par définition des valeurs de valence et excitation. La valence étant le degré de positivité de l'émotion et se traduit visuellement par les traits du visage alors qu'excitation correspond au degré d'agitation de l'émotion, mieux transmis par la fréquence vocale.

Dans un deuxième temps, à partir de modèles pré-entraînés 2D-CNN nous avons pu développer des architectures 3D-CNN en pratiquant la méthode d'expansion des poids préconisée par Carreira & Zisserman (2017). Avec cette méthode, nous avons pu bénéficier du transfert d'apprentissage à partir d'ensembles de données d'images pour une application sur des séquences d'images. Il était ainsi possible de transférer les connaissances apprises d'une distribution 2D à une distribution 3D. De plus la méthode d'expansion des poids offrait une certaine modularité dans l'initialisation (centrage ou recopiage des poids) et l'apprentissage (Ancrage des poids 2D) de nos architectures que nous avons voulu appliqué à notre tâche de régression de l'émotion. Cependant les architectures 3D-CNN se sont montrées trop complexe pour apprendre efficacement des données cibles pour notre application. Le transfert d'apprentissage si essentiel pour une initialisation adéquate de nos modèles a pu bénéficier à une architecture de type CNN-LSTM, mais l'initialisation de noyaux de convolutions 3D avec des noyaux de convolutions 2D pré-entraîné n'a, quant à elle, pas montré de meilleurs performances avec RAF-DB et VGG-Face, au contraire dans certains cas une initialisation avec ImageNet a été plus bénéfique. Enfin, étonnament alors que la valence a mieux été detectée que l'excitation dans notre

première architecture CNN-LSTM, le 3D-CNN a montré une meilleure facilité à détecter les valeurs d'excitation que valence, quelqu'ait été l'initialisation de nos architectures. L'apprentissage simultané de l'information spatio-temporelle aurait plutôt bénéficié à l'excitation que la valence.

En résumé, les modèles CNN-LSTM ont de loin surpassé l'état de l'art mais aussi nos modèles 3D-CNN à la fois en terme de PCC et CCC sur l'ensemble de données SEWA-DB. Pour la valence, nous avons obtenu un CCC de 0.625 avec CNN-LSTM contre 0.304 avec 3D-CNN et 0.35 pour l'état de l'art. L'excitation obtient des résultats similaires.

Au regard des résultats obtenus avec nos modèles 3D-CNN ainsi que les nouvelles méthodes de conception et d'apprentissage présentés dans cette étude : ancrage des poids 2D, variation de l'initialisation des poids étendus sur i3D, et dilution de convolutions, il serait intéressant, pour de futurs travaux, d'observer les performances et pertinences de ces méthodes avec de nouveaux ensembles de données toujours plus larges et complexes, aussi bien sur des tâches similaires à la nôtre comme avec la classification d'émotions, par exemple en faisant varier les modalités d'émotions (audio, et autres signaux corporels) mais également d'autres tâches de reconnaissance. De plus une seule configuration de dilution de convolutions a été éprouvée, il serait intéressant d'observer les variations des degrés de dilution suivant différentes profondeurs d'architecture.

En conclusion, d'après la littérature, les modèles à convolutions 3D se sont montrés particulièrement performants pour des tâches de reconnaissances d'objets et d'actions dans le cadre d'une tâche de classification (Ji *et al.*, 2013; Tran *et al.*, 2014; Liu *et al.*, 2018b). Cependant la détection de valeurs continues de l'émotion dans la vidéo s'avère bien plus sensible et complexe. La variation temporelle des traits du visage étant plus difficilement mis en évidence. Globalement les architectures 3D-CNN ont un grand potentiel de représentation de caractéristiques spatio-temporelles grâce à l'apprentissage simultané des textures et des variations

temporelles. Mais le nombre de critères évolutifs de ces architectures rendent le choix des hyperparamètres des architectures plus complexe. D'après nos résultats, les réseaux en cascade CNN-LSTM représentent toujours un choix de référence et efficace pour notre application. Plus d'importance doivent encore être apporté au pré-traitement des données et à l'initialisation des architectures afin d'obtenir de meilleurs résultats. Notamment en tirant profit de base de données toujours plus complètes à l'avenir et particulièrement adaptée à l'étude émotionnelle de séquences vidéos. Néanmoins, les modèles de représentation de l'émotion multi-dimensionnels sont prometteurs et permettent de sonder plus finement l'affect humain, d'autant que les modèles d'apprentissage profond n'ont pas encore été exploité à leur plein potentiel pour ce type d'application.

