République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahleb, Blida1
Faculté des Sciences
Département Informatique

Mémoire de fin d'étude

Pour l'obtention du diplôme de Master en *informatique*

Option : *Ingénierie du Logiciel*

Intitulé

Résumé vidéo multi vues

Réalisé Par :
AMIMER Manel
REZOUG Abderrahmane Sami

Soutenu le : 20/10/2020 Devant le jury composé de :

Président : AROUSSI Sana
Examinateur : CHIKHI Imene
Encadreur : KAMECHE Abdallah Hicham

Année universitaire : **2019-2020**

Résumé

Le système de surveillance vidéo utilises des caméras qui sont installées dans des espaces publics afin de surveiller les divers risques qui peuvent se produire. Ces caméras capturent des images et des vidéos qui présentent différents événements et changements se produisant dans une scène donnée, toutefois, la plupart de ces événements sont sans grande importance. C'est pourquoi le résumé des vidéos revêt une importance sans précédent, il permet d'extraire automatiquement un résumé bref et informatif, met en évidence que les événements pertinents.

Les précédentes études de synthèse vidéo sont conçus pour générer des résumés efficaces pour les vidéos à vue unique, et les résultats ne seraient pas bons s'ils étaient appliqués directement aux vidéos à vues multiples, car les vidéos contiennent des événements inintéressants. La même scène est enregistrée dans différentes vues, ce qui entraîne des dépendances entre les vues et une redondance dans les vues multiples.

Dans ce travail, nous proposons une solution qui consiste à développer une application pour la génération de résumé vidéo multi vues basé sur l'apprentissage profond pour l'extraction des vecteurs caractéristiques profondes et l'utilisation d'une architecture neuronale basée sur les réseaux de neurones récurrents qui prend les fonctionnalités spatiaux-temporelles présentes dans les images de la vidéo pour la génération dynamique du résumé final.

Mots clé : *Résumé vidéo, multi vues, apprentissage profond, réseaux de neurones convolutifs, caractéristiques profondes, réseaux de neurones récurrents.*

Abstract

The video surveillance system uses cameras that are installed in public areas in order to monitor the various risks that may occur. These cameras capture images and videos that show different events and changes occurring in a given scene, however, most of these events are not very significant. This is why the video summary is of unprecedented importance, it automatically extracts a brief and informative summary of highlights only relevant events.

Previous video synthesis studies are designed to generate effective summaries for single-view videos, and the results would not be good if they were applied directly to multi-view videos because the videos contain uninteresting events. The same scene is recorded in different views, resulting in dependencies between views and redundancy in multiple views.

In this work, we propose a solution that consists in developing an application for multi-source video summary generation based on deep learning for the extraction of deep feature vectors and the use of a neural architecture based on recurrent neural networks that takes the spatial-temporal functionalities present in the images of the video for the dynamic generation of the final summary.

Keyword: *video-summarization, multi views, deep learning, convolutional neural networks, deep characteristics, recurrent neural networks.*

نبذة مختصرة

يستخدم نظام المراقبة بالفيديو كاميرات يتم تركيبها في الأماكن العامة من أجل مراقبة المخاطر المختلفة التي قد تحدث. تلتقط هذه الكاميرات صورًا ومقاطع فيديو تعرض أحداثًا وتغييرات مختلفة تحدث في مشهد معين ، ومع ذلك ، فإن معظم هذه الأحداث ليست مهمة جدًا. هذا هو السبب في أن ملخص الفيديو له أهمية غير مسبوقة ، فهو يستخرج تلقائيًا ملخصًا موجزًا ومفيدًا للأحداث البارزة فقط.

تم تصميم دراسات تركيب مقاطع الفيديو السابقة لإنشاء ملخصات فعالة لمقاطع الفيديو أحادية المشاهدة ، ولن تكون النتائج جيدة إذا تم تطبيقها مباشرة على مقاطع الفيديو متعددة العروض لأن مقاطع الفيديو تحتوي على أحداث غير مثيرة للاهتمام. يتم تسجيل نفس المشهد في طرق عرض مختلفة ، مما يؤدي إلى التبعيات بين المشاهدات والتكرار في طرق العرض المتعددة.

في هذه الأطروحة ، نقترح حلاً يتمثل في تطوير تطبيق لتوليد ملخص الفيديو متعدد المصادر استنادًا إلى التعلم العميق لاستخراج نواقل الميزات العميقة واستخدام بنية عصبية تعتمد على الشبكات العصبية المتكررة التي تأخذ البعد المكاني والزماني الوظائف الموجودة في صور الفيديو للتوليد الديناميكي للملخص النهائي.

الكلمة الرئيسية : تلخيص الفيديو، وجهات نظر متعددة، تعلم عميق، الشبكات العصبية التلافيفية الخصائص العميقة ، الشبكات العصبية المتكررة

Remerciements

Ce mémoire représente l'aboutissement de cinq années d'études au Faculté des Sciences de l'Université de Saad Dahleb.

Nous remercions le bon Dieu le tout puissant de nous avoir accordées le courage et la patience pour mener à terme le présent mémoire.

Nous tenons, également, à exprimer notre sincère reconnaissance et notre profonde gratitude à notre encadrant M. KAMECHE Abdallah Hicham, pour sa disponibilité, ses orientations, ses précieux conseils et ses encouragements qui nous ont permis de mener à bien ce travail et surtout sa générosité.

Nous tenons à exprimer notre gratitude aux membres de jury pour avoir accepté de juger ce travail.

Un merci particulier à nos parents, pour leur amour, leurs sacrifices et leurs patiences.

J'(Manel) adresse mes sincères remerciements particuliers à mes très chères parents, ma petite sœur Yousra, mon frère Mehdi, surtout mes amies Isma, Sara, Nesrine, Khouloud et mon binôme Sami, pour leur soutien inconditionnel et leurs encouragements.

Je (Sami) remercie en particulier mes ami(e)s : Younes, Ibra (H et G), Nabil, Mustapha, Sihem, ma binôme Manel et ma Chère Famille, qui ont tous été à mes côtés pendant ces périodes difficiles.

Un énorme merci à nos familles et amis pour leur éternel soutient et la confiance qu'ils ont en nos capacité.

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

ANN Artificial Neural Network

CNN Convolutional Neural Network

DMC Disagreement Minimizing Criterion

FPS Frames Per Second

CBGC Contradictory Binary Graph Coloring

GPU Graphics Processing Unit

HVS Hue Saturation Value

IA Intelligence Artificiel

LIP Local Interest Points

LSTM Long short-term memory. Neural

ML Machine Learning

MMR Maximal Marginal Relevance

DL Deep Learning

PAL Phase Alternating Line

ReLU Rectified Linear Unit

RGB Red Green Blue

RNN Recurrent Neural Network

SA Simulated Annealing

Introduction générale

La vidéosurveillance est un mécanisme de sécurité crucial pour de nombreux lieux publics et privés, les données récoltées des caméras de vidéosurveillance différent selon l'angle de vue, des variations de l'éclairage et souvent les événements intéressants passent inaperçus, ce qui entraîne un gaspillage des ressources de stockage et rend leur analyse difficile.

Avec la propagation des caméras de la vidéosurveillance, les techniques de la vision par ordinateur et d'apprentissage jouent un rôle primordial pour analyser les vidéos. Précisément, il s'agit de faire appel à certains algorithmes qui permettent d'analyser automatiquement la scène au lieu de la superviser manuellement. Parmi l'une des tâches de vision par ordinateur, on retrouve *le résumé vidéo.*

Afin d'avoir une meilleure sémantique pour un résumé vidéo, il serait donc intéressant de supprimer les images inintéressantes et redondantes, réduire ainsi la longueur de la vidéo tout en conservant les scènes et les événements intéressants et combiner les informations collectées depuis plusieurs angles de vue.

Les architectures d'apprentissage profond (*deep learning*) modernes ont atteint des performances compétitives sur de nombreuses tâches relatives. Ces techniques ont permis des progrès importants et rapides dans les domaines de l'analyse du signal sonore ou visuel et notamment de la reconnaissance faciale, de la reconnaissance vocale et de la vision par ordinateur.

L'objectif principal de notre travail est de concevoir et d'implémenter une solution de résumé vidéo multi vues basée sur la notion d'apprentissage profond, une technique prometteuse dans ce contexte afin de générer un résumé d'une haute qualité.

Notre mémoire se subdivise donc comme suit :

- **Chapitre I :** Dans ce chapitre nous avons invoqué une description de la structure des différents composants et des caractéristiques d'une vidéo, suivi d'une étude de quelques descripteurs d'image.
- **Chapitre II :** Ce chapitre contient une étude comparative des travaux proposés dans la littérature pour la génération des résumés vidéo et une analyse de performance pour chaque approche.

- **Chapitre III :**.Ce chapitre a été consacré à l'apprentissage profond, nous avons plus particulière les réseaux de neurones d'architecture CNN et RNN utilisé dans notre approche.
- **Chapitre IV :** Dans ce chapitre nous allons décrire la méthode que nous avons proposée pour la génération du résumé vidéo à partir de plusieurs vidéos.
- **Chapitre V :** Ce chapitre contient la présentation des résultats obtenus et une discussion.

Chapitre I : Résumé vidéo, concepts de base liés à la vidéo

I.1 Introduction

Ces dernières années, nous avons assisté à une croissance dramatique des vidéos dans divers scénarios de la vie réelle, telles que des vidéos de surveillance [1], des vidéos sportives [2] et des vidéos grand public [3]. Parmi les vidéos intéressantes prises en compte dans notre travail figurent les vidéosurveillances. D'après Troung et Venkatesh dans [4] il existe deux types fondamentaux de résumés vidéo : le résumé vidéo statique, qui est une séquence d'images clés, et l'écrémage vidéo dynamique, qui est une collection de sous-clips audio-vidéo composés dynamiquement, et dans les deux cas, l'objectif est de trouver les segments vidéo les plus intéressants ou les plus importants qui capturent l'essence des clips originaux

Dans ce chapitre nous présentons une description de la structure des vidéos, nous citons les principales méthodes que constituent les deux grandes familles de résumé de vidéo tel que la sélection d'images représentatives (*images clés*) et résumé dynamique, résultant d'une sélection de segments extraits de la vidéo, ensuite nous présentons les deux types de signal vidéo analogique et vidéo numérique et quelques descripteurs des caractéristiques d'une image.

I.2 Structure de la vidéo

Une vidéo se compose de succession d'images affichées à une fréquence de 25 images par seconde, accompagnées d'une bande son, chaque image est décomposée en ligne horizontales chaque ligne étant une succession de point (pixels) : 625 lignes (576 effectives) et 720 pixels par ligne pour le PAL. On caractérise la fluidité (vitesse) d'une vidéo par le nombre d'image par seconde (frame rate), exprimé en FPS.

La structure d'une vidéo est illustrée à la figure 1, L'image mobile complète d'une vidéo peut être discrétisée en une séquence d'images finie, c'est-à-dire en de nombreuses images fixes. Chaque image fixe est appelée "*frame*", qui est l'unité de base de la vidéo. La séquence d'images est naturellement indexée par le numéro de l'image. Toutes les images d'une vidéo ont la même taille et le temps entre chaque deux images est égal, généralement 1/25 ou 1/30 de seconde.

Les documents vidéo sont hiérarchiquement structurés en séquences, scènes, plans et images (comme le montre la Figure 1).

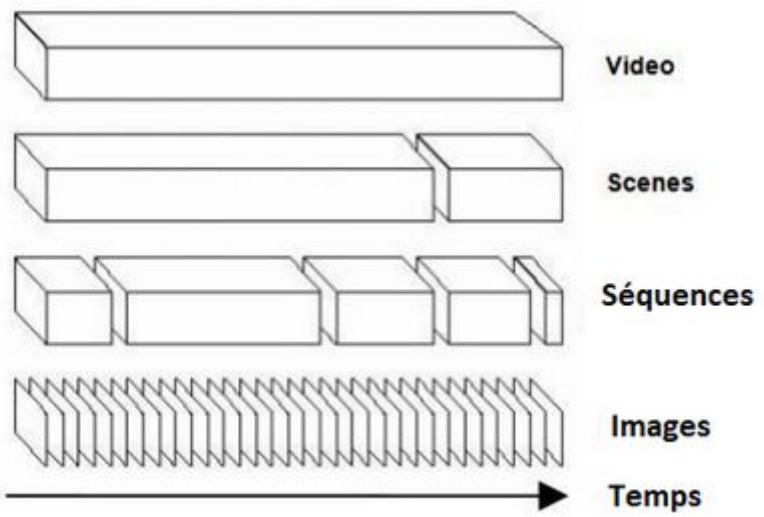


Figure 1: Structure hiérarchique d'une vidéo. [5]

1. **La scène**

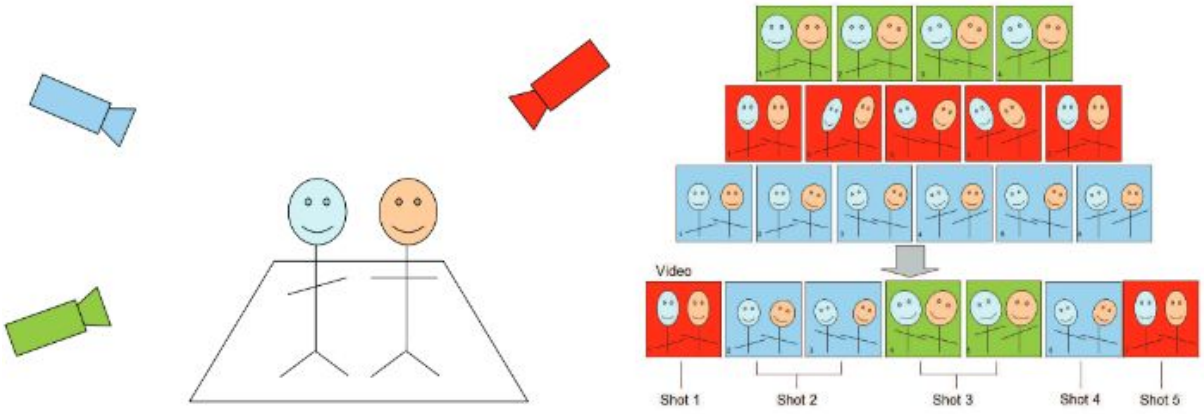
   Une scène est une série de plans qui sont cohérents d'un point de vue narratif. En d'autres termes, une scène est une collection de plans qui transmettent différentes vues d'un même événement ou d'un même objet et qui contiennent les mêmes objets d'intérêt, [6]. L'illustration d'une scène est présentée dans la figure 2.

2. **Le plan (*shot*)/ Séquence**

   Gargi et ses camarades [7] définissent le plan comme une séquence contagieuse d'images vidéo enregistrées à partir d'une seule caméra, représentant une action continue dans le temps et l'espace. Un plan est donc une unité élémentaire sous la forme d'une vidéo plus courte.

3. **Image (*frame*)**

   Une image clé est l'image d'un plan qui transmet le maximum d'informations sur le contenu visuel de l'ensemble du plan. Ainsi, une image clé est l'image la plus représentative d'un plan. [6].



(a)Scène physique capturée par différentes caméras (plans)

(b) des séquences captées par différentes caméras Combinées pour former la scène.

Figure 2: Une scène vidéo [6]

I.3 Résumé vidéo

Devant le volume grandissant des données audiovisuelles, la construction automatique de résumé de vidéo est devenue un domaine de recherche en pleine expansion. Le résumé de vidéo a pour objectif de fournir des informations pertinentes et concises afin d'aider l'utilisateur à naviguer ou à organiser ses fichiers vidéo plus efficacement. Deux sortes de résumé peuvent être retrouvées dans la littérature. [4]

A. Résumé statique

Le résumé statique, est également appelée images représentatives, images R, images fixes abstraites, story-board statique, se compose d'images clés (*key frames*) qui représentent principalement le contenu vidéo. Elle prend en compte l'information visuelle, mais ignore le message audio. Le résumé statique des vidéos est plus facile à parcourir et permet de réduire la complexité des calculs pour la recherche et l'analyse des vidéos.

B. Résumé dynamique

C'est ce que l'on appelle aussi le storyboard mobile ou vidéo écrémée (*Video skim*), ce type de résumé consiste en une collection de segments vidéo (et audio correspondant) extraits de la vidéo originale. Ces segments sont joints par un effet de coupe ou un effet progressif (par exemple, fondu, dissolution, effacement). Il s'agit d'un clip vidéo en soi, mais d'une durée nettement plus courte. La bande-annonce est un type de vidéoclip très populaire dans la pratique.

L'écrémage vidéo a la possibilité d'inclure des éléments audios et de mouvement qui améliorent potentiellement à la fois l'expressivité et l'information de l'abstraction.

La figure 3 illustre les deux types de résumé vidéo (statique et dynamique)

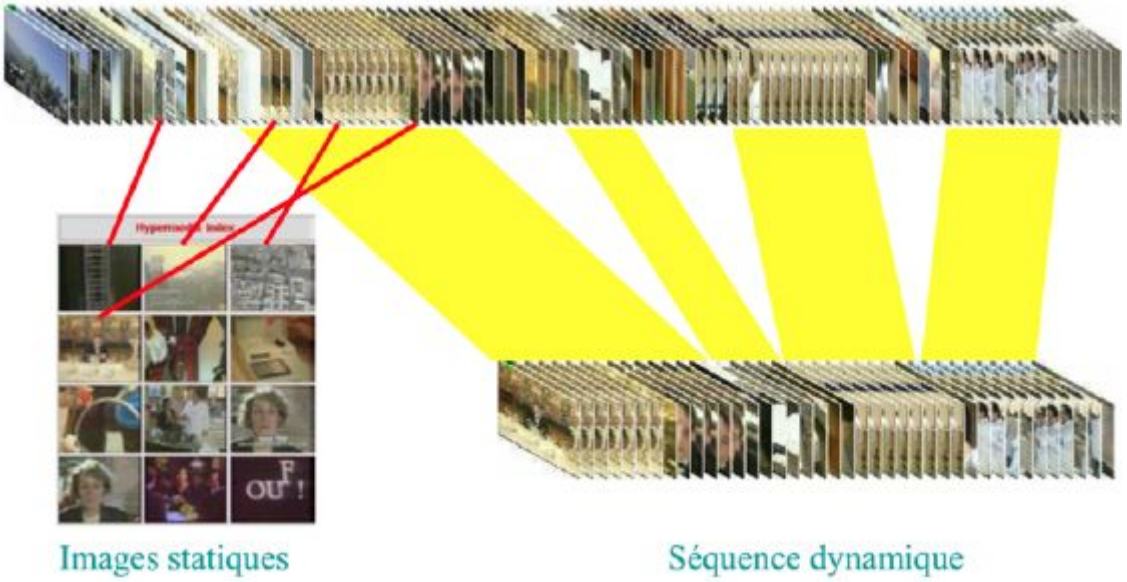


Figure 3:Illustration les deux types de résumé vidéo.[8]

I.4 Signal de vidéo : analogique ou numérique

Le signal vidéo permet de transporter une séquence d'images de la source à un dispositif d'affichage sous forme électrique, d'après [9] il existe deux grandes familles de systèmes vidéo, une vidéo analogique et vidéo numérique.

i. Signal analogique

Les signaux analogiques sont constitués de sons qui changent constamment. Autrement dit, le signal, à un instant donné, peut prendre n'importe quelle valeur comprise entre le minimum et le maximum autorisés (figure 4).

