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 intitulée : *Ingénierie du Logiciel*

Résumé vidéo multi vues

Réalisé Par :

Kobbi Fatima el zohra

Leulmi Touba yasmine

Les jurys sont composés de :

Président : Arkam

Examinatrice : Djeddar

Encadreur : KAMECHE Abdallah Hicham

Année universitaire : **2020 / 2021**

Remercîment

Avant tout, je remercie DIEU le tout puissant de m'avoir donnée la force et le courage afin que je puisse accomplir ce travail.

Je veux exprimer par ces quelques lignes de remerciement notre gratitude envers tout d'abord mon encadrant, Kameche Abdallah Hicham ses conseils, son encadrement et son assistance tout au long de cette année.

Enfin, je tiens à remercier également toutes les personnes qui ont participé de près ou de loin au bon déroulement de ce projet de fin d'étude.

Dédicace

A

Nos parents

Les plus chers au monde qui ont œuvrés pour ma réussite et qui n'ont jamais cessé de m'encourager et de me soutenir. Puisse Dieu faire en sorte que ce travail porte son fruit ; Merci pour les valeurs nobles, l'éducation et le soutient permanent. Je vous souhaite une longue vie plein de bonheur.

A

Nos sœurs et Non frères

Qui ont toujours été là pour moi, des exemples de persévérance, de courage et de générosité, pour leurs encouragements continus.

A

Nos amis

Et tous ceux que j'aime et me sont chers et que j'ai omis de citer, tout en leur souhaitant la réussite dans tout ce qu'ils entreprennent

Résumé

La vidéo surveillance est un système de caméras et de transmission d'images utilisé pour contrôler les conditions de respect et de la sécurité. Ces caméras capturent des images et des vidéos qui présentent des événements différents, dont la plupart peuvent être classés moins importants et moins significatifs que d'autres.

Afin de mettre en évidence seulement les événements pertinents, le résumé vidéo revêt une grande importance car il permet d'extraire automatiquement les scènes considérées comme essentiels pour former un résumé vidéo bref et informatif.

Les études précédentes se focalisent sur la génération d'un résumé vidéo d'une caméra unique (une seule vue). Dernièrement plus d'études ont commencé à se centraliser sur les problèmes de construction d'un résumé vidéo multi vues à cause des différentes contraintes et éléments qui s'imposent, tel que la redondance de la même scène dans différentes vues.

Dans notre 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 en utilisant un réseau de neurone convolutif suivi par l'utilisation d'un réseau de neurone récurrent lstm « long short terme memory » qui prend les fonctionnalités spatiaux-temporelles présentes dans les images de la vidéo pour la construction dynamique du résumé final.

Mots clés : Résumé vidéo, multi vues, apprentissage profond, caractéristiques profondes, réseau de neurone récurrent, lstm, réseau de neurone convolutif.

Absract

Video surveillance is a system of cameras and image transmission used to monitor compliance and security conditions. These cameras capture images and videos that show different events, most of which can be classified as less important and less significant than others.

In order to highlight only the relevant events, the video summary is of great importance as it automatically extracts the scenes considered essential to form a brief and informative video summary.

Previous studies focus on generating a single camera (single view) video summary. Lately more studies have started to focus on the problems of building a multiview video summary because of the different constraints and elements that arise, such as redundancy of the same scene in different views.

In our work, we propose a solution which consists in developing an application for the generation of multi-view summary video based on deep learning for the extraction of deep characteristic vectors using a convolutional neural network followed by the use of a “long short-term memory” lstm recurrent neural network which takes the spatial-temporal functionalities present in the images of the video for the dynamic construction of the final summary.

Key word： Video summary, multi views, deep learning, deep features, recurrent neuron network, lstm, convolutional neuron network.

نبذة مختصرة

المراقبة بالفيديو هي نظام يستعمل كاميرات تنقل الفيديوهات والصور، يستخدم لمراقبة الظروف الأمنية، تلتقط هذه الكاميرات صورا ومقاطع فيديو تعرض احداثا مختلفة، ويمكن تصنيف بعضها على انها اقل أهمية من غيرها.