BIBLIOGRAPHIE

Abbasnejad, I., Sridharan, S., Nguyen, D., Denman, S., Fookes, C. & Lucey, S. (2017). Using Synthetic Data to Improve Facial Expression Analysis with 3D Convolutional Networks. *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 1609-1618.

Bai, S., Zico Kolter, J. & Koltun, V. (2018). An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv e-prints*, arXiv :1803.01271.

Banda, N., Engelbrecht, A. & Robinson, P. (2015). Feature Reduction for Dimensional Emotion Recognition in Human-Robot Interaction. *2015 IEEE Symposium Series on Computational Intelligence*, pp. 803-810.

Bargal, S. A., Barsoum, E., Ferrer, C. C. & Zhang, C. (2016). Emotion Recognition in the Wild from Videos Using Images. *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, (ICMI '16), 433–436. doi : 10.1145/2993148.2997627.

Barros, P. & Wermter, S. (2016). Developing crossmodal expression recognition based on a deep neural model. *Adaptive Behavior*, 24. doi : 10.1177/1059712316664017.

Ben Henia, W. M. & Lachiri, Z. (2017). Emotion classification in arousal-valence dimension using discrete affective keywords tagging. *2017 International Conference on Engineering MIS (ICEMIS)*, pp. 1-6.

Campos, V., Salvador, A., Giro-i Nieto, X. & Jou, B. (2015). Diving deep into sentiment : Understanding fine-tuned cnns for visual sentiment prediction. *Proceedings of the 1st International Workshop on Affect and Sentiment in Multimedia. ACM*, 57–62.

Cao, Q., Shen, L., Xie, W., Parkhi, O. M. & Zisserman, A. (2017). VGGFace2 : A dataset for recognising faces across pose and age. *arXiv e-prints*, arXiv :1710.08092.

Carreira, J. & Zisserman, A. (2017). Quo Vadis, Action Recognition ? A New Model and the Kinetics Dataset. *arXiv e-prints*, arXiv :1705.07750.

Corneanu, C. A., Simon, M. O., Cohn, J. F. & Guerrero, S. E. (2016). Survey on rgb, 3d, thermal, and multimodal approaches for facial expression recognition : History, trends, and affect-related applications. *IEEE transactions on pattern analysis and machine intelligence*, 38(8), 1548–1568.

Cortes, C. & Vapnik, V. (1995). Support-Vector Networks. *Mach. Learn.*, 20(3), 273–297. doi : 10.1023/A:1022627411411.

Crispell, D., Biris, O., Crosswhite, N., Byrne, J. & Mundy, J. L. (2017). Dataset Augmentation for Pose and Lighting Invariant Face Recognition. *arXiv preprint arXiv :*. doi : 1704.04326.

Darwin, C. (2013). *The Expression of the Emotions in Man and Animals*. Cambridge University Press. doi : 10.1017/CBO9781139833813.

Deng, J., Dong, W., Socher, R., Li, L., Kai Li & Li Fei-Fei. (2009). ImageNet : A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248-255.

Dhall, A., Goecke, R., Lucey, S. & Gedeon, T. (2011a). Acted Facial Expressions In The Wild Database.

Dhall, A., Goecke, R., Lucey, S. & Gedeon, T. (2011b). Static facial expression analysis in tough conditions : Data, evaluation protocol and benchmark. *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 2106-2112.

Dhall, A., Kaur, A., Goecke, R. & Gedeon, T. (2018). EmotiW 2018 : Audio-Video, Student Engagement and Group-Level Affect Prediction. *arXiv e-prints*, arXiv :1808.07773.

Ding, H., Zhou, S. K. & Chellappa, R. (2016a). FaceNet2ExpNet : Regularizing a Deep Face Recognition Net for Expression Recognition. *arXiv e-prints*, arXiv :1609.06591.