Les images vidéo affichées lui sont transmises sous forme de signal analogique, par l'intermédiaire des ondes ou du câble (destiné à être affichées sur un écran de télévision). Chaque nouvelle transmission ou duplication provoquant inévitablement une accumulation de bruits supplémentaires, la qualité de sa finale est moins bonne à cause de la déperdition engendrée.

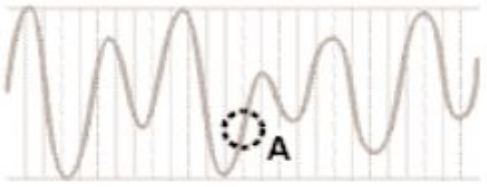


Figure 4: Illustration d'un signal analogique [9]

ii. Signal numérique

Les signaux numériques, sont exclusivement transmis sous forme de points sélectionnés par intervalles sur la courbe (figure 5)

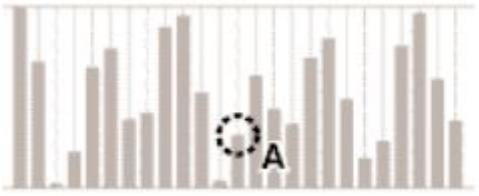


Figure 5: Illustration d'un signal numérique [9]

Un ordinateur peut utiliser un signal numérique de type binaire, qui décrit ces points sous la forme d'une suite de valeurs minimales ou maximales correspondant respectivement au « zéro » et au « un ». Cette suite de « zéros » et de « uns » peut ensuite être interprétée à la réception comme un ensemble de nombres représentatifs de l'information émise à l'origine.

Figure 6: Illustration d'un signal numérique type binaire [9]

Avec un signal numérique, il est beaucoup plus facile de distinguer l'information émise originale des bruits éventuels. De ce fait, un signal numérique peut être transmis et dupliqué aussi souvent qu'il est nécessaire sans perte de fidélité.

I.5 Nombre d'image par seconde (frame rate)

Le système SVH joue le rôle de percevoir, et d'interpréter les images du monde réel. La sensibilité du système SVH à la variation rapide d'une succession d'images permet à l'œil de percevoir un phénomène d'animation. Pour créer ce phénomène dans la bande vidéo, un nombre d'images par seconde est exigé, en général 25 ou 30 images par seconde.

Cependant, la qualité des vidéos ne dépend pas seulement du nombre d'images par seconde. La quantité d'informations contenues dans chaque image est également déterminante. Elle est désignée sous le terme de résolution d'image. La résolution correspond en règle générale au nombre d'éléments individuels constituant l'image (pixels) affichés à l'écran. Elle est exprimée sous la forme du nombre de pixels utilisés sur l'axe horizontal de l'image multiplié par le nombre de pixels utilisés sur l'axe vertical (par exemple, 640 x 480 ou 720x 480). Une résolution plus élevée permet d'obtenir une image de meilleure qualité.

Le nombre d'images par seconde et la résolution sont des paramètres très importants en matière de vidéo numérique, car ils déterminent le volume de données à transmettre et à enregistrer en vue de la diffusion. [9]

I.6 Extraction des caractéristiques

Un problème important dans la conception des résumés est le choix des caractéristiques qui représentent le contenu des images, ceux-ci sont souvent des descripteurs de bas niveau [9] (principalement en termes de couleurs, textures et formes). Il y a deux approches pour les caractéristiques qui peuvent être extraites.

La première est la construction de descripteurs globaux à toute l'image. Dans ce cas, il s'agit de fournir des observations sur la totalité de l'image. L'avantage des descripteurs globaux est la simplicité des algorithmes mis en œuvre, et le nombre réduit d'observations que l'on obtient. Cependant, l'inconvénient majeur de ces descripteurs est la perte de l'information de localisation des éléments de l'image. [11]

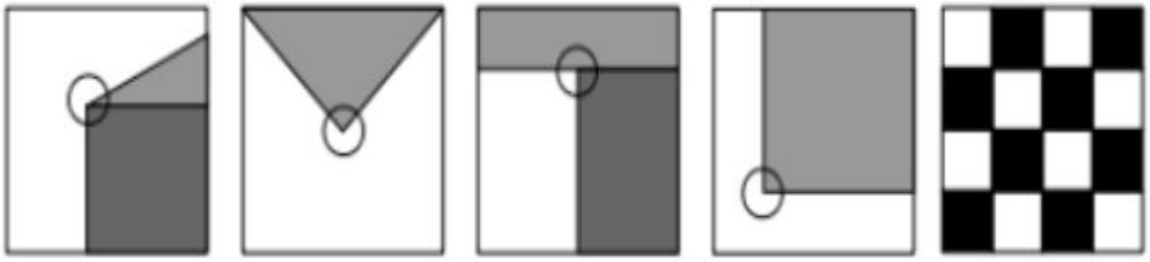
La seconde approche est locale consiste à calculer des attributs sur des portions restreintes de l'image. L'avantage des descripteurs locaux est de conserver une information localisée dans l'image, évitant ainsi que certains détails ne soient noyés par le reste de l'image. L'inconvénient majeur de cette technique est que la quantité d'observations produite est très grande, ce qui implique un gros volume de données à traiter. [11]

I.7 Descripteurs des caractéristiques

De nombreux algorithmes de vision par ordinateur reposent sur la localisation de points d'intérêt ou de points clés dans chaque image et le calcul d'une description d'entité à partir de la région de pixels entourant le point d'intérêt.

a. Détecteurs des points d'intérêt

Les points d'intérêts correspondent généralement à une discontinuité des niveaux de gris comme le montre la figure 8. Ils peuvent également apparaître lors d'une modification de la structure, de la texture ou de la géométrie de l'image [12]



(a) (b) (c) (d) (e)

Figure 7: Illustration les différents types de points d'intérêt : (a) coin simple, (b) jonction en « V », (c) jonction en « T », (d) jonction en « L », (e) jonction en « damier » [13]

Les points d'intérêt sont également utilisés dans différentes applications : la robotique, l'indexation ou la reconnaissance d'objets (on peut extraire d'une image une suite de points caractéristiques d'un objet afin de pouvoir l'indexer dans une base de données), le suivi (ou tracking des objets dans une séquence d'images, ou encore l'imagerie médicale (les points d'intérêt correspondent à des points anatomiques particuliers)

La figure 9 représente un exemple de détection et suivi de personnes, dans une application de surveillance du village, les objectifs peuvent être des personnes.

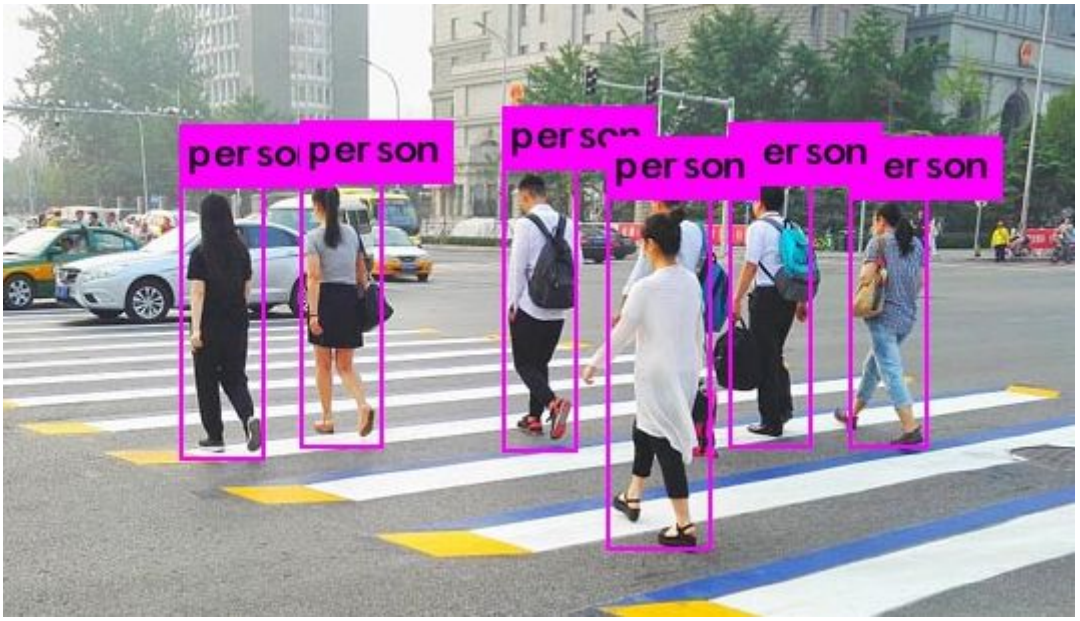


Figure 8: Détection et suivi de personnes [14]

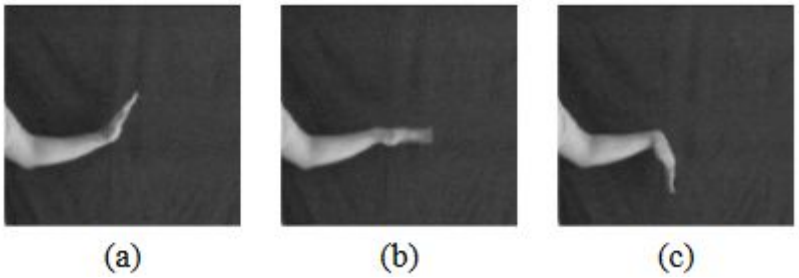


Figure 9: Des exemples de la reconnaissance de gestes, (a) : Début du geste, (b) image intermédiaire, (c) fin du geste. [15]

D'après Schmid ses camarades [16], on distingue trois catégories de détecteurs :

1) **Méthodes basées sur les contours**

   On commence par appliquer un détecteur de contours dans l'image puis on cherche sur ces contours les points d'inflexion, les points ou la courbure est localement maximale ou encore les points d'intersections de plusieurs contours (les jonctions).

2) **Méthodes basées sur l'intensité**

   Ces méthodes sont basées sur la dérivée des niveaux de gris pour repérer les points ou l'intensité varie fortement dans une ou plusieurs directions. Un des premiers détecteurs de cette catégorie est celui de Moravec, Lowe [16] utilise des différences de gaussiennes pour trouver les points clés.

3) **Méthodes basées sur des modèles paramétriques**

Ces méthodes s'appuient sur la déformation d'un modèle paramétrique de coin pour qu'il se rapproche des niveaux de gris au voisinage d'un coin. Ce détecteur est précis à condition d'avoir de bonnes valeurs initiales pour les paramètres du modèle. On peut citer par exemple, le détecteur de Baker. [17]

I.8 Conclusion

Nous nous sommes intéressés dans ce chapitre aux caractéristiques des vidéos. Nous avons présenté, dans un premier temps, la notion de résumé vidéo et ses deux méthodes qui sont les résumés statiques à base de sélection d'images clés appelé *keyframe*, ainsi que les résumés dynamiques à base d'une sélection d'extrait de la vidéo, puis nous avons abordé la structure de base des videos, suivi d'une brève étude sur les différents descripteurs des caractéristiques.

CHAPITRE II : L'ÉTAT DE L'ART SUR LE RÉSUMÉ VIDÉO.

II.1 Introduction

De nos jours, le résumé vidéo est devenu l'outil clé pour une navigation, un accès et une manipulation efficaces des grandes collections vidéo. Depuis les années 1990, le résumé vidéo a attiré l'attention d'un grand nombre de chercheurs. La plupart des travaux existants se concentrent sur le résumé d'une seule vidéo basé sur différentes techniques telles que le mouvement [18], l'audio [19] et la multimodalité [20].

Le résumé vidéo est un mécanisme qui permet de générer un court résumé d'une vidéo, qui peut être soit une séquence d'images fixes (images clés) (*key frames*) ou d'images animées (écrémages vidéo) (*video skims*) [4]. Le premier type est appelé le résumé vidéo statique (généralement présenté sous forme de story-board) se compose d'images clés qui représentent principalement le contenu vidéo. Elle prend en compte l'information visuelle mais ignore le message audio. Le deuxième appelé le résumé vidéo dynamique est un clip vidéo qui combine des informations image, audio et texte. Par rapport au résumé vidéo dynamique, le résumé vidéo statique est plus facile à parcourir et permet de réduire la complexité de calcul pour la récupération et l'analyse de la vidéo, mais le résumé dynamique a la possibilité d'inclure des éléments audios et de mouvement qui améliorent potentiellement à la fois l'expressivité et l'information de l'abstraction.

Dans ce chapitre, nous décrivons les techniques existantes de création de résumés vidéo regroupés en deux catégories, pour les vidéos à une vue unique et pour multi vue.

II.2 Résumé vidéo à vue unique

Parmi les nombreuses approches proposées pour résumer les vidéos à vue unique, les approches supervisées se distinguent généralement par leurs meilleures performances.

Les réseaux neuronaux récurrents en général (RNN), et la mémoire longue et courte durée (LSTM) en particulier, ont été largement utilisés dans le traitement vidéo pour obtenir les caractéristiques temporelles des vidéos [21], [22] et [23].

Zhang et al, en 2016 [24] utilisent un mélange de LSTM bidirectionnels (Bi-LSTM) et de Perceptron multicouches pour additionner des vidéos à vue unique de manière supervisée.

De plus, Mahasseni et al, en 2017 [25] présentent un cadre qui forme de manière contradictoire les LSTM, où le discriminateur est utilisé pour apprendre une mesure de similarité discrète pour former les LSTM de l'encodeur/décodeur actuel et du sélecteur de trame vidéo éparses qui représentent de manière optimale la vidéo d'entrée.

II.3 Résumé vidéo en multi vues

A. Résumé vidéo basé sur un graphique de prise de vue spatio-temporelle

Fu et al en 2010 [26] introduisent le problème de la synthèse multi-vues vidéo adaptée aux caméras de surveillance fixes. Dans un premier temps, un graphique spatio-temporel est construit pour la vidéo d'entrée et un étiquetage du graphique est effectué pour générer la vidéo résumée. Un hypergraphe est initialement créé dans lequel les bords contiennent la corrélation des différents attributs des plans vidéo multi-vues. Le graphe de plans spatio-temporels est dérivé d'un hypergraphe, le graphe de plans est ensuite partitionné et des groupes de plans centrés sur les événements sont identifiés par des marches aléatoires.

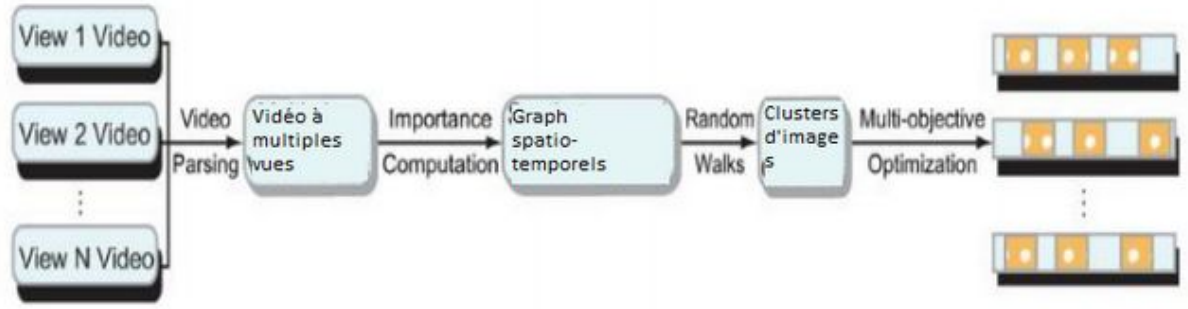


Figure 10: Aperçu de méthode de résumé vidéo multi-vues de Fu et al, [26]

Le résultat de la synthèse est généré par la résolution d'un problème d'optimisation multi-objectifs basé sur l'importance des tirs évalués à l'aide d'un schéma de fusion d'entropie gaussienne. Les différents objectifs de la synthèse, tels que la longueur minimale du résumé et la couverture maximale des informations, sont obtenus dans ce cadre. Les résumés multi-vues sont proposés par le storyboard multi-vues et le tableau d'événements présentés à la figure 10.

Dans la figure 11, Le *story-board* assemble en série des plans multi-vues centrés sur les événements dans un ordre temporel.

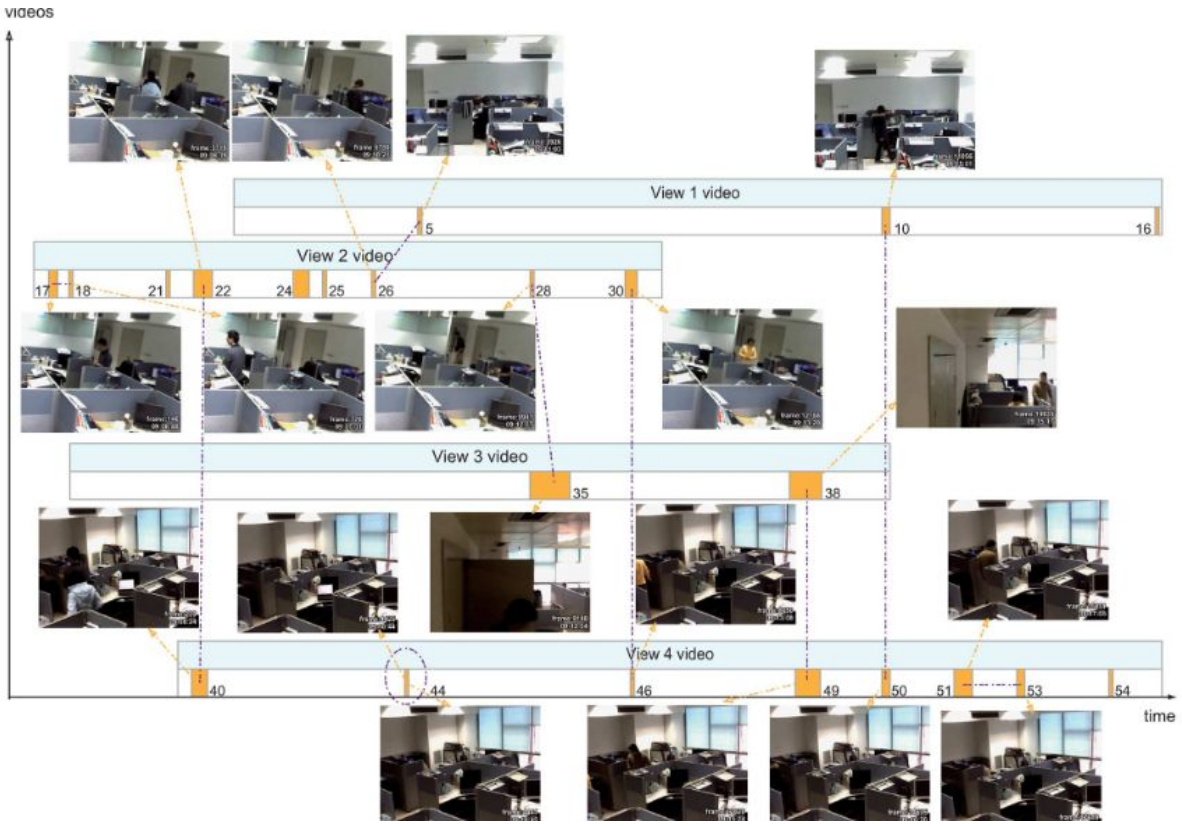


Figure 11: Résumé du Vidéo Story Board Multi-Vues [26]

B. Résumé de la vidéo à l'aide de l'algorithme MMR

Yingbo Li et Bernard Merialdo en 2010 [27] ont proposé la technique d'extraction d'images clés basée sur l'analogie de l'algorithme MMR (*Maximal Marginal Relevance)* pertinence marginale maximale de la vidéo avec l'algorithme classique de résumé de texte, la pertinence marginale maximale pour le résumé multi-vidéo. Le Vidéo MMR conserve les images clés pertinentes et supprime les images clés redondantes.