من اجل تصنيف الضوء على الاحداث المهمة فقط يعد ملخص الفيديو ذا أهمية كبيرة لأنه يستخرج تلقائيا المشاهد التي تعتبر ضرورية لتكوين فيديو موجز ملخص وغني بالمعلومات.

ركزت الدراسات السابقة على انشاء ملخص فيديو بكاميرا واحدة، في الآونة الأخيرة بدأت المزيد من الدراسات في التركيز على بناء ملخص بفيديوهات من كاميرات متعددة بزوايا مختلفة.

في هذه المذكرة نقترح حلا يتمثل في تطوير تطبيق لتوليد فيديو ملخص متعدد العروض يعتمد على التعلم العميق لاستخراج نواقل مميزة عميقة باستخدام شبكة عصبية تلافيفية متبوعة باستخدام ذاكرة قصيرة المدى " LSTM " الشبكة العصبية المتكررة التي تأخذ الوظائف المكانية والزمانية الموجودة في صور الفيديو من اجل البناء الديناميكي للملخص النهائي.

كلمات مفتاحية : ملخص الفيديو ، وجهات النظر المتعددة ، التعلم العميق ، السمات العميقة ، شبكة الخلايا العصبية المتكررة ، LSTM ، شبكة الخلايا العصبية التلافيفية.

Table de matière

Chapitre I : Apprentissage automatique (machine learning)

Chapitre II : Les concepts de base de données vidéo

Chapitre III : Approche proposée

Chapitre IV : Tests et résultats

Conclusion General

Liste des figures

Chapitre I : Apprentissage automatique (machine learning)

Chapitre II : Les concepts de base de données vidéo

Chapitre III : Approche proposée

Chapitre IV : Tests et résultats

Liste des Tableaux

Chapitre III : Approche proposée

Chapitre IV : Tests et résultats

Introduction générale

Introduction général

Le monde dans nos jour souffre d'une explosion considérable de données, parmi ses données on trouve des vidéosurveillances capter par des caméras situé dans plusieurs emplacement dont l'angles, l'heure et les conditions environnementales différent, nous laissons ainsi avec des vidéos de large volume de luminosité dissemblable, d'où les éventements pertinent passe inaperçu entrainant un gaspillage non seulement de ressource de stockage mais aussi rend leur analyse difficile.

Avec la propagation des caméras de la vidéosurveillance, les techniques de la vision par ordinateur et d'intelligence artificielle jouent un rôle essentiel pour analyser les vidéos, la détection d'événements, le suivi vidéo, la reconnaissance d'objets, l'apprentissage, l'indexation, l'estimation de mouvement, autrement dit les algorithmes de vision par ordinateur reposent sur les réseaux de neurones, censés imiter le cerveau de l'être humain.

Afin d'aboutir à un résumé vidéo il serait intéressant d'éliminer les images non pertinentes et redondantes, réduire ainsi la longueur de la vidéo en concevant les scènes et évènements pertinents et combiner les informations collectées depuis plusieurs angles de vue.

L'apprentissage profond (deep learning) a connus des progrès significatifs regroupent toutes les techniques qui font qu'un ordinateur soit compètent non seulement dans l'analyse du signal visuel mais aussi sonore, permettant la reconnaissance faciale et la reconnaissance de la voix humaine et de la vision par ordinateur.

L'objectif principal de notre travail est de concevoir une solution de résumé vidéo multi vues basée sur la notion d'apprentissage profond sur les réseaux de neurones convolutifs et récurrents afin de générer un résumé de haute qualité.

Notre mémoire se subdivise donc comme suit :

- **Chapitre I :** Dans ce chapitre nous avons présenté une description de la Machine learning, l'apprentissage profond et les réseaux neurones.
- **Chapitre II :** Ce chapitre contient la définition de résumé vidéo et tous les concepts liés à la vidéo avec les travaux proposés dans la littérature pour la génération des résumés.
- **Chapitre III :** Dans ce chapitre nous allons décrire la méthode que nous avons proposée pour la génération du résumé vidéo à partir de plusieurs vidéos.

- **Chapitre IV :** Ce chapitre est consacré pour la présentation des outils utilisés pour la conception de notre application et une présentation avec discussion sur les résultats obtenus.