Ding, W., Xu, M., Huang, D.-Y., Lin, W., Dong, M., Yu, X. & Li, H. (2016b). Audio and Face Video Emotion Recognition in the Wild Using Deep Neural Networks and Small Datasets. *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, (ICMI '16), 506–513. doi : 10.1145/2993148.2997637.

Ekman, P. (1994). Strong evidence for universals in facial expressions : a reply to russell's mistaken critique. *Psychological bulletin*, 115(2), 268–287.

Ekman, P. & Friesen, W. V. (1971). Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2), 124–129.

Ekman, P. & Friesen, W. V. (1978). Facial Action Coding System : Investigator's Guide. *Palo Altom, CA : Consulting Psychologists Press*.

Fan, Y., Lu, X., Li, D. & Liu, Y. (2016). Video-Based Emotion Recognition Using CNN-RNN and C3D Hybrid Networks. *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, (ICMI '16), 445–450. doi : 10.1145/2993148.2997632.

Fayolle, S. L. & Droit-Volet, S. (2014). Time Perception and Dynamics of Facial Expressions of Emotions. *PLOS ONE*, 9(5), 1-9. doi : 10.1371/journal.pone.0097944.

Georgescu, M.-I., Ionescu, R. T. & Popescu, M. (2019). Local Learning With Deep and Handcrafted Features for Facial Expression Recognition. *IEEE Access*, 7, 64827–64836. doi : 10.1109/access.2019.2917266.

Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.-H., Zhou, Y., Ramaiah, C., Feng, F., Li, R., Wang, X., Athanasakis, D., Shawe-Taylor, J., Milakov, M., Park, J., Ionescu, R., Popescu, M., Grozea, C., Bergstra, J., Xie, J., Romaszko, L., Xu, B., Chuang, Z. & Bengio, Y. (2013). Challenges in Representation Learning : A report on three machine learning contests.

Greene, S., Thaplyal, H. & Caban-Holt, A. (2016). A survey of Affective Computing for Stress Detection. *M.I.T Media Laboratory Perceptual Computing Section Technical Report*. doi : 10.1109/MCE.20162590178.

Gunes, H. & Schuller, B. (2013). Categorical and dimensional affect analysis in continuous input : Current trends and future directions. *Image and Vision Computing*, (2), 120–136.

Guo, Y., Tao, D., Yu, J., Xiong, H., Li, Y. & D.Tao. (2016). Deep Neural Networks with Relativity Learning for facial expression recognition. *2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pp. 1-6.

H.Lu, K.Kpalma & J.Ronsin. (2018). Motion descriptors for micro-expression recognition. *Signal Processing : Image Communication*, 67, 108-117. doi : 10.1016/j.image.2018.05.014.

Hochreiter, S. & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Comput.*, 9(8), 1735–1780. doi : 10.1162/neco.1997.9.8.1735.

Huang, R., Zhang, S., Li, T. & He, R. (2017). Beyond face rotation : Global and local perception gan for photorealistic and identity preserving frontal view synthesis. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2439–2448.

Ioffe, S. & Szegedy, C. (2015). Batch Normalization : Accelerating Deep Network Training by Reducing Internal Covariate Shift.

Jack, R. E., Garrod, O. G., Yu, H., Caldara, R. & Schyns, P. G. (2012). Facial expressions of emotion are not culturally universal. *Proceedings of the National Academy of Sciences*, 109(19), 7241–7244.

Ji, S., Xu, W., Yang, M. & Yu, K. (2013). 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1), 221-231.

Jung, H., Lee, S., Yim, J., Park, S. & Kim, J. (2015). Joint fine-tuning in deep neural networks for facial expression recognition. *2015 IEEE International Conference on Computer Vision (ICCV)*, 2983–2991.

Jyoti, S. & Dhall, A. (2018). Expression Empowered ResiDen Network for Facial Action Unit Detection. *arXiv e-prints*, arXiv :1806.04957.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R. & Fei-Fei, L. (2014, 06). Large-Scale Video Classification with Convolutional Neural Networks. pp. 1725-1732. doi : 10.1109/CVPR.2014.223.