Les histogrammes des mots visuels sont les caractéristiques extraites des images vidéo. Le descripteur SIFT est calculé en détectant les points d'intérêt locaux (LIP *Local interest points)* dans l'image, en prenant la différence du gaussien et du laplacien du gaussien. K-means est appliqué aux descripteurs SIFT pour composer un vocabulaire visuel de 500 mots. Le cosinus de similarité entre les images successives est calculé et l'algorithme Vidéo MMR est appliqué pour sélectionner les images clés représentatives. Il propose également deux méthodes : le résumé global et le résumé vidéo individuel. La synthèse individuelle génère un résumé pour chaque vidéo de l'ensemble et concatène ces résumés. La synthèse globale prend en compte simultanément les relations inter et intra- des vidéos individuelles et évite la redondance de la synthèse individuelle.

C. Résumé vidéo multi-vues sur de nombreux GPU

Pandurang Matkar et al, en 2016 [28] ont proposé un cadre pour la synthèse vidéo multi-vues sur de nombreux GPU (*Graphics Processing Unit*) de base. Une unité de traitement graphique, un processeur à une seule puce utilisé principalement pour gérer et augmenter les performances de la vidéo et des graphiques. La vidéo d'entrée est divisée en cubes de données adjacents par un algorithme de segmentation temporelle. Deux images vidéo consécutives sont transformées par DWT, puis les différences de caractéristiques statistiques des deux images sont calculées. Si la valeur de la différence d'une paire est supérieure au seuil, la dernière image de la paire est considérée comme une image clé. Une synthèse vidéo est créée par les images clés extraites. La sortie est un résumé vidéo statique.

D. Résumé base sur le cadre de synopsis vidéo multi-vues

Mahapatra et al en 2016 [29] ont proposé un cadre pour la création d'un synopsis de vidéos à vues multiples capturées par des caméras de surveillance (intérieures et extérieures) dont les champs de vision se chevauchent. Dans les synopsis vidéo, les emplacements spatiaux des objets sont inchangés mais les objets sont déplacés le long de l'axe temporel et représentés simultanément dans un plan de base commun.

Un plan de base commun est créé pour les vidéos capturées par plusieurs caméras. Pour les vidéos d'extérieur, l'ensemble de données PETS 2009, la vue de dessus du site est trouvée par Google Map et pour les vidéos d'intérieur, un plan de base commun est identifié. Le travail proposé est limité aux actions humaines identifiées dans la vidéo. La création du synopsis est obtenue par trois techniques : colorisation de graphes binaires contradictoires (CBGC, *contradictory binary graph coloring*), approche par tableau et approche basée sur le recuit simulé (SA, *simulated annealing*).

Le module de reconnaissance des actions est utilisé pour identifier les actions importantes des humains dans la vidéo. Ces actions importantes réduisent la longueur du synopsis. Le système d'inférence floue calcule le score de visibilité de chaque piste d'objet dans la vidéo, ce qui réduit encore la longueur du synopsis. Dans l'approche CBGC, la réduction maximale de la longueur du synopsis est obtenue. L'approche stochastique utilisant la SA, en revanche, permet d'obtenir un meilleur compromis entre les multiples critères d'optimisation.

La figure suivante (figure 12) illustre un framwork pour le synopsis vidéo multi-vues

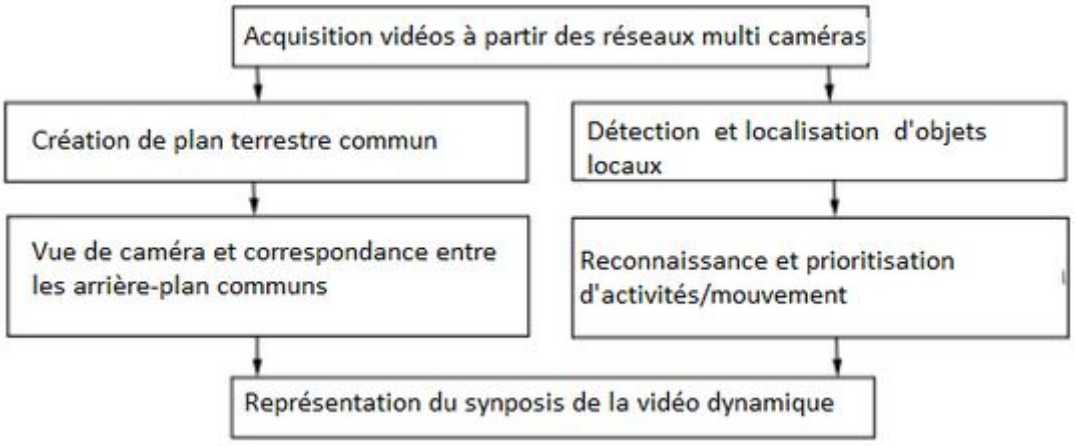


Figure 12: Un Framework pour le synopsis vidéo multi-vues [29]

E. Cadre d'apprentissage métrique multi-vues

La méthode de Linbo Wang et al, en 2016 [30] utilise l'apprentissage à noyaux multiples pour résoudre le problème des vues multiples et la métrique de distance optimale est utilisée pour obtenir des groupes cohérents. Elle propose un cadre d'apprentissage *Unified Metric* en intégrant à la fois le *Disagreement Minimizing Criterion* (DMC) et le *Maximum Margin Criterion* (DMC). La vidéo d'entrée est convertie en une séquence d'images. Chaque vue vidéo est représentée dans son propre vecteur dimensionnel *Feature*. Ces images sont introduites dans le cadre d'apprentissage métrique qui construit l'espace métrique commun, c'est-à-dire que les caractéristiques de haut niveau de chaque vue sont intégrées dans le même espace commun de bas niveau. Après K-mean, l'algorithme de mise en grappes est appliqué sur les images pour extraire les images clés. Les images clés sont disposées dans l'ordre temporel pour obtenir la vidéo résumée.

F. Mise en sac de l'événement, résumé vidéo de l'ensemble

Krishan Kumar et al, en 2017 [31] ont proposé la méthode de l'apprentissage automatique en groupe pour résumer le contenu de la vidéo. La méthode d'agrégation bootstrap est utilisée. La vidéo d'entrée est convertie en Frames **N**. Dans la phase de formation, des échantillons bootstrap de différentes scènes de la vue individuelle de la vidéo sont pris. La taille de l'échantillon est **m**, qui doit être inférieure à **N (m<N)**.

Pour les vues **P**, des échantillons bootstrap **P** sont pris et donnés en entrée aux classificateurs **P** qui donnent l'arbre de décision en sortie. Un nœud de l'arbre de décision est la trame et l'arbre est formé par la variance (**σ**) entre les trames. L'arbre de décision n'est pas élagué et présente donc une variance élevée. Le cadre proposé est donné par la figure 13.

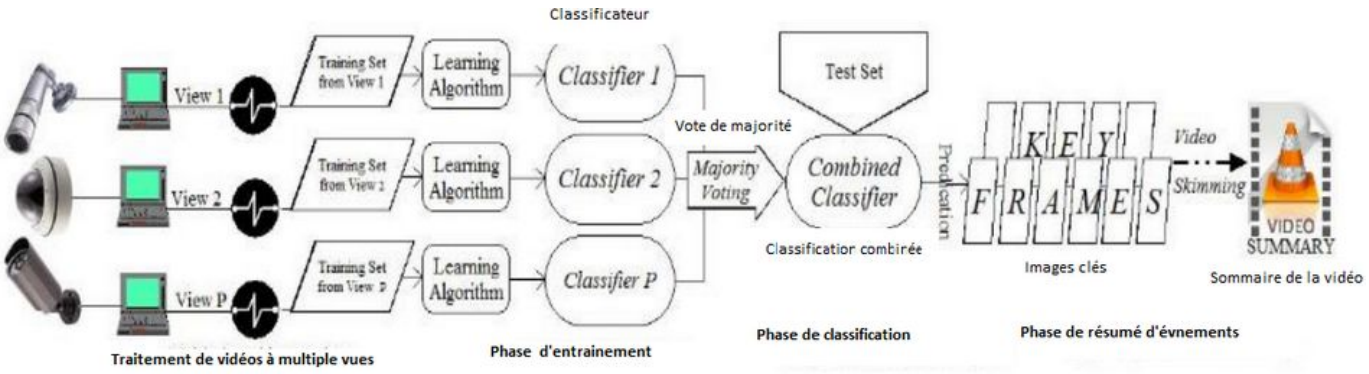


Figure 13: Mise en sac de l'événement [31]

Après la formation, sur la base de la sortie du classificateur précédent, le classificateur actuel est pesé et ajouté à l'ensemble.

Dans la phase de test, les images vidéo de la vue individuelle de la caméra sont données en entrée au classificateur combiné. Si une image d'une vue quelconque apparaît dans plus de 70% des arbres classés, elle est déclarée comme l'image clé, sinon elle est rejetée. C'est ce qu'on appelle la politique de vote à la majorité. Les images en double sont également supprimées à ce stade. L'étape suivante est le résumé de l'événement. Si la distance euclidienne entre une trame et la trame clé de l'événement est égale ou supérieure à la valeur du seuil de limite de l'événement, alors la trame courante est comptée dans l'événement courant, sinon elle est rejetée.

G. Résumé des vidéos multi-vues via l'intégration conjointe et l'optimisation des éléments

Rameswar Panda et al en 2017 [32] ont proposé une nouvelle méthode de cadre non supervisé pour résumer les vidéos multi-vues via l'intégration conjointe et l'optimisation éparse. L'intégration est utilisée pour capturer les corrélations de contenu dans un ensemble de données multi-vues. La sélection représentative éparse est utilisée pour générer des résumés Multi vues basés sur la demande de longueur de l'utilisateur sans coût de calcul supplémentaire.

La vidéo est segmentée en plusieurs plans en mesurant la différence des espaces colorimétriques RGB (rouge vert bleu) et HSV (Hue saturation value) de deux images consécutives dans la vidéo. Les caractéristiques visuelles sont extraites en appliquant des filtres convolutifs 3D à un ensemble de 16 images vidéo d'entrée et les réponses sont enregistrées au niveau de la couche FC6. La structure d'ordre locale dans un plan est maintenue par un schéma de mise en commun de la moyenne temporelle.

Le schéma de mise en commun donne le vecteur de caractéristique final d'un tir (4096 en dimension), qui est utilisé pour l'optimisation de l'éparpillement. Tous les plans sont intégrés dans un espace latent commun en tenant compte des similitudes entre deux plans dans une vidéo individuelle (Inter vue) et dans deux vidéos différentes (Intra vue). Le résumé des vidéos multi vues est le sous-ensemble optimal de tous les plans intégrés

H. FASTA

Krishan Kumar et Shrimankar en 2018 [33] ont proposé l'approche FASTA qui est une méthode basée sur l'alignement local pour résumer les événements dans les vidéos Multi vues. Le réseau neuronal convolutif (CNN) est formé avec des images d'entrée RGB avec de multiples filtres multicanaux.

Au départ, **N** images de longueur égale d'une seule vue sont introduites dans ces CNN pour en extraire les caractéristiques visuelles et la détection des objets. Les caractéristiques extraites des CNN sont utilisées pour un traitement vidéo ultérieur. Une image peut être classée dans l'un des types suivants, en fonction de la présence d'éléments de preuve (nombre d'objets en mouvement).

1) **NE** : Pas de preuve.
2) **SH** : Quelques indices
3) **SE** : Preuve significative.
4) **SV** : le cadre comporte plus de deux objets en mouvement.

La séquence de nucléotides est formée en attribuant un label "**A**", "**C**", "**G**", "**T**" aux trames qui présente une similarité cosinusoïdale maximale entre la trame actuelle et la trame précédente. FASTA, un algorithme d'alignement local rapide est utilisé pour supprimer la redondance entre les vues et pour capturer les corrélations entre plusieurs vues en utilisant une approche d'alignement optimisée. D'autres images redondantes sont supprimées en utilisant la méthode de suivi d'objet. L'architecture de la méthode proposée est donnée par la figure 41.

Les images clés extraites sont ensuite disposées dans l'ordre chronologique pour obtenir la vidéo résumée.

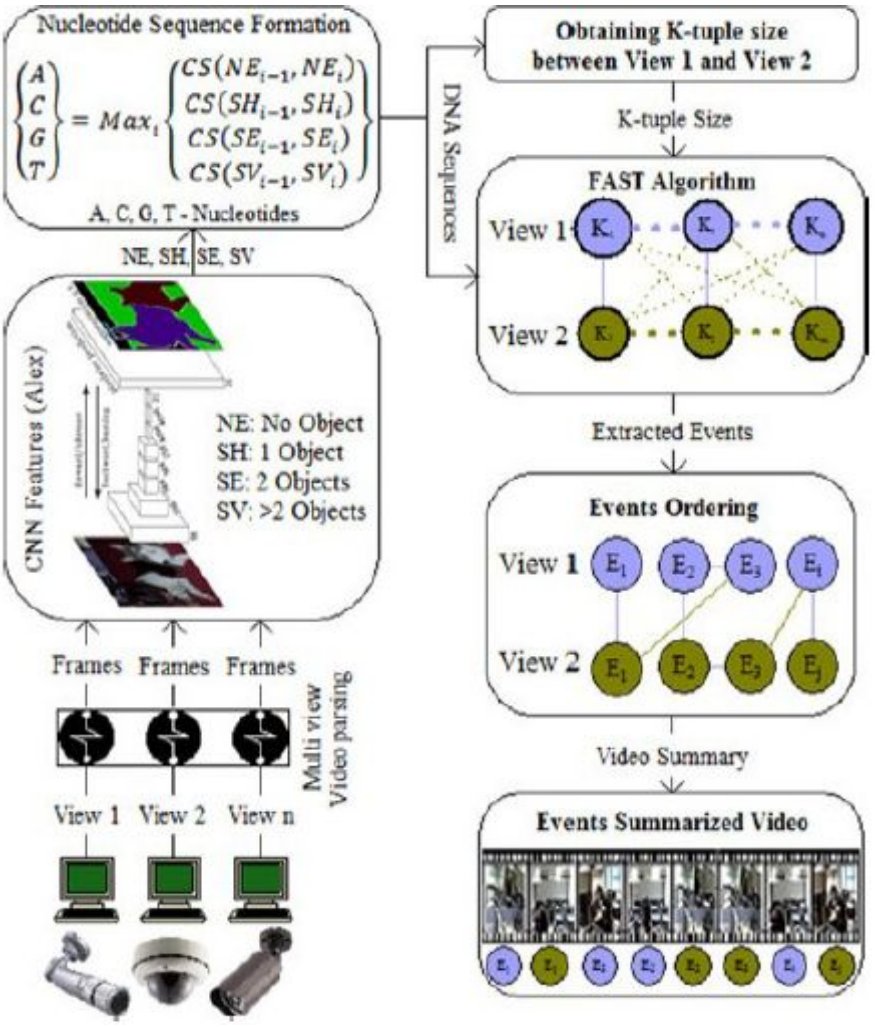


Figure 14: Architecture proposée pour FASTA [33]

Le tableau suivant résume chaque approche

[26]	[27]	[28]	[29]
•Un graphique spatio-temporel pour la vidéo d'entrée •Les différents objectifs de la synthèse, tels que la longueur minimale du résumé et la couverture maximale des informations, sont obtenus dans ce cadre	•d'extraction d'images clés basée sur l'analogie de l'algorithme MMR conserve les images clés pertinentes et supprime les images clés redondantes. •La synthèse globale prend en compte simultanément les relations inter et intra- des vidéos	•un processeur à une seule puce utilisé pour gérer et augmenter les performances de la vidéo et des graphiques. •Une synthèse vidéo est créée par les images clés extraites	•la création d'un synopsis de vidéos à vues multiples •Le module de reconnaissance des actions est utilisé pour identifier les actions importantes des humains dans la vidéo •la synthèse video , permet d'obtenir un meilleur compromis entre les multiples critères d'optimisation.

[30]	[31]	[32]	[33]
•l'apprentissage à noyaux multiples pour résoudre le problème des vues multiples •. Les images clés sont disposées dans l'ordre temporel pour obtenir la vidéo résumée.	•la méthode de l'apprentissage automatique en groupe pour résumer le contenu de la vidéo •utilise la politique de vote à la majorité pour classes les images clés	•La vidéo est segmentée en plusieurs plans en mesurant la différence des espaces colorimétriques •Les caractéristiques visuelles sont extraites en appliquant des filtres convolutifs 3D •Le résumé des vidéos multi vues est le sous-ensemble optimal de tous les plans intégrés	•une méthode basée sur l'alignement local pour résumer les événements dans les vidéos Multi vues •Les images clés extraites sont ensuite disposées dans l'ordre chronologique pour obtenir la vidéo résumée.

Tableau 1: Résumé pour chaque approche

II.4 Conclusion

Dans ce chapitre nous avons présenté les techniques existantes de création automatique de résumées vidéo, nous commençons par les vidéos mono vue, ensuite les vidéos multi-vues. Dans une troisième partie, nous avons faire une analyse de performance pour chaque proche des vidéos multi vues.

CHAPITRE III : LES RÉSEAUX DE NEURONES

III.1 Introduction

L'intelligence artificiel (IA) est une discipline relative au traitement des connaissances et au raisonnement dans le but de permettre à une machine d'exécuter des fonctions normalement associées à l'être humain. Elle tente de reproduire les processus cognitifs humains dans le but de réaliser des actions « intelligente »

Les tâches relevant de l'IA sont parfois très simples pour les humains, comme par exemple reconnaître et localiser les objets dans une image, planifier les mouvements d'un robot pour attraper un objet, ou conduire une voiture. Elles requièrent parfois de la planification complexe, comme par exemple pour jouer aux échecs ou au Go. Les tâches les plus compliquées requièrent beaucoup de connaissances et de sens commun, par exemple pour traduire un texte ou conduire un dialogue.

L'apprentissage automatique (ML, *Machine learning*) cherche à permettre l'ordinateur d'imiter la capacité humaine d'apprendre à partir d'exemples, lui donnant la possibilité d'agir sans être explicitement programmé. En général, ce domaine se concentre sur les algorithmes qui apprennent à partir d'exemples pour ensuite permettre de généraliser sur de nouveaux exemples non observés auparavant. L'apprentissage automatique est maintenant une composante importante de plusieurs domaines tels que le traitement automatique du langage naturel, la reconnaissance d'objets, la reconnaissance de la parole, la bio-informatique et bien d'autres encore. [34].

L'apprentissage automatique peut être décomposé en quatre principales catégories : l'apprentissage supervisé, semi supervisé, non supervisé et par renforcement. Nous allons les détailler dans ce qui suit.

Ce chapitre s'intéresse donc à l'utilité de l'apprentissage profond pour la génération du résumé vidéo. Nous allons dans un premier temps présenter les différents types d'apprentissage automatique suivi d'apprentissage automatique et enfin une étude sur les réseaux de neurones utilisé dans notre approche.

III.2 Les types d'apprentissage automatique

Les différents types d'apprentissage automatique sont

a. Apprentissage supervisé

Les données utilisées pour l'apprentissage supervisé sont étiquetées : on connaît le résultat auquel le modèle doit parvenir pour chaque exemple utilisé lors de l'apprentissage. Le but du modèle est d'apprendre à associer chaque exemple à une étiquette.

L'apprentissage supervisé est défini par Cunningham et al. [35] en ces termes : « L'apprentissage supervisé implique l'apprentissage d'une mise en correspondance entre un ensemble de variables d'entrée X et une variable de sortie Y et en appliquant cette correspondance pour prédire les sorties pour des données non visualisées ».

Le but est donc de déduire une fonction f : X → Y d'un ensemble de d'apprentissage A_n composé de paires (exemple, label) tel que :

$$A_t = (x_1, y_1) , \ldots, (x_t y_t) \ \epsilon (x * y)^t$$

Où x_t, t ∈ 0…T est généralement défini dans R et y_t est généralement défini soit dans R pour un problème de régression, soit dans une sous partie de N pour un problème de classification. On a donc un problème de régression quand il s'agit de prédire une valeur numérique continue (par exemple prédire le prix d'une maison) et un problème de classification quand il s'agit d'assigner une classe (par exemple déterminer si une image est celle d'un chien ou d'un chat).

a. Apprentissage non supervisé

Contrairement à l'apprentissage supervisé, les données utilisées pour l'apprentissage non supervisé ne sont pas étiquetées. Cela signifie que pour une entrée donnée, on ne sait pas dire quel est le résultat correct, celui qui est juste.

L'objectif est induit par la formulation du problème. On cherche en général à mettre en évidence une structure dans les données. On va chercher par exemple à déterminer la distribution de probabilités ayant permis de générer le jeu de données ou regrouper des données similaires. On s'intéresse ici à trois catégories de modèles. Les modèles de partitionnement, de détection d'anomalies, et quelques réseaux de neurones. Le but de ces modèles est d'apprendre le schéma de distribution interne des données, dans le but de le visualiser, en retirer des informations, ou le reproduire.

b. Apprentissage semi supervisé

L'apprentissage semi-supervisé est en fait un mélange des deux approches que l'on vient de présenter, soit l'apprentissage supervisé et non-supervisé. L'apprentissage semi-supervisé concerne le cas où le jeu de données est partiellement étiqueté. L'objectif est d'entrainer un modèle qui soit capable de tirer parti à la fois des cibles présentes mais aussi des données non étiquetées [36].

c. Apprentissage par renforcement

Le domaine de l'apprentissage par renforcement cherche à apprendre à un agent à se comporter de la bonne façon à l'intérieur d'un environnement spécifique, c'est-à-dire de façon à atteindre un but choisi préalablement par l'utilisateur. Le problème que l'on désire résoudre est divisé en une séquence d'étapes. À chaque étape, un agent doit choisir parmi un ensemble d'actions, lui donnant la possibilité d'interagir avec son environnement. Contrairement à l'apprentissage supervisé, il n'y a pas de cible qui donne la possibilité d'apprendre un comportement. À la place, l'agent reçoit un signal (déterminé par l'utilisateur) qui lui permet de savoir s'il a agi correctement. Pour chaque étape de la séquence, l'agent reçoit de l'information sur son environnement qui l'aidera à choisir l'action appropriée. Durant l'apprentissage, l'agent cherchera à maximiser le nombre de signaux positifs afin d'améliorer son comportement. [35].

III.3 Apprentissage profond

L'apprentissage profond (DL *Deep Learning*) est un nouveau domaine de recherche de l'apprentissage automatique (ML *Machine Learning*), qui a été introduit dans le but de rapprocher le ML de son objectif principal à savoir : l'intelligence artificielle.

L'apprentissage profond est un ensemble d'algorithmes d'apprentissage automatique qui tentent d'apprendre à plusieurs niveaux, correspondant à différents niveaux d'abstraction. Il a la capacité d'extraire des caractéristiques à partir des données brutes grâce aux multiples couches de traitement composé de multiples transformations linéaires et non linéaires et apprendre sur ces caractéristiques petites à petit à travers chaque couche avec une intervention humaine minime. [37]

La figure suivante montre la différence entre l'apprentissage automatique et profond,

Une des grandes différences entre l'apprentissage profond et les algorithmes de l'apprentissage automatique traditionnelles c'est qu'il s'adapte bien, plus la quantité de données fournie est grande plus les performances d'un algorithme de l'apprentissage profond sont meilleures. Contrairement à plusieurs algorithmes de l'apprentissage automatique classiques qui possèdent une borne supérieure à la quantité de données.

Autre différence entre les algorithmes de ML traditionnelles et les algorithmes de DL c'est l'étape de l'extraction de caractéristiques. Dans les algorithmes de ML traditionnelles l'extraction de caractéristiques est faite manuellement, c'est une étape difficile et coûteuse en temps et requiert un spécialiste en la matière alors qu'en DL cette étape est exécutée automatiquement par l'algorithme.

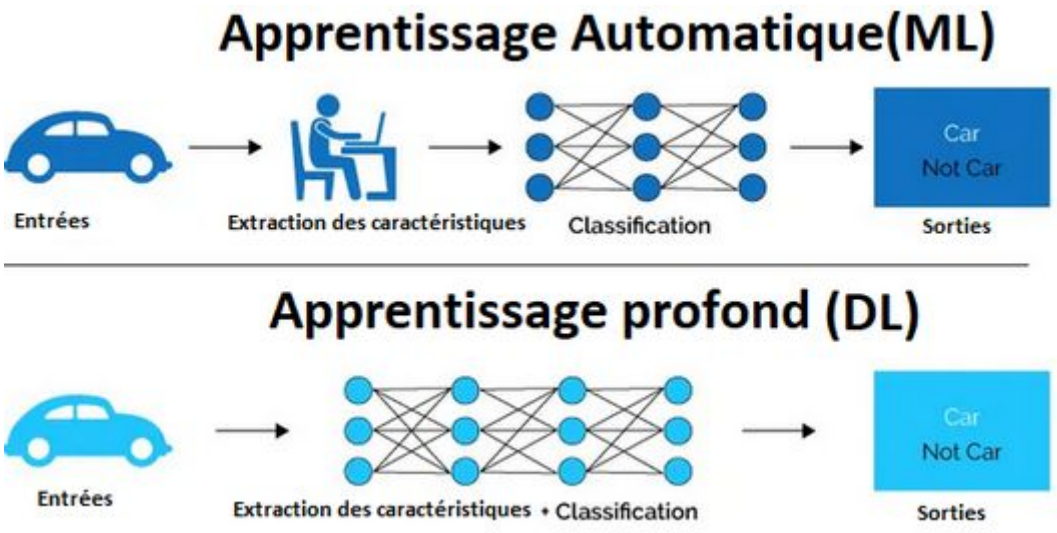


Figure 15:Comparaison entre ML et DL [38]

III.4 Le réseau de neurone artificiel

La pratique, de tous les algorithmes de ML sont des réseaux neuronaux. Les réseaux neuronaux artificiels, aussi appelés ANN, sont des modèles de traitement de l'information qui simulent le fonctionnement d'un système nerveux biologique. C'est similaire à la façon dont le cerveau manipule l'information au niveau du fonctionnement. Tous les réseaux neuronaux sont constitués de neurones inter connectés qui sont organisés en couches. [39]

Les deux figures suivantes montrent une représentation d'un neurone réel et d'un neurone artificiel.

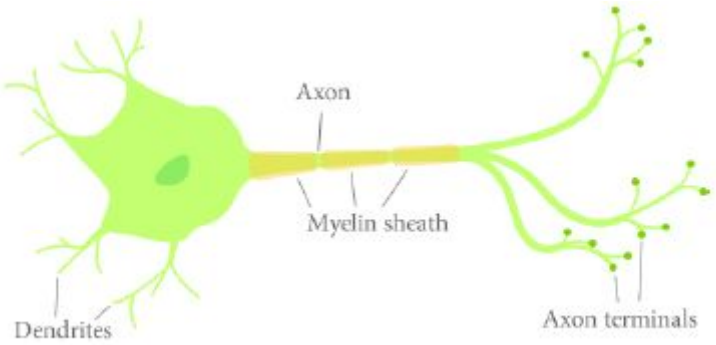


Figure 16: Un neurone réel [40]

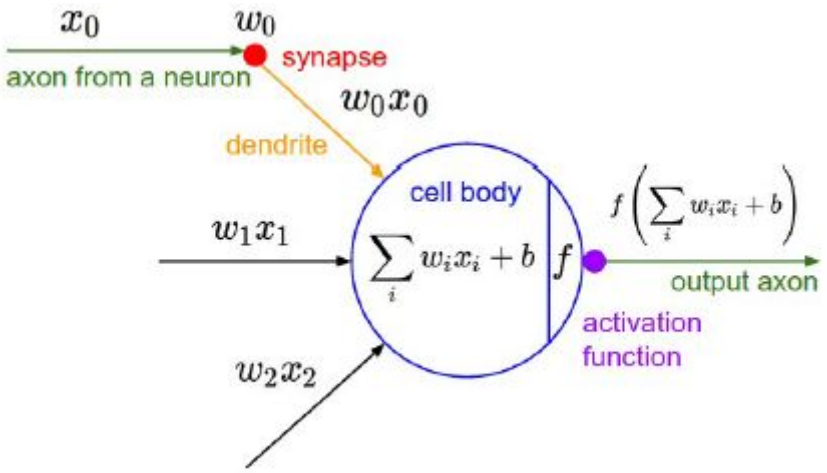


Figure 17: Un neurone artificiel [40]

Dans la figure 17, les Xi sont des valeurs numériques qui représentent soit les données d'entrée, soit les valeurs sorties d'autres neurones. Les poids Wi sont des valeurs numériques qui représentent soit la valeur de puissance des entrées, soit la valeur de puissance des connexions entre les neurones. Il existe des opérations qui se passent au niveau du neurone artificiel. Le neurone artificiel fera un produit entre le poids (w) et la valeur d'entrée (x), puis ajouter un biais (b), le résultat est transmis à une fonction d'activation (f) qui ajoutera une certaine non-linéarité. [40]

a. Les fonctions d'activation

La fonction d'activation est une composante essentielle du réseau neuronal. Ce que cette fonction a décidé est si le neurone est activé ou non. Il calcule la somme pondérée des entrées et ajoute le seuil. Il existe de nombreux types de fonctions d'activation, nous trouvons :

1. **La fonction Sigmoïde**

   Cette fonction est l'une des plus couramment utilisées. Elle est bornée entre 0 et 1, et elle peut être interprété stochastiquement comme la probabilité que le neurone s'active, et elle est généralement appelé la fonction logistique ou le sigmoïde logistique. Sa formule est

   $$f(x) = \frac{1}{1+e^{-x}}$$

   Le figure 18 montre la représentation graphique de la fonction Sigmoïde

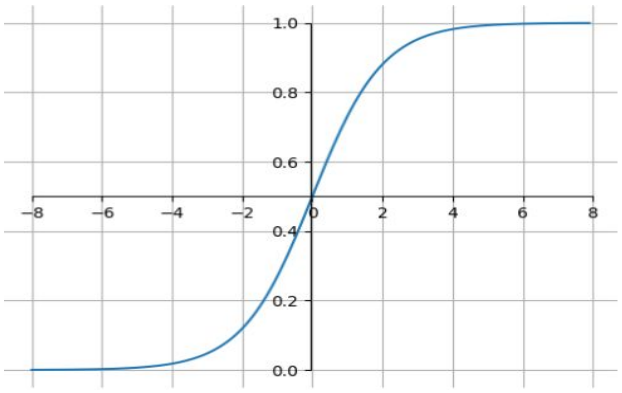


Figure 18: Illustration graphique de la fonction Sigmoïde [41]

2. La fonction Tanh

La fonction Tangente hyperbolique est une fonction trigonométrique hyperbolique, Tout comme la tangente représente un rapport entre les côtés opposés et adjacents d'un triangle rectangle, Tanh représente le rapport entre le sinus hyperbolique et le cosinus hyperbolique : $\tanh tanh\,(x) = \frac{x)}{\cosh cosh\,(x)}$

Contrairement à la fonction Sigmoïde, la plage normalisée de Tanh est comprise entre -1et 1. L'avantage de Tanh est qu'elle peut traiter plus facilement les nombres négatifs.

La figure 19 montre représentation graphique de la fonction Tanh.

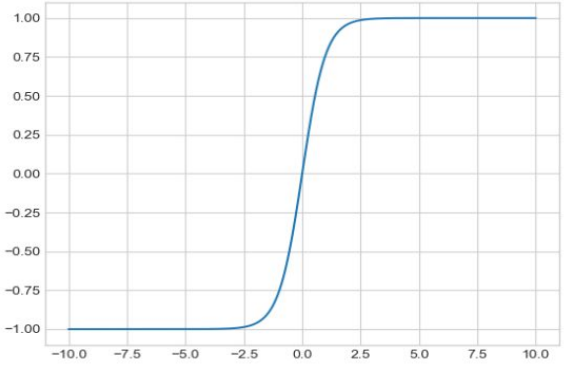


Figure 19: Illustration graphique de la fonction Tanh [41]

3. La fonction ReLu

La fonction ReLu est une transformation plus intéressante qui active un nœud uniquement si l'entrée dépasse une certaine quantité. Lorsque l'entrée est inférieure à zéro, la sortie est égale à zéro, mais lorsque l'entrée dépasse un

certain seuil, elle présente une relation linéaire avec la variable dépendante f(x)= max (0, x), et la figure 20 montre la représentation graphique de la fonction ReLu.

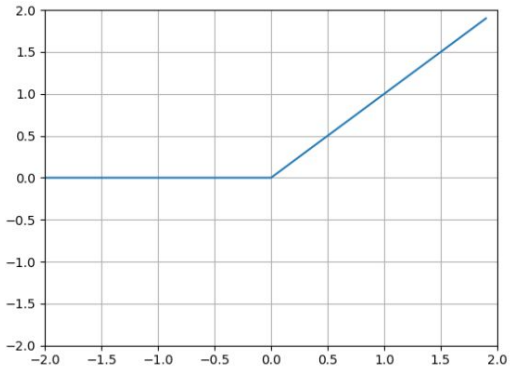


Figure 20: Représentation graphique de la fonction ReLu [41]

4. La fonction Softmax

C'est une généralisation de la régression logistique dans la mesure où il peut être appliqué à des données continues et peut contenir plusieurs limites de décision. Une représentation de sa sortie est donnée à la figure 21.

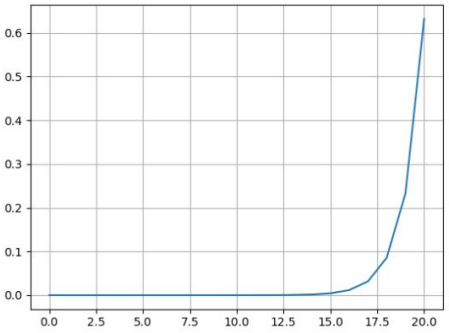


Figure 21: Représentation graphique de la fonction Softmax [41]

b. Rétropropagation du gradient

La phase d'apprentissage des MLP consiste à adapter les poids des connexions en fonction des erreurs de prédiction constatées à chaque classification d'une nouvelle instance. La rétropropagation du gradient (*backpropagation*) [42] est la méthode la plus utilisée pour l'adaptation desdits poids. Cet algorithme permet de déterminer le gradient de l'erreur pour

chaque neurone du réseau en partant de la dernière couche et en arrivant jusqu'à la première couche cachée.

L'objectif de la rétropropagation du gradient est d'ajuster les poids des connexions dans le but de minimiser l'erreur quadratique.

III.5 Les modèles de réseaux de neurones

Il existe beaucoup des modèles pour les réseaux de neurones et parmi ces modèles nous cotions

a. Réseau de neurone récurrent

Un réseau de neurones récurrent (RNN *Recurrent Neural Network*) est un réseau de neurones dont le graphe de connexion contient au moins un cycle, permettent ainsi aux informations historiques de persister dans les états cachés des RNN.

Les RNN sont donc particulièrement adaptés au traitement de données temporelles et à l'utilisation d'informations séquentielles. Cependant, la capacité des RNN vanille à traiter des séquences sur une longue période est très limitée en raison du problème de la disparition et de l'explosion des gradients. [43]

La représentation graphique de RNN est illustrée à la figure 22

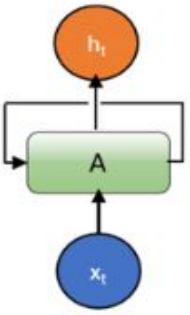


Figure 22: Illustration de la structure du RNN de base avec une boucle [44]

Une boucle permet de faire passer l'information d'une étape à l'autre du réseau. Un réseau neuronal récurrent peut être considéré comme une copie multiple du même réseau, chaque réseau transmettant un message à un successeur. Le schéma ci-dessous (Voir la figure 23) montre ce qui se passe si on déroule la boucle.

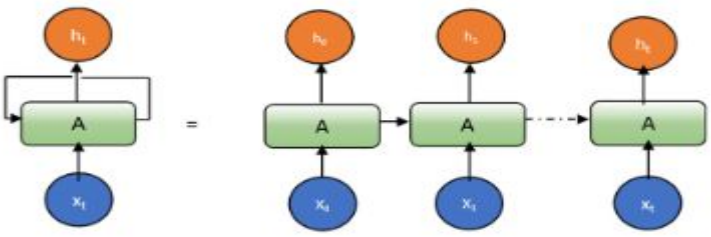


Figure 23: Illustration d'un un RNN déroulé. [44]

La caractéristique la plus importante des réseaux neuronaux récurrents est l'utilisation d'informations contextuelles entre les séquences d'entrée et de sortie. Cependant, l'accès au flux d'informations dans les architectures RNN est limité en pratique, en raison de la décroissance exponentielle de l'influence des entrées sur les couches cachées autour des connexions récurrentes. Ce problème est appelé le problème du gradient de disparition [45].

Au cours des dernières décennies, des groupes de recherche ont proposé des solutions pour surmonter les problèmes de disparition.

b. Réseau de neurone à long et court terme

Le réseau de neurone à long terme et à court terme LSMT définit par Hochreiter et Schmidhuber [46] est l'une des nombreuses variantes de l'architecture des réseaux neuronaux récurrents (RNN).

Le concept de LSTM est similaire à celui d'un RNN, est formée d'un ensemble de composants connectés de façon récurrente appelés blocs de mémoire. Chaque bloc de mémoire contient souvent une cellule de mémoire auto-connectée, des portes d'entrée, de sortie et d'oubli qui permettent la mise à jour du bloc donné. La figure 23 illustre un bloc de mémoire unique de LSTM.

Les portes multiplicatives permettent aux cellules de mémoire LSTM de stocker et d'accéder aux informations sur de longues périodes, atténuant ainsi le problème du gradient de disparition. Tant que la porte d'entrée reste fermée (c'est-à-dire qu'elle a une activation proche de 0), l'activation de la cellule ne sera pas écrasée par les nouvelles entrées arrivant dans le réseau et peut donc être mise à disposition du réseau beaucoup plus tard dans la séquence en ouvrant la porte de sortie.