Kaya, H., Gürpınar, F. & Salah, A. (2017). Video-Based Emotion Recognition in the Wild using Deep Transfer Learning and Score Fusion. *Image and Vision Computing*. doi : 10.1016/j.imavis.2017.01.012.

Kim, B., Dong, S., Roh, J., Kim, G. & Lee, S. (2016). Fusing Aligned and Non-aligned Face Information for Automatic Affect Recognition in the Wild : A Deep Learning Approach. *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1499-1508.

Kim, B.-K., Lee, H., Roh, J. & Lee, S.-Y. (2015). Hierarchical Committee of Deep CNNs with Exponentially-Weighted Decision Fusion for Static Facial Expression Recognition. *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, (ICMI '15), 427–434. doi : 10.1145/2818346.2830590.

Kim, H.-R., Kim, Y.-S., Kim, S. J. & Lee, I.-K. (2017). Building Emotional Machines : Recognizing Image Emotions through Deep Neural Networks. *arXiv e-prints*, arXiv :1705.07543.

Kingma, D. P. & Ba, J. (2014). Adam : A Method for Stochastic Optimization. *arXiv e-prints*, arXiv :1412.6980.

Knyazev, B., Shvetsov, R., Efremova, N. & Kuharenko, A. (2017). Convolutional neural networks pretrained on large face recognition datasets for emotion classification from video. *arXiv e-prints*, arXiv :1711.04598.

Kollias, D., Tzirakis, P., Nicolaou , M. A., A.Papaioannou, Zhao, G., Schuller, B., Kotsia, I. & Zafeiriou, S. (2019). Deep Affect Prediction in-the-Wild : Aff-Wild Database and Challenge, Deep Architectures, and Beyond. *International Journal of Computer Vision*,

127(6-7), 907–929. doi : 10.1007/s11263-019-01158-4.

Kollias, D. & Zafeiriou, S. (2018). A Multi-component CNN-RNN Approach for Dimensional Emotion Recognition in-the-wild. *arXiv e-prints*, arXiv :1805.01452.

Kossaifi, J., Schuller, B. W., Star, K., Hajiyev, E., Pantic, M., Walecki, R., Panagakis, Y., Shen, J., M.Schmitt, Ringeval, F. & et al. (2019). SEWA DB : A Rich Database for Audio-Visual Emotion and Sentiment Research in the Wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1. doi : 10.1109/tpami.2019.2944808.

Kouw, W. M. & Loog, M. (2018). An introduction to domain adaptation and transfer learning. *arXiv preprint*. doi : 1812.11806.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM*, 60(6), 84–90. doi : 10.1145/3065386.

L. Tran, X. Y. & Liu, X. (2017). Disentangled representation learning gan for pose-invariant face recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1415–1424.

Lea, C., Vidal, R., Reiter, A. & Hager, G. D. (2016). Temporal Convolutional Networks : A Unified Approach to Action Segmentation.

LeCun, Y., Boser, B., Denker, J. S., D. Henderson, R. E. H., Hubbard, W. & Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *ATT Bell Laboratories*.

Lettvin, J. Y., Maturana, H. R., McCulloch, W. S. & Pitts, W. H. (1959). What the Frog's Eye Tells the Frog's Brain. *Proceedings of the IRE*, 47(11), 1940-1951.

Li, S. & Deng, W. (2018). Deep facial expression recognition : A survey. *arXiv preprint arXiv :*. doi : 1804.08348.

Li, S. & Deng, W. (2019). Reliable Crowdsourcing and Deep Locality-Preserving Learning for Unconstrained Facial Expression Recognition. *IEEE Transactions on Image Processing*, 28(1), 356-370.

Li, S. & Deng, W. (2020). A Deeper Look at Facial Expression Dataset Bias. *IEEE Transactions on Affective Computing*, 1–1. doi : 10.1109/taffc.2020.2973158.

Li, S., Deng, W. & Du, J. (2017). Reliable Crowdsourcing and Deep Locality-Preserving Learning for Expression Recognition in the Wild. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2584-2593.