La porte d'entrée et de sortie multiplie l'entrée et la sortie de la cellule tandis que la porte d'oubli multiplie l'état précédent de la cellule. Aucune fonction d'activation n'est appliquée dans une cellule donnée. La fonction d'activation de la porte est généralement le sigmoïde logistique, donc les activations de la porte sont comprises entre zéro et un. Les fonctions d'activation d'entrée et de sortie de la cellule sont tnah ou sigmoïde logistique. [47]

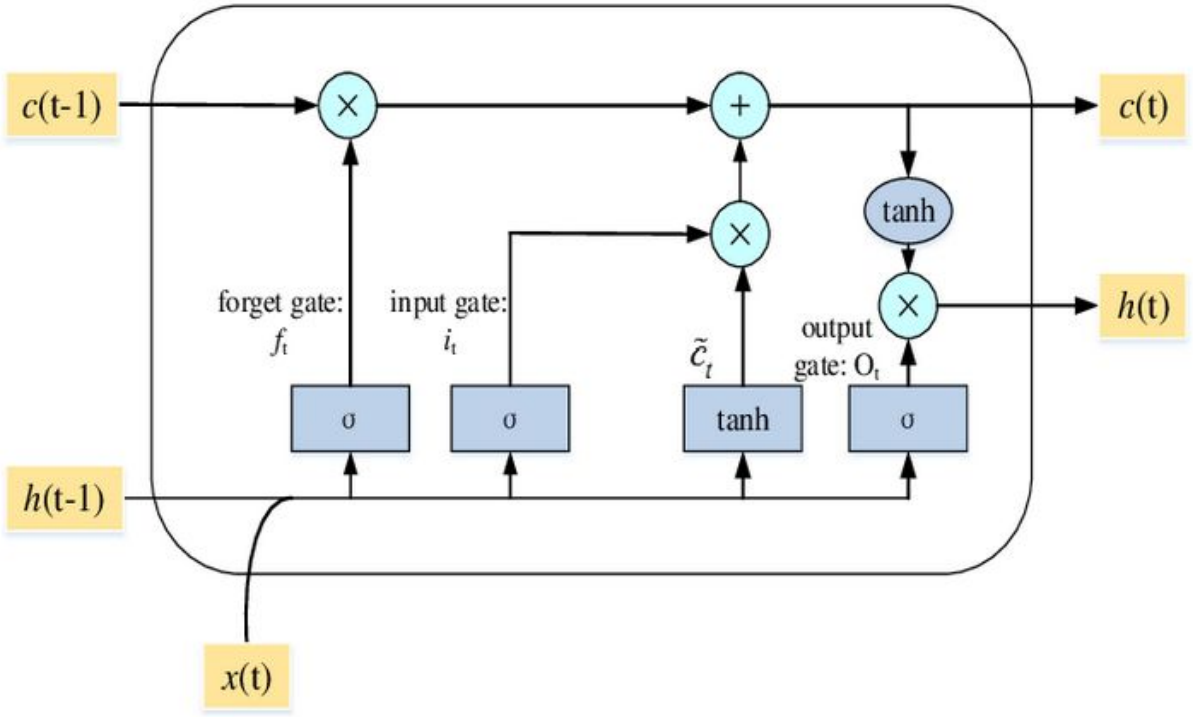


Figure 24: Illustration d'un bloc de mémoire LSTM avec une cellule.[48]

Le LSTM bidirectionnel est une extension de LSTM typiques qui peut améliorer les performances du modèle sur les problèmes de classification des séquences. Lorsque tous les pas de temps de la séquence d'entrée sont disponibles, les DB-LSTM entraînent deux LSTM au lieu d'un LSTM sur la séquence d'entrée. Le premier sur la séquence d'entrée telle quelle et l'autre sur une copie inversée de la séquence d'entrée.

La structure bidirectionnelle du LSTM traite la séquence dans les directions avant et arrière, ce qui permet de tirer des enseignements des changements passés et futurs dans la séquence et les résultats sont plus rapides. Habituellement, le LSTM fournit une sortie à différents intervalles de temps qui sont décidés par l'activation sigmoïde de la porte de sortie.

Cependant, nous avons utilisé la sortie de l'état final de LSTM qui présente une séquence traitée complète qui est ensuite envoyée au classificateur Softmax pour la prédiction finale.

c. Réseau de neurone convolutif

Les réseaux de neurones convolutif (CNN ou ConvNet) opèrent généralement sur des images et sont l'un des types de réseaux de neurones le plus répandu, il s'agit d'une forme particulière de MLP. [49] Ces réseaux sont capables de catégoriser les informations des plus simples aux plus complexes (figure 25). Ils consistent en un empilage multicouche de

neurones, des fonctions mathématiques à plusieurs paramètres ajustables, qui prétraitent de petites quantités d'informations.

Un réseau de neurones convolutif se compose de deux parties essentielles, ou chaque partie à un rôle à jouer et un objectif à remplir, la première partie (*feature extraction*) se charge de l'extraction des caractéristiques, des couches de convolution et des couches de sous-échantillonnage y sont alterné dedans tandis que la seconde partie effectue la classification en fonction des caractéristiques extraite dans la partie précédente.

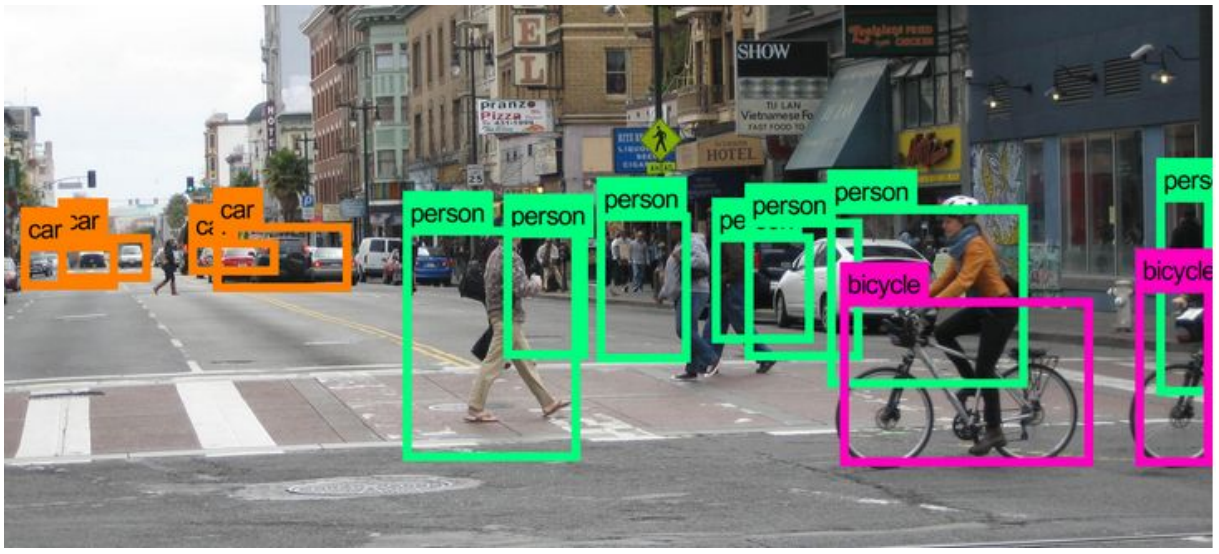


Figure 25: Illustration CNN pour la reconnaissance des objets [50]

L'architecture globale du réseau neuronal convolutif (CNN) est représenté dans la figure26 comprend une couche conventionnelle, plusieurs couches de convolution et de regroupement maximum alternés, une couche entièrement connectée et une couche de classification.

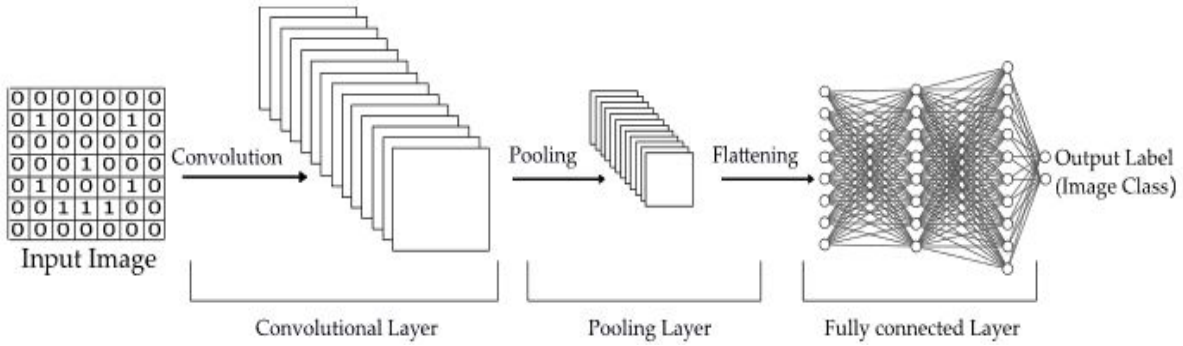


Figure 26: L'architecture de CNN [51]

- **Couche convolutif**

Dans cette couche, chaque filtre (également appelé "Kernel") est appliqué à l'image dans des positions successives le long de l'image et par des opérations de convolution, génère une carte des caractéristiques.

Ces filtres ont des dimensions spatiales (largeur, hauteur) et une dimension de profondeur, et différents filtres peuvent être utilisés dans différentes parties du réseau, les filtres sont appliqués à l'entrée de la même manière qu'une fenêtre coulissante et une opération de multiplication avec la valeur d'entrée sont effectuées avec les filtres (voir la figure 27)

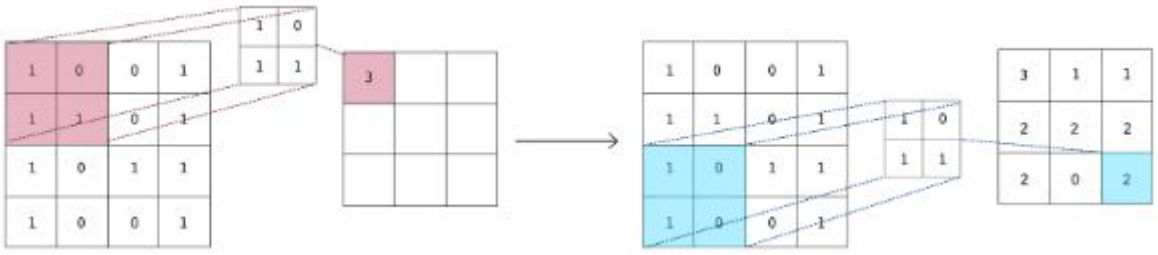


Figure 27: Exemple de convolution avec un filtre de 2x2 appliqué à une image 4x4x1 [51]

Le résultat de cette multiplication est ensuite suivi d'une opération non linéaire, nous appelons cette fonction d'activation.

- **Couche non linéaire** : une fonction d'activation non linéaire, telle que la fonction ReLu, est utilisée pour éviter la linéarité dans le système.

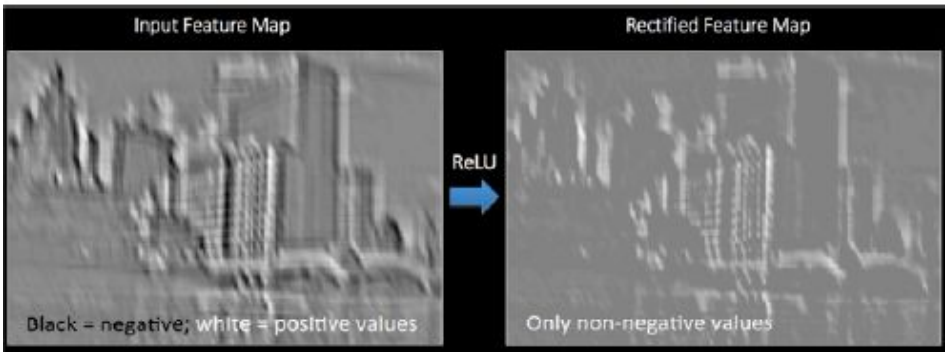


Figure 28: Application de la fonction d'activation ReLU [8]

- **Couche Pooling (sous-échantillonnage)**

Une autre composante importante d'un CNN. En bref, le pooling est une fonction qui permet de sous-échantillonner la sortie d'une autre couche. Cela permet de réduire la dimensionnalité des dimensions spatiales, ce qui réduit le temps de traitement.

La figure 29 montre un exemple des deux opérations de pooling les plus courantes, le pooling maximale et le pooling moyenne. Dans le pooling maximale, une partie rectangulaire de la carte des caractéristiques est réduite à la valeur maximale à l'intérieur de celle-ci. La même opération est effectuée dans le pooling moyenne, mais la moyenne est calculée au lieu de la valeur maximale. [52]

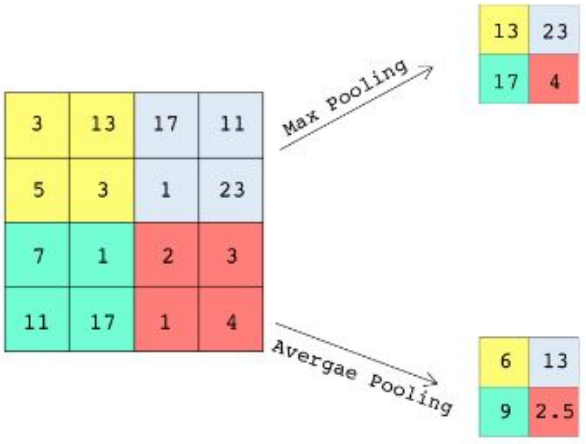


Figure 29: Exemple de Pooling maximale et moyenne des opérations de Pooling. [52]

- **Couches entièrement connectées**

La principale caractéristique des couches entièrement connectées est que chaque neurone est connecté à tous les neurones (c'est-à-dire les activations) de la couche précédente. Dans la dernière couche, la sortie de la couche précédente est donnée en entrée à la couche entièrement connectée ; ensuite, ces couches aplatissent l'entrée donnée à un vecteur à N dimensions où N est le nombre de classes dans le problème de classification. Le vecteur est ensuite transmis à un classificateur tel qu'un KNN ou une couche softmax qui prédit l'étiquette. [52].

d. Les différents modèles de CNN

Nous avons cité 3 modèles de CNN utilisé dans notre approche

1. Le modèle AlexNet

Alex-Net est formé pour classer les 1,2 million d'images de la base de données image-Net en 1000 classes différentes, définit par Krizhevsky et ses camarades en 2012. [53]

AlexNet, en tant qu'architecture de réseau neuronal convolutif (CNN) relativement simple, a obtenu un grand succès dans les tâches de classification des scènes et s'est révélé être une excellente technique de classification hiérarchique et automatique des scènes.

Nous décrivons ci-dessous certaines des caractéristiques nouvelles de réseau.

- **Non-linéarité de ReLU**

AlexNet utilise la fonction linéaire rectifiées (ReLU) au lieu de la fonction Tanh, qui était standard à l'époque. L'avantage de la ReLU réside dans le temps de formation.

La figure montre que ReLU (ligne continue) atteint un taux d'erreur de formation de 25% six fois plus vite qu'un réseau équivalent avec des anévrons (ligne pointillée).

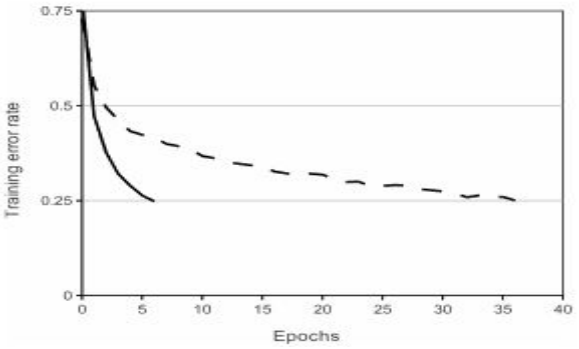


Figure 30: Taux d'apprentissage [53]

- **Plusieurs GPU.**

À l'époque, les GPU fonctionnaient encore avec 3 giga octets de mémoire. C'était d'autant plus grave que le jeu d'entraînement comportait 1,2 million d'images. AlexNet permet l'entraînement multi-GPU en mettant la moitié des neurones du modèle sur un GPU et l'autre moitié sur un autre GPU. Cela permet non seulement de former un modèle plus grand, mais aussi de réduire le temps de formation.

- **Mise en commun par chevauchement**. (*Overlapping Pooling*)

Les CNN regroupent traditionnellement les sorties de groupes de neurones voisins sans chevauchement. Toutefois, lorsque les auteurs ont introduit le chevauchement, ils ont constaté une réduction de l'erreur d'environ 0,5 % et ont constaté que les modèles avec mise en commun chevauchante ont généralement plus de mal à se chevaucher.

- **Augmentation des données.**

Les auteurs ont utilisé une transformation préservant l'étiquette pour rendre leurs données plus variées. Plus précisément, ils ont généré des traductions d'images et des réflexions horizontales, ce qui a multiplié par 2048 l'ensemble de la formation. Ils ont également effectué une analyse en composantes principales (ACP) sur les valeurs des pixels RVB (RGB) pour modifier les intensités des canaux RVB, ce qui a réduit le taux d'erreur de plus de 1 % dans le top 1

- **Abandon** *(Dropout)*.

Cette technique consiste à éteindre les neurones avec une probabilité prédéterminée (par exemple 50%). Cela signifie que chaque itération utilise un échantillon différent des

paramètres du modèle, ce qui oblige chaque neurone à avoir des caractéristiques plus robustes qui peuvent être utilisées avec d'autres neurones aléatoires. Cependant, l'abandon augmente également le temps de formation nécessaire à la convergence du modèle.

- **L'architecture globale de AlexNet**

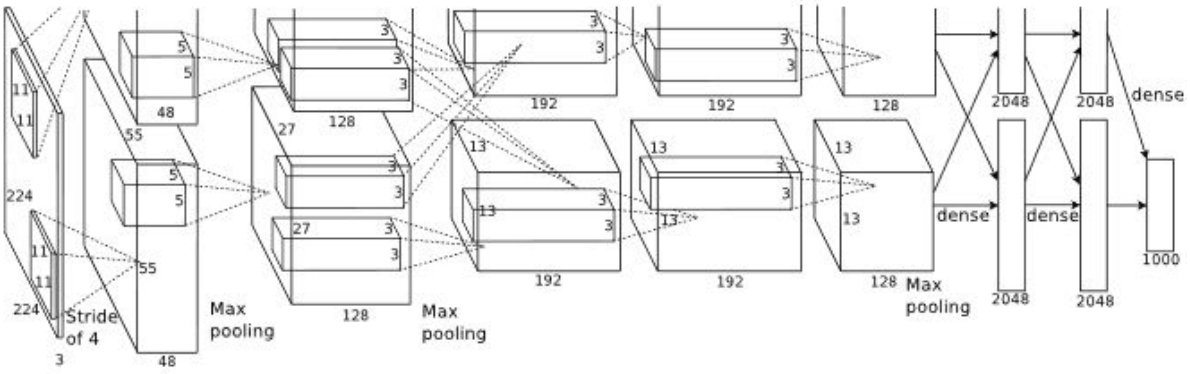


Figure 31: Illustration de l'architecture de AlexNet [53]

Le réseau contient huit couches avec des poids, la couche d'entrée est la première couche qui définit les dimensions d'entrée. Les couches intermédiaires constituent l'essentiel du réseau Alex-Net. Ces couches sont constituées de séries de cinq couches convolutives, Les noyaux des deuxième, quatrième et cinquième couches convolutifs sont uniquement connectés aux cartes de noyaux de la couche précédente qui résident sur le même GPU (Voir la figure 35), Les noyaux de la troisième couche convolutif sont connectés à toutes les cartes de noyaux de la deuxième couche, À côté de ces couches, trois couches entièrement connectées sont connectées à tous les neurones de la couche précédente. La couche de classification est la couche finale. La ReLU non-linéarité est appliquée à la sortie de chaque couche convolutive et entièrement connectée.