Liong, S.-T., Gan, Y. S., Yau, W.-C., Huang, Y.-C. & Lit Ken, T. (2018). OFF-ApexNet on Micro-expression Recognition System. *arXiv e-prints*, arXiv :1805.08699.

Liu, C., Tang, T., Lv, K. & Wang, M. (2018a). Multi-Feature Based Emotion Recognition for Video Clips. *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, (ICMI '18), 630–634. doi : 10.1145/3242969.3264989.

Liu, K., Liu, W., Gan, C., Tan, M. & Ma, H. (2018b). T-C3D : Temporal Convolutional 3D Network for Real-Time Action Recognition. *AAAI*.

Liu, P., Han, S., Meng, Z. & Tong, Y. (2014). Facial expression recognition via a boosted deep belief network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1805–1812.

Liu, X., Kumar, B. V. K. V., You, J. & Jia, P. (2017). Adaptive Deep Metric Learning for Identity-Aware Facial Expression Recognition. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 522-531.

Liu, Y.-J., Zhang, J.-K., Yan, W.-J., Wang, S.-J., Zhao, G. & Fu, X. (2015a). A Main Directional Mean Optical Flow Feature for Spontaneous Micro-Expression Recognition. *IEEE Transactions on Affective Computing*, 7, 1-1. doi : 10.1109/TAFFC.2015.2485205.

Liu, Y.-J., Zhang, J.-K., Yan, W.-J., Wang, S.-J., Zhao, G. & Fu, X. (2015b). A Main Directional Mean Optical Flow Feature for Spontaneous Micro-Expression Recognition. *IEEE Transactions on Affective Computing*, 7, 1-1. doi : 10.1109/TAFFC.2015.2485205.

Lowe, D. (1999). Object recognition from local scale-invariant features. *The proceedings of the seventh IEEE international conference*, 2, 1150–1157.

Lowhur, A. & Chuah, M. C. (2015). Dense Optical Flow Based Emotion Recognition Classifier. *2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems*, pp. 573-578.

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z. & Matthews, I. (2010). The Extended Cohn-Kanade Dataset (CK+) : A complete dataset for action unit and emotion-specified expression. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pp. 94-101.

Lyons, M., Akamatsu, S., Kamachi, M. & Gyoba, J. (1998). Coding facial expressions with Gabor wavelets. *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 200-205.

Mollahosseini, A. & Mahoor, M. H. (2019). Categorical and dimensional affect analysis in continuous input : Current trends and future directions. *IEEE Transactions on Affective Computing*, 31, 18-31.

Mollahosseini, A., Chan, D. & Mahoor, M. H. (2016). Going deeper in facial expression recognition using deep neural networks. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1–10.

Mou, W., Celiktutan, O. & Gunes, H. (2015). Group-level arousal and valence recognition in static images : Face, body and context. *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 05, 1-6.

Ng, H.-W., Nguyen, V. D., Vonikakis, V. & Winkler, S. (2015). Deep Learning for Emotion Recognition on Small Datasets Using Transfer Learning. *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, (ICMI '15), 443–449. doi : 10.1145/2818346.2830593.

Nguyen, D., Nguyen, K., Sridharan, S., Ghasemi, A., Dean, D. & Fookes, C. (2017). Deep Spatio-Temporal Features for Multimodal Emotion Recognition. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1215-1223.

Osgood, C. (1952). The nature and measurement of meaning. *Psychological bulletin*, 49(3), 197.

Ouyang, X., Kawaai, S., Goh, E., Shen, S., Ding, W., Ming, H. & Huang, D.-Y. (2017, 11). Audio-visual emotion recognition using deep transfer learning and multiple temporal models. pp. 577-582. doi : 10.1145/3136755.3143012.

Palm, G. (1986). Warren McCulloch and Walter Pitts : A Logical Calculus of the Ideas Immanent in Nervous Activity. *Brain Theory*, pp. 229–230.

Pan, S. J. & Yang, Q. (2010a). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345-1359.

Pan, S. J. & Yang, Q. (2010b). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345-1359.

Pantic, M., Valstar, M., Rademaker, R. & Maat, L. (2005). Web-based database for facial expression analysis. *2005 IEEE International Conference on Multimedia and Expo*, pp. 5 pp.-.

Parkhi, O. M., Vedaldi, A. & Zisserman, A. (2015). Deep Face Recognition. *British Machine Vision Conference*, 1(3), 1-12.

Peng, K.-C., Chen, T., Sadovnik, A. & Gallagher, A. C. (2015). A mixed bag of emotions : Model, predict, and transfer emotion distributions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 860–868.

Pfister, T., Li, X., Zhao, G. & Pietikäinen, M. (2011, 11). Recognising spontaneous facial micro-expressions. pp. 1449-1456. doi : 10.1109/ICCV.2011.6126401.

Picard, R. (1995). A survey of Affective Computing for Stress Detection. *Digital Object Identifier*.

Pitaloka, D. A., Wulandari, A., Basaruddin, T. & Liliana, D. Y. (2017). Enhancing cnn with preprocessing stage in automatic emotion recognition. *Procedia Computer Science*, 116, 523–529.

Polikovsky, S., Kameda, Y. & Ohta, Y. (2009). Facial micro-expressions recognition using high speed camera and 3D-gradient descriptor. *3rd International Conference on Imaging for Crime Detection and Prevention (ICDP 2009)*, pp. 1-6.

Posner, J., Russell, J. A. & Peterson, B. S. (2005). The circumplex model of affect : an integrative approach to affective neuroscience, cognitive development, and psychopathology. doi : https://doi.org/10.1017/S0954579405050340.

Pramerdorfer, C. & Kampel, M. (2016). Facial Expression Recognition using Convolutional Neural Networks : State of the Art. *arXiv e-prints*, arXiv :1612.02903.

Prasanna Teja Reddy, S., Teja Karri, S., Ram Dubey, S. & Mukherjee, S. (2019). Spontaneous Facial Micro-Expression Recognition using 3D Spatiotemporal Convolutional Neural Networks. *arXiv e-prints*, arXiv :1904.01390.

Q. You,J. Luo, H. J. & Yang, J. (2015). Robust image sentiment analysis using progressively trained and domain transferred deep networks. *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Ringeval, F., Schuller, B., Valstar, M., Jaiswal, S., Marchi, E., Lalanne, D., Cowie, R. & Pantic, M. (2015). AV+EC 2015 : The First Affect Recognition Challenge Bridging Across Audio, Video, and Physiological Data. *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*, (AVEC '15), 3–8. doi : 10.1145/2808196.2811642.

Sagonas, C., Panagakis, Y., Zafeiriou, S. & Pantic, M. (2015). Robust statistical face frontalization. *Proceedings of the IEEE International Conference on Computer Vision*, 3871–3879.

Shan, C., Gong, S. & McOwan, P. W. (2009). Facial expression recognition based on local binary patterns : A comprehensive study. *Image and Vision Computing*, 27(6), 803–816.

Shin, M., Kim, M. & Kwon, D. (2016). Baseline CNN structure analysis for facial expression recognition. *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 724-729.

Shreve, M., Godavarthy, S., Goldgof, D. & Sarkar, S. (2011). Macro- and micro-expression spotting in long videos using spatio-temporal strain. *Face and Gesture 2011*, pp. 51-56.

Simonyan, K. & Zisserman, A. (2014a). Two-Stream Convolutional Networks for Action Recognition in Videos. *arXiv e-prints*, arXiv :1406.2199.

Simonyan, K. & Zisserman, A. (2014b). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv e-prints*, arXiv :1409.1556.

Susskind, J. M., Anderson, A. K. & Hinton, G. E. (2010). The Toronto face dataset. *Technical Report UTML TR 2010-001, U. Toronto*.

Tian, Y.-I., Kanade, T. & Cohn, J. F. (2001). Recognizing action units for facial expression analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 23(2), 97–115.

Tran, D., Bourdev, L., Fergus, R., Torresani, L. & Paluri, M. (2014). Learning Spatiotemporal Features with 3D Convolutional Networks. *arXiv e-prints*, arXiv :1412.0767.