La première couche convolutif filtre l'image de (longueur × hauteur × largeur) entrées avec 96 noyaux de taille 11×11×3 avec une enjambée de 4 pixels. La deuxième couche convolutif prend en entrée la sortie de la première couche convolutif et la filtre avec 256 noyaux de taille 5×5×48. Les troisième, quatrième et cinquième couches convolutifs sont reliées entre elles sans aucune couche de mise en commun. La troisième couche convolutif comporte 384 noyaux de taille 3×3×256 connectés aux sorties (mises en commun) de la deuxième couche convolutif. La quatrième couche convolutif a 384 noyaux de taille 3×3×192, et la cinquième couche convolutif a 256 noyaux de taille 3×3×192. Les couches entièrement connectées ont chacune 4096 neurones (unité de sortie).

La dernière couche de l'architecture AlexNet produit un vecteur de caractéristiques de 1×1000 pour une seule image.

2. Le modèle GoogleNet

GoogleNet est présenté dans les travaux de Szegedy et al [54] en 2014.

GoogleNet possède sept millions de paramètres et contient neuf modules de départ, quatre couches convolutifs, quatre couches de regroupement maximum, trois couches de regroupement moyen, cinq couches entièrement connectées et trois couches softmax pour les principaux classificateurs auxiliaires du réseau. En outre, il utilise la régularisation des abandons dans la couche entièrement connectée et applique l'activation ReLU dans toutes les couches convolutifs. Cependant, ce réseau est beaucoup plus profond et plus large, avec un total de 22 couches, mais il a un nombre de paramètres de réseau beaucoup plus faible que celui d'AlexNet.

Cette architecture utilise 3 filtres de taille différente (c'est-à-dire 1x1, 3x3, 5x5) pour la même image et combine les caractéristiques pour obtenir une sortie robuste. La convolution 1x1 est introduite pour la réduction des dimensions. Cette architecture trouve le meilleur poids lors de l'entraînement du réseau et sélectionne naturellement les caractéristiques appropriées.

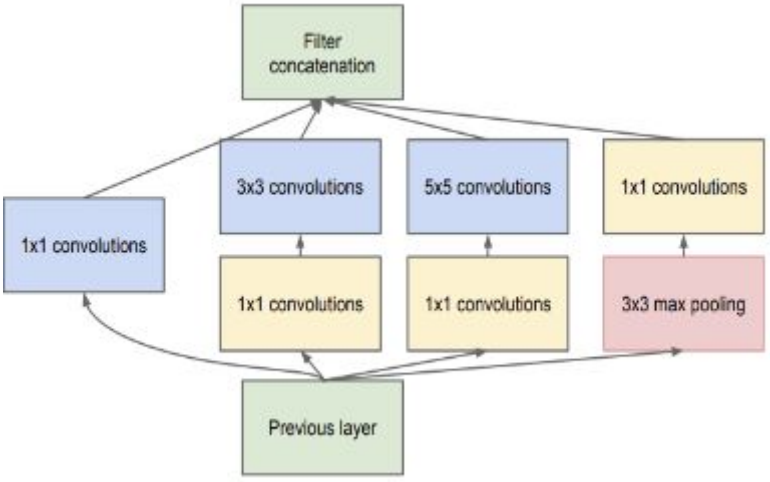


Figure 32: Inception avec la réduction de la dimensionnalité [54]

La figure 32 illustre la convolution multiple avec un filtre 1x1, un filtre 3x3, un filtre 5x5 et une couche max-pooling.

3. Le modèle Inception version2

Inception V2 [55] est une architecture convolutif profonde largement utilisée pour les tâches de classification. Le concept modèle a été introduit par Szegedy dans l'architecture GoogleNet, où Inception V2 a été proposé en mettant à jour le module d'inception V1. Le réseau Inception V2 est constitué de plusieurs blocs de construction symétriques et asymétriques, chaque bloc ayant plusieurs branches de convolutions, un regroupement moyen, un regroupement maximal, une concaténation, des abandons et des couches entièrement connectées. Ce réseau compte 42 couches au total et possède 29,3 millions de paramètres, ce qui signifie que le coût de calcul n'est qu'environ 2,5 fois plus élevé que celui de GoogleNet. Enfin, les auteurs ont conclu que la combinaison d'un nombre de paramètres plus faible et d'une régularisation supplémentaire avec un lissage des étiquettes des classificateurs auxiliaires normalisés par lots permet de former un réseau de haute qualité sur des ensembles de formation de taille relativement modeste.

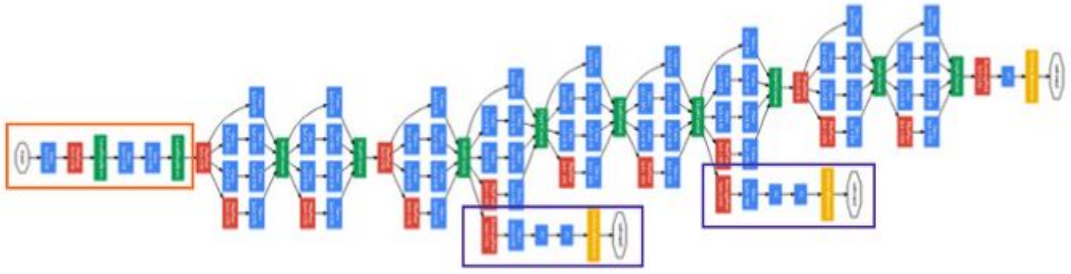


Figure 33: Architecture globale d'Inception [56]

La figure 33 montre le module de base de l'architecture globale du réseau. Il existe plusieurs modules de départ combinés pour former un réseau plus profond permettant d'obtenir une grande précision.

III.6 Conclusion

Dans ce chapitre, nous avons donné une vue d'ensemble sur l'apprentissage automatique, ainsi que sur ses différents type (apprentissage supervisé, apprentissage non supervisé, apprentissage semi-supervisé et apprentissage par renforcement). Nous avons réalisé aussi une étude détaillée sur les réseaux de neurones RNN et CNN. Ils ont pu donner des résultats performants pour les tâches de traitement d'images et traitement vidéo.

CHAPITRE IV : L'APPROCHE PROPOSÉE

IV.1 Introduction

Les réseaux de caméras de surveillance sont partout de nos jours. Le volume de données collectées par un tel réseau de capteurs de vision déployés dans de nombreux contextes allant des besoins de sécurité à la surveillance de l'environnement ce qui répond clairement aux exigences des grandes masse de données (Voir le figure 34)

Les difficultés d'analyse et le traitement de ces grandes données vidéo sont évidentes chaque fois qu'il y a un incident qui nécessite de fouiller dans de vastes archives vidéo pour identifier les événements d'intérêt.

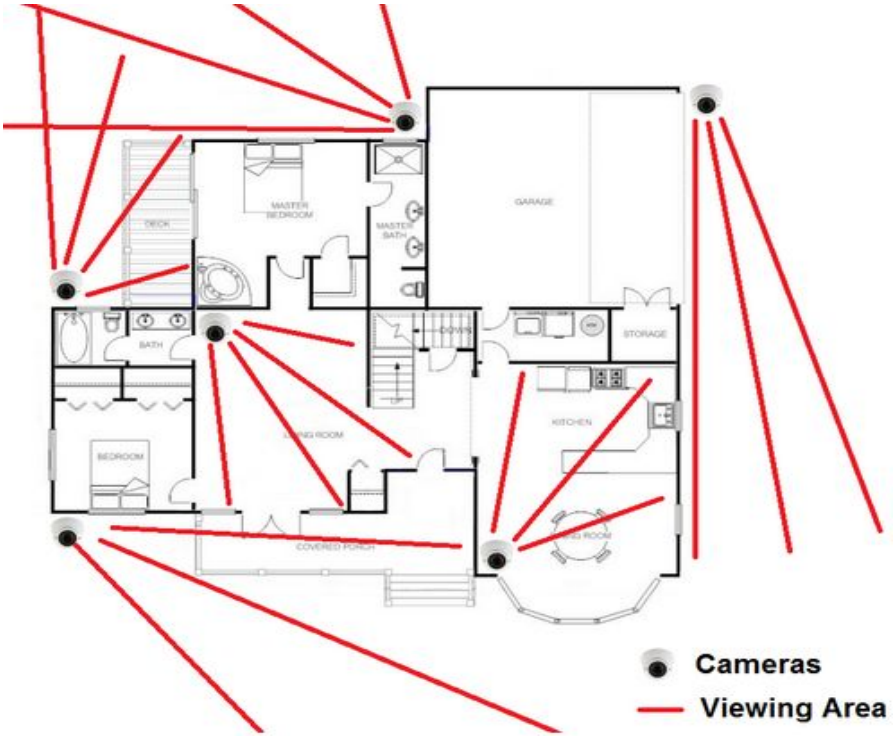


Figure 34: Illustration d'un réseau de caméra multi vues [57]

Une solution à ce problème est donc de créer automatiquement un résumé de la vidéo permet de répondre à ce besoin en fournissant un aperçu général et rapide de l'ensemble du contenu audiovisuel de la vidéo originale et en présentant les parties intéressantes.

À travers ce chapitre nous allons présenter, et expliquer en détail, les principes de fonctionnement de notre approche utilisée pour la création automatique de résumé vidéo tiré de plusieurs vues d'une même scène et qui est basée sur *deep learning*.

IV.2 Vue globale de l'approche

Un aperçu global de notre approche pour la génération du résumé vidéo est illustré dans la figure ci-dessous.

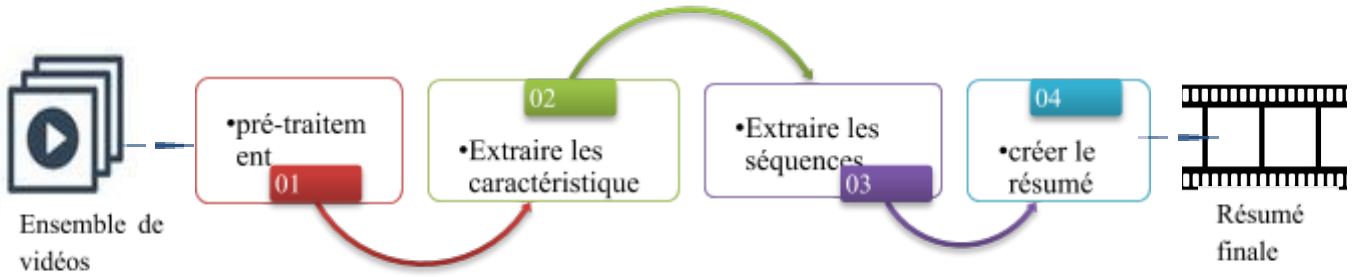


Figure 35: Illustration de schéma globale de notre approche

Dans ce qui suit nous décrivons les différentes phases que constitue notre approche :

1. Phase de pré-traitement
2. Phase d'extraction des caractéristiques profondes
3. Phase d'extraction des séquences frames
4. Phase de création du résume

IV.2.1 Phase de pré-traitement

Dans notre approche, une étape de prétraitement des vidéos d'entrées est nécessaire pour la création automatique du résumé. Cette phase consiste à l'extraction des frames de chaque vidéo, puis de les redimensionner. Cette phase permet d'optimiser le temps de calcul. (Voir la figure 36 illustre ces différentes étapes)

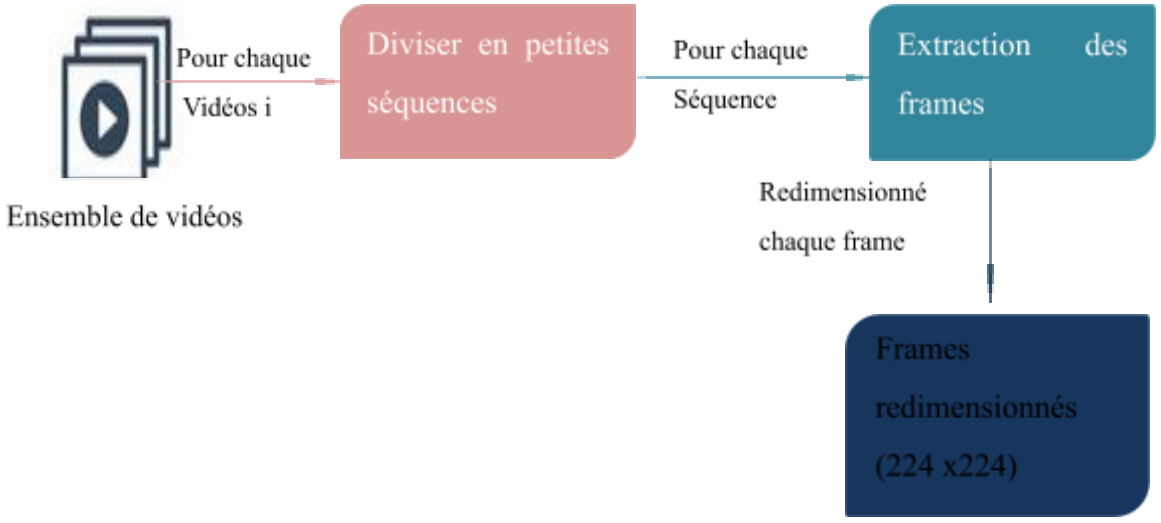


Figure 36: Schéma globale de la phase prétraitement

1. Pour chaque vidéo i parmi l'ensemble des vidéos d'entrée,

2. Segmentation de la vidéo en petite séquence d'une durée déterminé (dans notre étude, nous avons choisis une durée t = 20 secondes),
3. Extraction des frames de chaque segments vidéo,
4. Redimensionnement des frames de chaque segments vidéo (dans notre étude, nous avons choisi les résolutions 224 x 224)

À noter que le résumé final sera construit en rassemblant les segments vidéo les plus représentatifs extraits de la source vidéo (contenant les images-clés)

IV.2.2 Phase d'extraction les caractéristiques profondes

Cette phase consiste à extraire, analyser et de caractériser l'information brute présente sur les images des segments vidéo afin d'identifier les éléments qui les constituent. Le rôle principal de ces caractéristiques en vision par ordinateur est de transformer l'information visuelle sous formes de vecteurs caractéristiques. Le schéma global de cette phase est illustré dans la figure37.

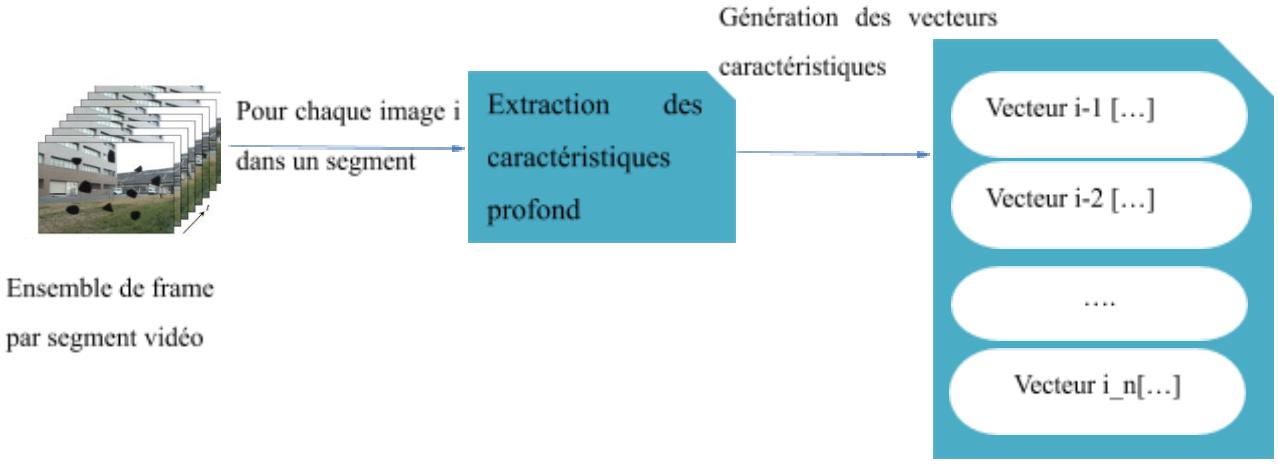


Figure 37: Schéma globale de la phase d'extraction des caractéristiques profondes

Afin de construire ces vecteurs caractéristiques des vidéos, nous avons opté pour une architecture profonde obtenue en entraînant un réseau convolutif tridimensionnel : AlexNet, GoogleNet et Inception v2 ils sont détaillé dans le chapitre présidente.

Bien que l'extraction de caractéristiques profondes nécessite une grande puissance de calcul, dans le cadre que nous proposons, nous effectuons cette étape sur Google Colab pour économiser du temps.

IV.2.3 Extraction des séquences frames

Une fois que toutes les caractéristiques profondes sont extraites, nous avons utilisé un autre type de réseau de neurone récurrent c'est le LSTM-bidirectionnel (Voir le détail dans le chapitre présidente)

Dans le cadre proposé, nous avons empilé deux couches de LSTM l'une sur l'autre pour former un LSTM bidirectionnelle qui aide à l'apprentissage des changements à long terme.

Les caractéristiques profondes sont extraites d'une séquence des frames au temps T et se propagent vers RNN pour savoir si une séquence de frames est informative ou non.

Enfin, nous obtenons un modèle formé qui est capable de classer les séquences comme informatives ou non informatives, ce qui aide à la génération du résumé final.

IV.2.4 Résumé final

Bien que les skims finalistes soient suffisamment importants, il est possible que certaines séquences visuellement similaires soient sélectionnées comme skims résumés. Ce problème rend difficile l'étape de génération du résumé final qui est abordée dans la technique proposée en ne sélectionnant comme résumé final que les séquences dont la probabilité est maximale pour la classe d'informativité.

Ainsi, les images les mieux ajustées et les plus probables sont prises en compte lors de l'étape de post-traitement, ce qui permet d'obtenir un résumé diversifié et représentatif. Les résumés finaux avec les probabilités peuvent être observés à partir de la figure 38, où les images avec les probabilités maximales d'une séquence font partie du résumé final.

Exemples de cadres pour chaque vue générée par le cadre que nous proposons.

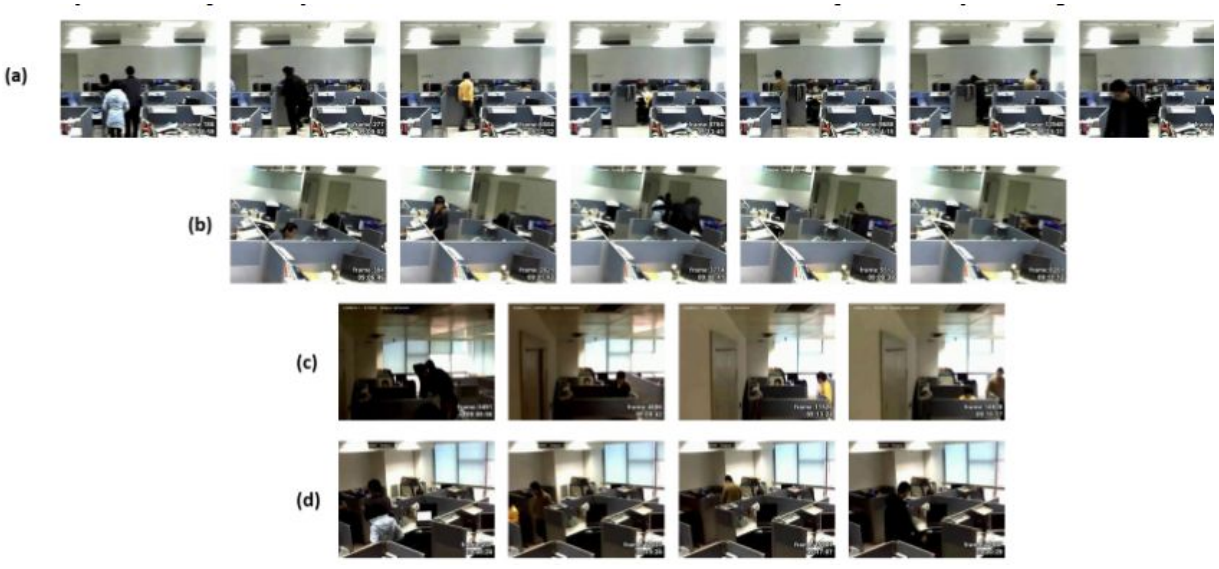


Figure 38: Illustration des images de résume final

IV.3 Conclusion

Dans ce chapitre, nous avons détaillé notre approche de résumé vidéo, basée sur l’apprentissage profond en utilisant une architecture neuronale basée sur les réseaux de neurones convolutifs pour l’extraction des caractéristiques profondes de chaque frame d'une séquence et les transmet à un autre réseau de neurone RNN pour acquérir des probabilités d'information et générer un résumé dynamique d’une vidéo.