Valstar, M. & Pantic, M. (2010). Induced disgust, happiness and surprise : An addition to the mmi facial expression database. *Proc. Int'l Conf. Language Resources and Evaluation, Workshop EMOTION*, 65-70.

Van den Oord, A., Kalchbrenner, N. & Kavukcuoglu, K. (2016). Pixel Recurrent Neural Networks. *arXiv e-prints*, arXiv :1601.06759.

Vielzeuf, V., Pateux, S. & Jurie, F. (2017). Temporal Multimodal Fusion for Video Emotion Classification in the Wild. *arXiv e-prints*, arXiv :1709.07200.

Viola, P. & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference. Computer Vision and Pattern Recognition (CVPR)*, 1.

Wan, L., Liu, N., Huo, H. & Fang, T. (2017). Face Recognition with Convolutional Neural Networks and subspace learning. *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pp. 228-233.

Wang, K., Peng, X., Yang, J., Meng, D. & Qiao, Y. (2019). Region Attention Networks for Pose and Occlusion Robust Facial Expression Recognition.

Wang, Y., Yu, H., Stevens, B. & Liu, H. (2015). Dynamic facial expression recognition using local patch and LBP-TOP. *8th International Conference on Human System Interaction (HSI)*, pp. 362-367.

Warr, P., Bindl, U., Parker, S. & Inceoglu, I. (2014). Four-quadrant investigation of job-related affects and behaviours. *European Journal of Work and Organizational Psychology*, 23, 342-363.

Xie, S., Girshick, R., Dollár, P., Tu, Z. & He, K. (2016). Aggregated Residual Transformations for Deep Neural Networks. *arXiv e-prints*, arXiv :1611.05431.

Xu, C., Cetintas, S., Lee, K.-C. & Li, L.-J. (2014). Visual Sentiment Prediction with Deep Convolutional Neural Networks. *arXiv e-prints*, arXiv :1411.5731.

Yin, X., Yu, X., Sohn, K., Liu, X. & Chandraker, M. (2017). Towards largepose face frontalization in the wild. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3990–3999.

Yu, F. & Koltun, V. (2015). Multi-Scale Context Aggregation by Dilated Convolutions. *arXiv e-prints*, arXiv :1511.07122.

Zhang, K., Zhang, Z., Li, Z. & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10), 1499–1503.

Zhang, Z., Luo, P., Loy, C. C. & Tang, X. (2014a). Facial landmark detection by deep multi-task learning. *European Conference on Computer Vision. Springer*, 94–108.

Zhang, Z., Luo, P., Loy, C. & Tang, X. (2014b, 09). Facial Landmark Detection by Deep Multi-task Learning. doi : 10.1007/978-3-319-10599-4_7.

Zhang, Z., Luo, P., Loy, C. C. & Tang, X. (2015, 12). Learning Social Relation Traits from Face Images. pp. 3631-3639. doi : 10.1109/ICCV.2015.414.

Zhao, G. & Pietikainen, M. (2007a). Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6), 915-928.

Zhao, G. & Pietikainen, M. (2007b). Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE transactions on pattern analysis and*

machine intelligence, 29, 915–928.

Zhao, J., Mao, X. & Zhang, J. (2018). Learning deep facial expression features from image and optical flow sequences using 3D CNN. *The Visual Computer*, 34. doi : 10.1007/s00371-018-1477-y.

Zhao, X., Liang, X., Liu, L., Li, T., Han, Y., Vasconcelos, N. & Yan, S. (2016). Peak-piloted deep network for facial expression recognition. *European conference on computer vision, Springer*, 425–442.

Zhi, R., Flierl, M., Ruan, Q. & Kleijn, W. B. (2011). Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41, 38–52.

Zhong, L., Liu, Q., Yang, P., Liu, B., Huang, J. & Metaxas, D. N. (2012). Learning active facial patches for expression analysis. *Computer Vision and Pattern Recognition (CVPR)*, 2562–2569.

Zhu, X. & Ramanan, D. (2012, 06). Face Detection, Pose Estimation, and Landmark Localization in the Wild. pp. 2879-2886. doi : 10.1109/CVPR.2012.6248014.