CHAPITRE V : TESTS ET DISCUSSION

V.1 Introduction

Après avoir défini notre approche de création automatique de résumé vidéo ainsi que tous ces concepts liés, nous passons maintenant à l'expérimentation. Ce chapitre décrit l'environnement matériel et logiciel utilisé pour notre travail, ainsi que le jeu de test sur lequel nous avons travaillé et les mesures utilisées. Enfin nous terminerons ce chapitre par la présentation des résultats obtenus durant les tests.

V.2 Outils de développement et langage de programmation

Nous avons décrit les outils ainsi le langage utilisé pour la réalisation de notre application.

V.2.1 Outils de développement

Notre application va être réalisée sur une machine qui comporte les caractéristiques suivantes :

- **Marque :** n/a
- **Processeur :** Ryzen
- **Carte graphique :** Amd Radeon RX series
- **Mémoire :** 16 GB
- **Stockage :** 1TB
- **Système d'exploitation :** Windows 10, 64bits

Nous avons aussi utilisé l'environnement Google Colab

- **Google Colab** 

C'est un service cloud, offert par Google (gratuit), basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l'apprentissage automatique. Cette plateforme permet d'entraîner des modèles de *Machine Learning* directement dans le cloud. Sans donc avoir besoin d'installer quoi que ce soit sur notre ordinateur à l'exception d'un navigateur.[58]

V.2.2 Langage de programmation

- **Python** 

Python est un langage de programmation interprété (à ne pas confondre avec un langage compilé) créé par le *néerlandais Guido Van Rossum* au *Centrum voor Wiskunde* aux Pays-Bas en 2001. Le langage Python peut être exécuté directement sans passer par une phase de compilation. Il est possible de traduire un programme dans un langage (byte code) qui est ensuite interprété par une machine virtuelle Python (mécanisme semblable au langage Java). Python est principalement inspiré du langage ABC (indentation comme syntaxe, …), mais aussi du langage C et des outils Unix. Python est un langage libre placé sous licence PSFL (*Python Software Foundation License*), il fonctionne sur de nombreuses plates-formes avec une grande communauté active. Python est aussi un langage orienté objet, il gère l'héritage de classe ainsi que l'héritage multiple (hérite de plusieurs classes) et le polymorphisme. [59]

Quelques avantages du langage Python :

- Proche du langage C.
- Proche des langages fonctionnels.
- Pas de perte de temps pour déclarer les types, variables, …
- Types de données complexes intégrés (listes, …)
- Permet d'intégrer d'autres codes cibles.
- Possède un garbage collector (permettant de ne pas gérer les fuites de mémoire)

- **NumPy** 

NumPy est une extension du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux. Plus précisément, cette bibliothèque logicielle libre et open source fournit de multiples fonctions permettant notamment de créer directement un tableau depuis un fichier ou au contraire de sauvegarder un tableau dans un fichier, et manipuler des vecteurs, matrices et polynômes. [60]

- **Sklearn** 

Scikit-learn (Sklearn) est la bibliothèque la plus utile et la plus robuste pour l'apprentissage machine en Python. Elle fournit une sélection d'outils efficaces pour l'apprentissage machine et la modélisation statistique, y compris la classification, la régression, le regroupement et la réduction de la dimension via une interface de cohérence en Python.

Il s'appelait à l'origine scikits-learn et a été initialement développé par *David Cournapeau* dans le cadre du *projet Google summer of code* en 2007. Plus tard, en 2010, *Fabian Pedregosa*, et son groupe, de la FIRCA (Institut français de recherche en informatique et en automatique), ont porté ce projet à un autre niveau et en ont fait la première version publique (v0.1 beta) le 1er février 2010. Le projet repose aujourd'hui sur un effort mondial en code source ouvert rassemblant plus de 200 contributeurs. [61]

- **TensorFlow** 

TensorFlow est un outil open source d'apprentissage automatique développé par Google. Le code source a été ouvert le 9 novembre 2015 par Google et publié sous licence Apache. Il est basé sur l'infrastructure *DistBelief,* initiée par Google en 2011, et est doté d'une interface Python. TensorFlow est l'un des outils les plus utilisés dans le domaine d'intelligence artificiel et l'apprentissage machine. [62]

- **OpenCV** 

OpenCV (*Open Computer Vision*) est une bibliothèque graphique libre, initialement développée par Intel, spécialisée dans le traitement d'images et de vidéos en temps réel. Elle propose un ensemble de plus de 2500 algorithmes de vision par ordinateur ce qui permettra de donner à une machine le pouvoir d'analyser, traiter et comprendre une ou plusieurs images prises par un système d'acquisition, accessibles au travers d'API pour les langages C, C++, et Python. Elle est distribuée sous une licence BSD (libre) pour les plateformes Windows, GNU/Linux, Android et MacOS. [63]

- **SciPy** 

SciPy « Sigh Pie » est un logiciel open source pour les mathématiques, les sciences et l'ingénierie. La bibliothèque SciPy est conçue pour fonctionner avec les tableaux NumPy et

fournit de nombreuses routines numériques conviviales et efficaces, telles que des routines d'intégration et d'optimisation numériques. Ensemble, elles fonctionnent sur tous les systèmes d'exploitation courants, sont rapides à installer et sont gratuites. NumPy et SciPy sont faciles à utiliser, mais suffisamment puissants pour être utilisés par certains des plus grands scientifiques et ingénieurs du monde. [64]

- **OS**

Le module OS en Python fournit un large éventail de méthodes utiles pour manipuler des fichiers et des répertoires, permettant de créer et de supprimer un répertoire, d'en récupérer le contenu, de modifier et d'identifier le répertoire courant, etc. [65]

- **Caffe** Caffe

Il s'agit d'un cadre d'apprentissage approfondi introduit dans en 2014], créé par le *Berkeley Vision et LearningCenter* (BVLC), UC *Berkeley*. Il réduit le travail de l'utilisateur en lui permettant de définir des réseaux neuronaux profonds complexes. Il prend en charge divers types d'entrées, notamment des listes d'images brutes, des données multidimensionnelles, LevelDB, HDF5, etc. Il fournit une bibliothèque MATLAB et une bibliothèque Python pour l'interaction avec d'autres environnements. [66]

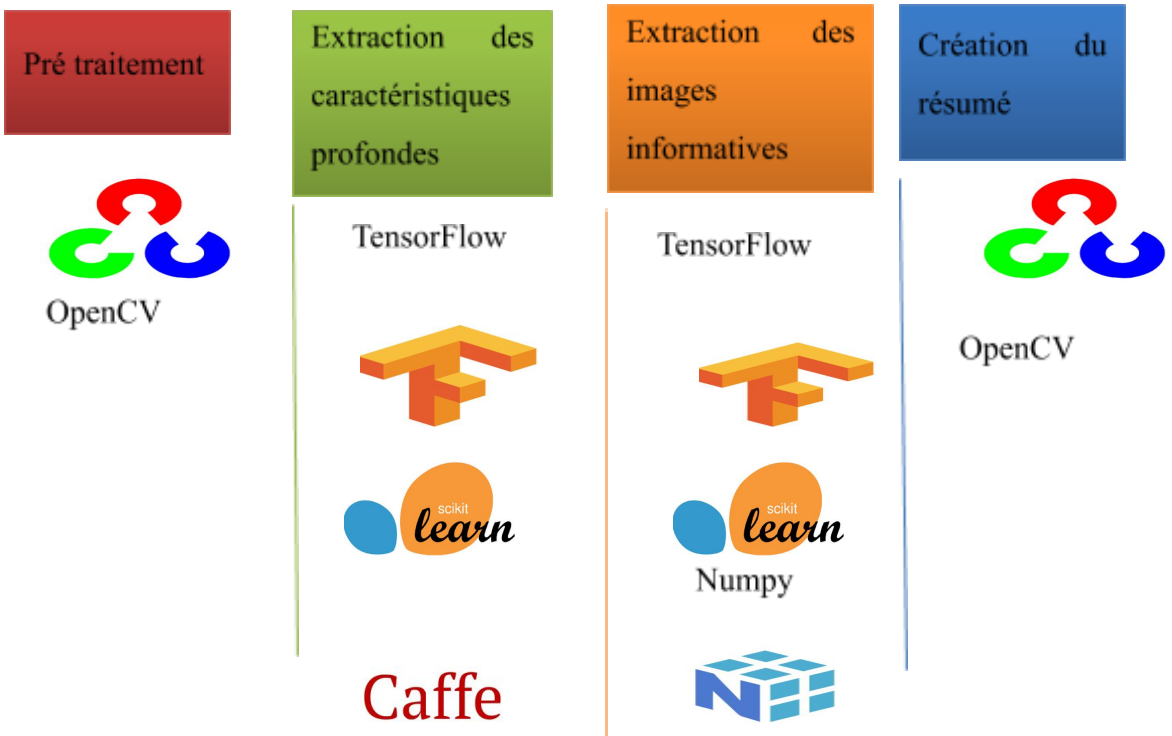


Figure 39: les bibliothèques pour chaque phase

La figure 39 présente les bibliothèques utilisées pour chaque phase de notre approche.

V.3 Ensemble de donnée (Dataset)

Dans ce qui suit, nous présentons les trois jeux de données vidéo multi-vues de notre dataset.

V.3.1 Office1

Il s'agit du dataset Office fourni par [26], qui a été pris avec 4 caméras stables mais non fixes dans un bureau (voir figure 40). Les vibrations des caméras et le changement des conditions d'éclairage rendent difficile la production d'un bon résumé vidéo. Les quatre vidéos ne sont pas synchronisées, et certaines d'entre elles souffrent même d'une fréquence d'images instable.

Figure 40: Une image d'exemple du dataset Office [67]

V.3.2 Campus

Où 19 caméras de surveillance qui sont installées au 7ème étage du bâtiment *BerryLam* de l'Université Nationale de *Taiwan*, qui couvre l'ensemble du couloir et l'une des salles de bureaux. Toutes ces vidéos ont une durée de sept minutes et dix secondes (07 :10).

Figure 41: Une image d'exemple du dataset Campus [67]

V.3.3 Lobby

Il s'agit de l'ensemble de donnée Lobby fourni par [26], qui a été pris avec 3 caméras dans un grand hall d'entrée. Cet ensemble de données a également été pris avec des caméras stables mais non fixes. Toutes les caméras sont synchronisées. Par rapport à l'ensemble de données de l'Office, cet ensemble de données contient plus de scènes surpeuplées avec des activités plus riches (voir figure 42), ce qui rend plus difficile à résumer.

Figure 42:Une image d'exemple du dataset Lobby [67]

V.4 Les mesures d'évaluation

Afin de mesurer la qualité de notre résumé vidéo. Nous s'avons intéressé à trois mesures souvent utilisées dans les travaux liés à l'apprentissage automatique, à savoir le rappel (*recall)*, la précision et la F-mesure. Afin de les calculer, nous définissons les valeurs dans le tableau suivant : A, B, C, D sont des valeurs

	Nombre des frames pertinentes	**Nombre des frames non pertinentes**	**Total**
Nombre des frames Retrouvé	A	B	a+b
Nombre des frames non Retrouvé	C	D	c+d
Total	a+c	b+d	a+b+c+d

Tableau 2: Calcul des paramètres Rappel, Précision et F-mesure

Où

- ✔ **Retrouvé** : signifie que les frames existent dans le résumé proposé.

- ✔ **Pertinent** : signifie que les frames existent dans le résumé généré (créé)

✔ **Rappel** : Le rappel mesure la capacité du système à restituer l'ensemble de frames pertinents. (Rappel exact par rapport à l'ensemble de frames retrouvées), obtenue en calculant [68]

$$Rappel = \frac{Nombre\ de\ f\ rames\ pertinents\ retrouve}{nombre\ de\ f\ rames\ pertinente} = \frac{a}{a+c}$$

✔ **Précision** : Mesure la capacité du système à ne restituer que des frames pertinents [69], elle est définie comme suit :

$$Présicion = \frac{Nombre\ de\ f\ rames\ pertinents\ retrouve}{nombre\ de\ f\ rames\ retrouvs} = \frac{a}{a+b}$$

✔ **F-Mesure** Mesure qui combine le rappel et la précision. En effet, le rappel et la précision ont tendance à varier en sens inverse. [69], elle est définie comme suit :

$$F - mesure = 2 * \frac{Rappel * Présicion}{Rappel + Présicion}$$

V.5 Résultats et discussion

Dans cette section, nous présentons diverses expériences et comparaisons pour valider l'efficacité et l'efficience des algorithmes que nous proposons pour le résumé vidéos multi-vues.

V.5.1 La matrice de comparaison

Nous nous sommes intéressés à trois mesures souvent utilisées dans les travaux liés à l'apprentissage automatique, à savoir le rappel, la précision et la F-mesure tout en se basant sur précédents entraînements par d'autres travaux sur des situations similaires [70]. En addition nous avons calculé la fonction de perte, et temps d'exécution pour l'entraînement, Afin de les calculer, nous définissons les valeurs dans le tableau suivant qui montrent les résultats de résumé vidéo sur les trois jeux de données multi-vues « Office, Campus et Lobby ».

1. **Le modèle AlexNet**

Dans ce cas notre modèle est basé sur le modèle AlexNet

Le tableau 3 montre les résultats de résumé vidéo sur les trois jeux de données multi-vues « Office, Campus et Lobby ».

Datasets	**Office**			**Lobby**			**Campus**		
Méthodes	R	P	F	R	P	F	R	P	F
[26]	100	61	75,77	70	55	61,56	100	77	86,81
[32]	100	81	89,36	84	72	77,78	100	86	92,52
Notre approche	98	68	80,28	75	63	68,47	85	72	77,96

Tableau 3: Comparaison des performances avec AlexNet

- « Office » : notre approche produit une valeur de F_mesure plus de 5% que l'approche dans [26] et moins de 9% que l'approche propose dans [32]
- « Lobby » : notre approche est améliorée de 6% par rapport à [26] moins de 9% par rapport à [32]
- « Campus » : notre approche est moins que les deux autres.

2. Le modèle GoogleNet

Notre approche dans cette étape basée sur le modèle GoogleNet

Le Tableau 4 montre les résultats de résumé vidéo sur les trois jeux de données multi-vues « Office, Campus et Lobby ».

Datasets	**Office**			**Lobby**			**Campus**		
Méthodes	R	P	F	R	P	F	R	P	F

[26]	100	61	75,77	70	55	61,56	100	77	86,81
[32]	100	81	89,36	84	72	77,78	100	86	92,52
Notre approche	96	82	88.4	100	83	90,71	94	80	86,43

Tableau 4: Comparaison des performances avec GoogleNet

- « Office » : notre approche produit des résumés proche à [32] par 1% et plus de 13% que [26]
- « Lobby » : notre approche améliore de 13% que [32] et de 29% que [26]
- « Campus » : notre approche produit des résumés avec la même F_mesure que [26] et mois que l'approche dans [32]

3. Le modèle Inception v2

Notre approche dans cette étape basée sur le modèle Inception V2

Le Tableau 5 montre les résultats de résumé vidéo sur les trois jeux de données multi vues « Office, Campus et Lobby ».

Datasets	**Office**			**Lobby**			**Campus**		
Méthodes	R	P	F	R	P	F	R	P	F
[26]	100	61	75,77	70	55	61,56	100	77	86,81
[32]	100	81	89,36	84	72	77,78	100	86	92,52
Notre approche	98	92	94,90	70	86	77,17	68	90	77,46

Tableau 5: Comparaison des performances avec Inception v2

- « Office » : notre approche est améliorée pour la valeur de f_mesure par 5% par rapport à [32] et de 19% par rapport à [26] indique que la capacité de notre méthode à conserver des informations plus importantes dans le résumé par rapport à les deux approches.
- « Lobby » : notre approche produit des résumés avec la même f_mesure que [32], et de plus de 16% par rapport à [26].

En combinant les modèles CNN

Le Tableau 6 montre les résultats de résumé vidéo sur les trois jeux de données multi-vues « Office, Campus et Lobby » pour les trois modèles de CNN.

Datasets	**Office**			**Lobby**			**Campus**		
Méthodes	R	P	F	R	P	F	R	P	F
AlexNet	98	68	80,28	75	63	68,47	85	72	77,96
GoogleNet	96	82	88,4	100	83	90,71	94	80	86,43
Inception v2	98	92	94,90	70	86	77,17	68	90	77,46

Tableau 6: les 3 modèles CNN

- « Office » : le modèle Inception v2 améliore la valeur de f_mesure environ 6% par rapport à GoogleNet et 14% par rapport à AlexNet.
- « Lobby » : le modèle GoogleNet améliore la valeur de f_mesure environ 13% par rapport à Inception v2 et 22% par rapport à AlexNet.
- « Campus » : les modèles AlexNet et Inception v2 produit même valeur de f_mesure mais GoogleNet est plus de 9% que les deux.

V.5.2 Graphe

D'après le graphe illustré dans la figure 43, nous avons fait une comparaison entre le modèle AlexNet, le modèle Inception2 et le modèle GoogleNet pendant le training.

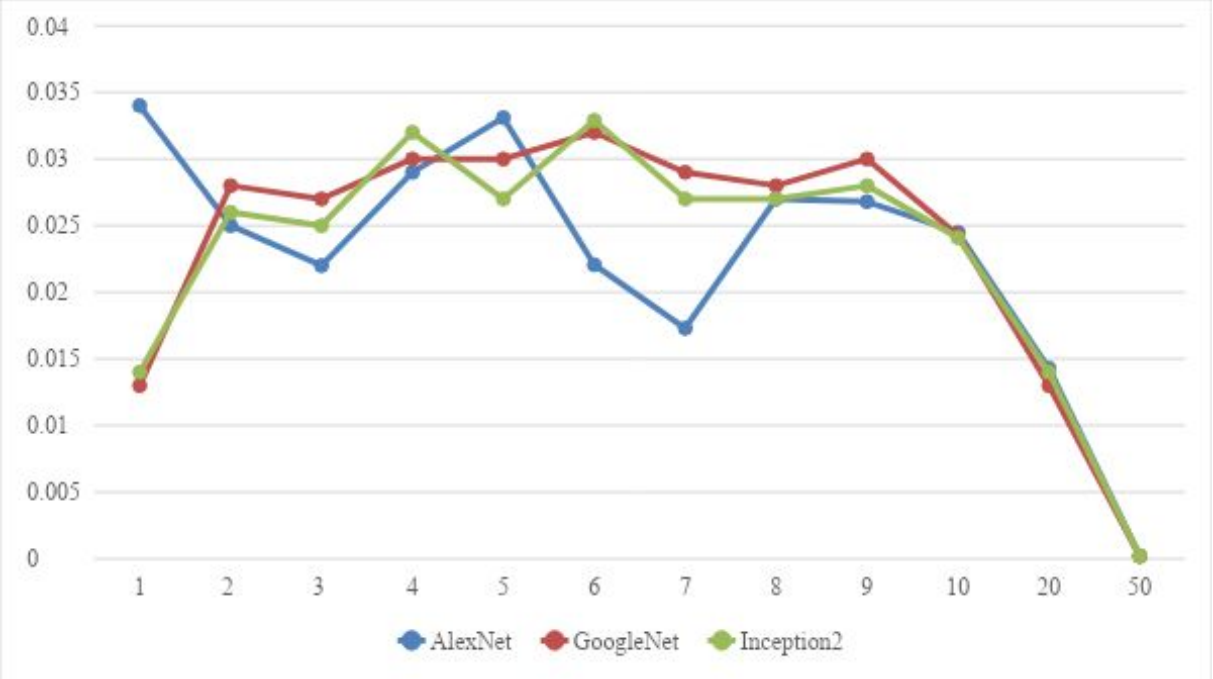


Figure 43: Graphe pour les modèles AlexNet / Inception2 / GoogleNet.

Le modèle AlexNet a des changements aléatoires de valeurs de fonction de pertes (VAL) contrairement au modèle Inception2 qui nous présente des VAL plus consistantes et nous remarquons qu'en moyenne la VAL est plus haute.

AlexNet se stabilise après un nombre d'epoch supérieur à celui d'Inception 2 à sa stabilisation signifiant que le modèle Inception 2 est meilleur que AlexNet dans le cas où le nombre d'epoch est bas.

Le modèle GoogleNet a plus ou moins des performances similaires à inception2 ce qui peut être expliqué du fait que GoogleNet est basé sur inception1 (qui est une version antérieure de inception2)

V.5.3 Comparaison d'architecture du réseau de neurones convolutifs pour l'entraînement

Avec les paramètres (200 Epoch un chunk-size de 1000 et un n-chunk de 15) et sur un entraînement basé sur les mêmes Caractéristiques Extraites nous obtenons Les résultats qui sont présentés dans la figure suivante

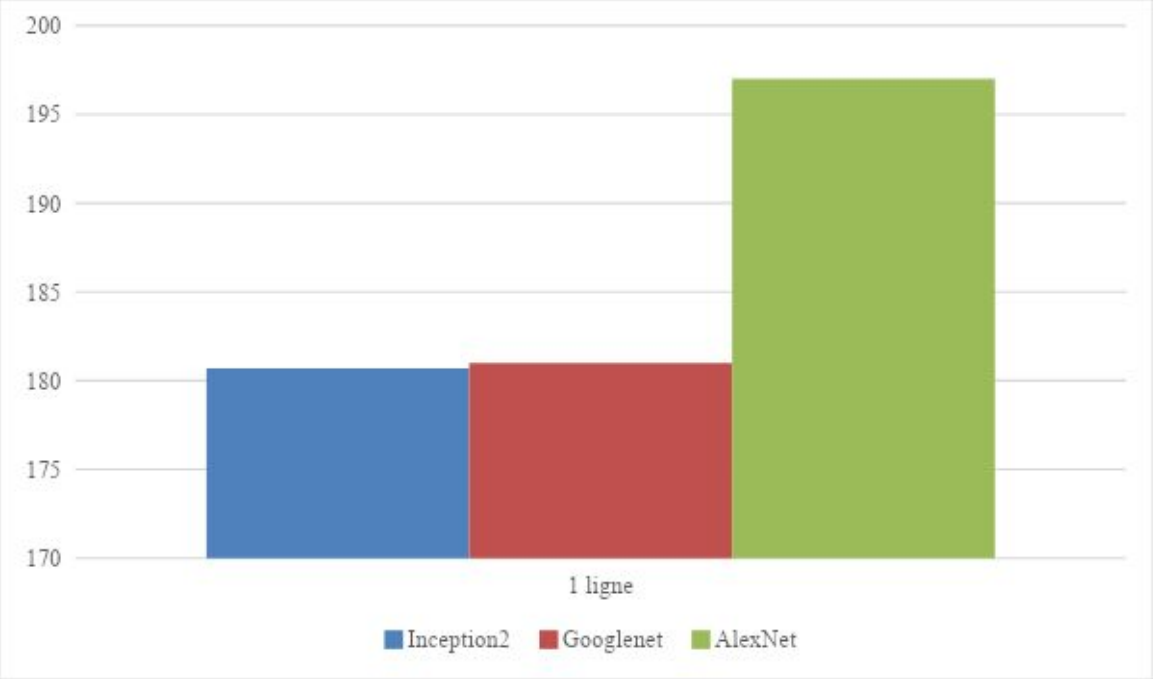


Figure 44: Comparaison le temps de réponse

D'après la figure 44, nous constatons que le modèle Inception2 est le plus rapide à l'exécution, Suivi de GoogleNet puis AlexNet (variances de dizaines de minute constatés).

Ceci peut être due au modèle AlexNet qui n'est pas habitué à des environnement aussi statiques et monochrome comme des Bureaux (Office, Campus, Lobby)

V.6 Conclusion

Dans ce chapitre nous avons présenté l'environnement matériel et logiciel sur lesquels nous avons travaillé, ainsi que les différents résultats obtenus pour le jeu de données « Office, Lobby et Campus »

Conclusion et perspectives

À l'ère moderne, des réseaux de surveillance sont installés presque partout. Ces réseaux génèrent quotidiennement des vidéos 24 heures sur 24 avec une redondance importante, ce qui entraîne un gaspillage des ressources de stockage et rend leur analyse difficile. Motivés par ces défis, nous avons proposé dans notre mémoire un outil de résumé des vidéos multi vues efficace basé sur CNN et RNN.

Notre travail consistait à prédire pour chaque image de la vidéo une probabilité pour indiquer si l'image est sélectionnée ou non dans le résumé final, pour cela nous avons d'abord segmenter chaque vidéo a des séquences puis nous avons utilisé une architecture CNN pour extraire les caractéristiques profondes d'une séquence des images. Nous introduisons ces caractéristiques dans RNN, qui est formé pour apprendre les séquences de frames informatives et non informatives et qui fournit des probabilités de sortie pour ces deux classes. Enfin, les séquences ayant les plus grandes probabilités d'informativité sont incluses dans le résumé final. Des expériences et des comparaisons approfondies avec d'autres techniques de pointe permettent de vérifier la dominance de notre système.

Perspectives : Bien qu'on ait aboutit à de bons résultats, le travail peut être amélioré :

- Nous avons utilisé le modèle CNN-RNN lourd que nous voulons remplacer par un modèle d'apprentissage profond optimisé avec une précision similaire ou supérieure.
- L'étude de l'impact de la taille de la vidéo pour appliquer un résumé vidéo efficace en appliquant une méthode d'optimisation pour travailler sur différentes tailles de vidéo, même pour les vidéos de grande taille.
- Etudier l'impact de la qualité de la vidéo résumé.

Références bibliographiques

[1]: Chua, J.L., Chang, Y.C., Lim, W.K.*: ''A simple vision-based falldetection technique for indoor video surveillance''*, SIViP9(3), 623–633, 2015

[2]: Sharma, R.A., Gandhi, V., Chari, V., Jawahar, C.V.: "*Automatic analysis of broadcast football videos using contextual prior",* SIViP *(*2016). doi:10.1007/s11760-016-0916-3

[3]: Jiang, W., Cotton, C., Loui, A.C.: "*Automatic consumer video summarization by audio and visual analysis".* Dans: International Conference of Multimedia and Expo (ICME), pp. 1–6. IEEE , 2011

[4]: B.T. Truong et S. Venkatesh, "*Video abstraction: a systematic review and classification*", (TOMM) 3 (1) pp. 1–37), 2007

[5]: Haiyan Xie, "*Key Frame Segmentation in Video Sequences-Applied to Reconstruction of 3D Scene*", these, l'Université de Kalmar, November, 2008.

[6]: Padalkar Milind Gajanan, "*Histogram Based Efficient Video Shot Detection Algorithms"*, 2010, DOI: 10.13140/RG.2.1.1590.3847.

[7]: U. Gargi, R. Kasturi, and S. Strayer, "*Performance characterization of video-shot-change detection methods*," Circuits et Systeme pour Video Technology, pp. 1 –13, février 2000.

[8] : Benteftifa Kheireddine, Bersali Mahmoud, ''*résume vidéo multi source''*, mémoire M2, Blida 2018.

[9] : Le groupe Adobe Dynamic Media. Initiation à la vidéo numérique, Juin 2000.

[10]: Mickael Guironnet. "*methods of video summarization from low level information, camera motion or visual attention"*. Theses, l'Université de Joseph-Fourier - Grenoble I, October 2006.

[11] : Bedouhene Saïda. ''*Recherche d'images par le contenu'',* université Mouloud Mammeri, TIZI-OUZOU, département automatique, option : Traitement d'Images et Reconnaissance de Formes, 2011.

[12] : Manuel Grand-brochier, ''*Descripteurs 2D et 2D+t de points d'intérêts pour des appariements robustes''*, thèse, Université de Blaise Pascal-Clermont II, 2011

[13] : Bogdan Ionescu, Didier Coquin et Patrick Lambert, ''*Reconnaissance de gestes dynamiques de la main, Universitaire''* , BP 806, 74016 Annecy Cedex , France, janvier 2003

[14]: Pscal 1129, "YOLOv3 trainde on VOC2007(person class) failed to detect small objects , anchors are changed but have no effect #972", 15 Juillet 2018.

[15]: C.Schmid, R.Mohret C.Bauckhage. "*Evaluation of Interest Point Detectors"*. International Journal de Computer Vision, 37(2) :151–172, 2000

[16] : H. P.Moravec. *''Toward automatic visual obstacle avoidance''*. Dans International Joint Conference on Artificial Intelligence, volume 2, page584, Massachusetts, Etats-Unis, aout 1977.

[17]: S.Baker, R.Szeliskiet P.Anandan.*A "Layered Approach to Stereo Reconstruction"*. pages 434–441, Santa Barbara,´Etats-Unis, juin 1998.

[18]: Liu, H. J. Zhang, and F. Qi, "A Novel Video Keyframe Ex-traction Algorithm based on Perceived Motion Energy Model",2003

[19]: C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, et E. Delp, "*Automated Video Program Summarization using Speech Tran-scripts*", IEEE Trans. on Multimedia, vol. 8,, 2006.

[20]: Y. F. Ma, L. Lu, H. J. Zhang, et M. Li, "*A User AttentionModel for Video Summarization*", ACM Multimedia ,2002.

[21]: Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. "*Spatio-temporal lstm with trust gates for 3d human actionrecognition"*. l'European Conference on Computer Vision, pages 816–833. Springer, 2016.

[22]: Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yuet-ing Zhuang. "*Hierarchical recurrent neural encoder for video representation with application to captioning"*, In Proceed-ings de IEEE Conference on Computer Vision et Pattern Recognition, pages 1029–1038, 2016.

[23]: Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Don-ahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. "*Sequence to sequence-video to text"*, Proceedings de IEEE international conference on computer vision, pages4534–45422015.

[24]: Zhang, K.; Chao, W.-L.; Sha, F.; and Grauman, K. "*Video summarization with long short-term memory"*.2016

[25]: Behrooz Mahasseni, Michael Lam, et Sinisa Todorovic. "*Unsupervised video summarization with adversarial lstm networks"*. 2017.

[26]: Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z. Zhou. "*Multi-view video summarization"*, Novembre 2010.

[27]: Yingbo Li and Bernard Merialdo. "*Multi-video summarization based on video-mmr"*., en 11eme International Workshop d'Image Analysé de Multimedia Interactive Services WIAMIS 10 pages 1–4, Avril 2010

[28]: Pandurang Matkar, Aditya Tajne, Sushil Bomane, Piyush Bansal, Prof. S. A. Saoji, 2016, "*Framework for Multi-View Video Summarization on Many core GPU"*, International Journal de l'Engineering Research et Technology (Ijert) Volume 05, Issue 01 2016

[29] : Ansuman Mahapatra, Pankaj K. Sa, Banshidhar Majhi, et Sudarshan Padhy. Mvs: "*A multi-view video synopsis framework. Signal Processing: Image Communication"*, 2016

[30] : L. Wang, X. Fang, Y. Guo, et Y. Fu. "*Multi-view metric learning for multi-view video summarization"*. pages 179–182, Septembre. 2016

[31]: K. Kumar, D. D. Shrimankar, and N. Singh. Event bagging: "*A novel event summarization approach in multiview surveillance videos"*. International Conference on Innovations de l' Electronics, Signal Processing and Communication (IESC), pages 106–111, Avril 2017

[32]: R. Panda and A. K. Roy-Chowdhury. "*Multi-view surveillance video summarization via joint embedding and sparse optimization"*. IEEE Transactions on Multimedia, 19(9):2010–2021,2017.

[33]: K. Kumar and D. D. Shrimankar. "*F-des: Fast and deep event summarization"*. IEEE Transactions on Multimedia, 20(2):323–334, 2018.

[34] : Stanislas Lauly. ''*Exploration des réseaux de neurones à base d'autoencodeur dans le cadre de la modélisation des données textuelles''*. PhD thesis, université de Sherbrooke, Québec, Canada, 2016.

[35]: Cunningham, Pádraig, Cord, Matthieu et Delany, Sarah Jane, "*Supervised learning. Machine learning techniques for multimedia"*, pages 21–49, 2008

[36] : William Thong. ''*Apprentissage de représentations pour la classification d'images biomédicales'',* mémoire de maîtrise, École polytechnique de montréal, tiré de

https ://publications.polymtl.ca/1842/. 2015

[37]: Deng, L et Yu, D. ''*Deep learning: methods and applications''*. Foundations et Trends® de Signal Processing, *7*(3–4), 197-387.2004

[38]: Jagreet Kaur Gill, "*Automatic Log Analysis using Deep Learning and AI",* 27 Aout 2020 https://www.xenonstack.com/blog/log-analytics-deep-machine-learning/ consulter le 12/09/2020.

[39]: Ivan Vasilev, Daniel Slater, Gianmario Spacagna, Peter Roelants, Valentino Zocca, "*Python Deep Learning",* 2019

[40] : Medjdoubi Abdelkader*, '' L'analyse Du Sentiment Utilisant Le Deep Learning''* thèse, Université Tahar Moulay Saida*,* juin 2019

[41] : Chabot, Florian. '' *Analyse fine 2D/3Ddevéhicule sparrése aux de neurones profonds''*. Université Clermont Auvergne, 2017.

[42]: D. E. Rumelhart, G. E. Hinton, R. J. Williams, ''*Learning representations by back-propagating errors''*. Lear-ning representations by back-propagating errors.Cognitive modeling 5(3), 1.1988

[43]: Sushen Zhang, S. M. Hosseini Bamakan, Qiang Qu, Sha Li, "*Learning for Personalised Medicine: A Comprehensive Review from Deep Learning Perspective"*, IEEE Reviews in Biomedical Engineering, Aout 2018, DOI: 10.1109/RBME.2018.2864254.

[44]: Md Zahangir Alom1, Tarek M. Taha, Chris Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Mahmudul Hasan, Brian C.Van Essen, Abdul A.S. Awwal et Vijayan K. Asari, "*A State-of-the-Art Survey on Deep Learning Theory and Architectures*", 2019.

[45]: A. Graves, "*Practical variational inference for neural networks*", l'Advances de Neural Information Processing Systems 2011

[46]: Hochreiter, S., Schmidhuber, J, ''*Long short-term memory. Neural Computation''* 1997

[46]: Saman Sarraf, "*french word recognition through a quick survey on recurrent neural networks using long-short term memory rnn-lstm"*, 2018

[47]: Sonia Barrios, David Buldain, María Paz Comech , Ian Gilbert et Iñaki Orue, "*Partial Discharge Classification Using Deep Learning Methods—Survey of Recent Progress"*, 27 June 2019

[48]: Xiaofeng Yuan, Lin Li, Yalin Wang*, "Nonlinear dynamic soft sensor modeling with supervised long short-term memory network"*, p2, 2019, DOI: 10.1109/TII.2019.2902129

[49]: Moez Baccouche. "*Neural learning of spatio-temporal features for automatic video sequence classification",* Theses, INSA de Lyon, juillet 2013

[50]: IndustryWired, "*The Era of Computer Vision Is Here"24/01/*2020, https://industrywired.com/the-era-of-computer-vision-is-here/ consulter le 02/09/2020

[51]: Sonia Barrios, David Buldain, María Paz Comech , Ian Gilbert et Iñaki Orue, "*Partial Discharge Classification Using Deep Learning Methods—Survey of Recent Progress"*, 27 June 2019

[52]: Nura Aljaafari, "*chthyoplankton Classification Tool using GenerativeAdversarial Networks and Transfer Learning*", King Abdullah Université de Science et Technologie Thuwal, Kingdom de Saudi Arabia, Février. 2018.

[53]: A. Krizhevsky, I. Sutskever, and G. E. Hinton, "*Imagenet classification with deep convolutional neural networks*", 2012

[54]: Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.;Anguelov, D.; Erhan, D.; Vanhoucke, V. et Rabinovich,A. 2015. "*Going deeper with convolutions"*.2015

[55]: Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. "*Rethinking the inception architecture for computervision"*. 2016

[56]: Bharath Raj, "*A Simple Guide to the Versions of the Inception Network"* 29 Mai 2018,

https://towardsdatascience.com/a-simple-guide-to-the-versions-of-the-inception-network-7fc52b863202 consulter le 05/08/2020

[57]:

https://www.backstreet-surveillance.com/education-advice-tips/business-camera-placement.html consulter le 05/08/2020.

[58] : Henri Michel, « *Google Colab : Le guide Ultime »*. 4 Nov 2019, https://ledatascientist.com/google-colab-le-guide-ultime/

[59] Cyril-Alexandre Artificial Intelligence - Functional programming, https://www.supinfo.com/cours/3AIT/chapitres/06-python

[60]: E. Bressert. SciPy and NumPy: "*An Overview for Developers"*. O'Reilly Media, 2012.

[61] : Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel et Mathieu Blondel. *«Scikit-learn: Machine Learning in python* ».12, Octobre 2011.

[62], Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga,

Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, et Xiaoqiang Zheng. Tensorflow : "*Large-scale machine learning on heterogeneous distributed systems"*, 2016.

[63]: Kari Pulli, Anatoly Baksheev, Kirill Kornyakov, et Victor Eruhimov. "*Realtim computer vision with opencv"*. June 2012

[64] https://pypi.org/project/scipy/

[65] : https://www.tutorialspoint.com/python3/python_tutorial.pdf

[66]: Anurag Kishore, Stuti Jindal et Sanjay Singh, *"Designing Deep Learning Neural Networks using Caffe"*, September 17, 2015.

[67]: S. H. Ou, C. H. Lee, V. S. Somayazulu, Y. K. Chen, and S. Y. Chien. "*On-line multi-view video summarization for wireless video sensor network", 2015*

[68] : Alain Baccini, Sébastien Déjean, Nongdo Désiré Kompaoré, et Josiane Mothe. ''*Analyse des critères d'évaluation des systèmes de recherche d'information. Technique et Science Informatiques''*, 2010

[69]: Alvaro Arcos-Garc´ıa, Juan A. Alvarez-Garcıa, Luis M. Soria-Morillo Dpto. "*Evaluation of Deep Neural Networks for traffic sign detection systems"* de Lenguajes y Sistemas Informaticos, Sevilla, Spain, 2018