Chapitre I : L’apprentissage automatique (Deep Learning)

I.1 Introduction

L'apprentissage automatique (Machine Learning) englobe plusieurs sous domaines allant du plus général (apprentissage et perception) au plus spécifique, comme jouer aux échecs, démontrer des théorèmes mathématiques, écrire des poèmes, conduire une voiture ou diagnostiquer des maladies. M L se révèle être utile dans toutes les taches intellectuelles. On peut décomposer l'apprentissage automatique en quatre sous types : apprentissage supervisé, apprentissage non-supervisé, semi-supervisé, et par renforcement, nous allons détailler dans ce qui suit.

Ce chapitre est consacré pour la définition de l'apprentissage automatique et l'utilité de l'apprentissage profond pour la génération du résumé vidéo.

En va décrire très brièvement l'apprentissage automatique et leur différent type et une étude bien détaillée sur les réseaux neurones.

I.2 apprentissage automatique (machine learning)

L'apprentissage automatique (en anglais Machine Learning) est un type d'intelligence artificielle qui confère aux ordinateurs la capacité d'apprendre sans être explicitement programmés (H. P.Moravec.1977) (C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon,2006).

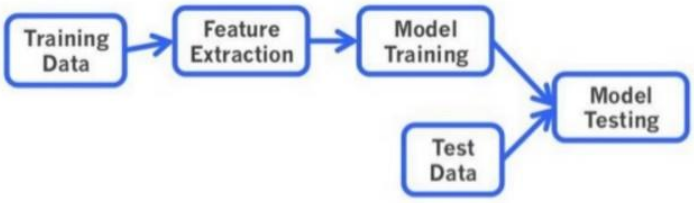


Figure 1 : le processus typique du ML

Il consiste en *la* mise en place d'algorithmes ayant pour objectif d'obtenir une Analyse prédictive à partir de données, dans un but précis. Les algorithmes de Machine Learning utilisent donc nécessairement une phase dite d'apprentissage.

Les programmes d'apprentissage automatique détectent des schémas dans les données et ajustent leur fonctionnement en conséquence, Les algorithmes d'apprentissage peuvent se catégoriser selon le mode d'apprentissage qu'ils emploient :

I.3 les types d'apprentissage automatique

Nous allons présenter les différents types majeurs de l'apprentissage automatique.

1. Apprentissage supervisé :

L'apprentissage supervisé commence généralement par un ensemble de données bien défini et une certaine compréhension de la façon dont ces données sont classifiées. L'apprentissage supervisé a pour but de déceler des modèles au sein des données et de les appliquer à un processus analytique. Ces données comportent des caractéristiques associées à des libellés qui définissent leur signification (par exemple, créer une application d'apprentissage automatique capable de faire la distinction entre plusieurs millions d'animaux, en se basant sur des images et des descriptions écrites. (Liu, H. J. Zhang, and F. Qi, 2003) (S.Baker, R.Szeliskiet P.Anandan. 1998).

2. Apprentissage non supervisé

L'apprentissage non supervisé est utilisé lorsque le problème nécessite une quantité massive de données non étiquetées. Par exemple, les applications de réseaux sociaux, telles que Twitter, Instagram et Snapchat, exploitent toutes de très grandes quantités de données non étiquetées. Pour comprendre le sens de ces données, il est nécessaire d'utiliser des algorithmes qui classifient les données en fonction des tendances ou des clusters qu'ils décèlent.

L'apprentissage non supervisé mène un processus itératif, analysant les données sans intervention humaine. Il est utilisé avec la technologie de détection de spam envoyé par e-mail. Les e-mails normaux et les spams comportent un nombre de variables beaucoup trop élevé pour qu'un analyste puisse étiqueter les e-mails indésirables envoyés en masse. En revanche, les discriminants d'apprentissage automatique, basés sur la mise en cluster et l'association, sont appliqués pour identifier les courriers électroniques non désirés (S.Baker, R.Szeliskiet P.Anandan. 1998)

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 (Y. F. Ma, L. Lu, H. J. Zhang, et M. Li, 2002)

3. Apprentissage par renforcement

L'apprentissage par renforcement est un modèle d'apprentissage comportemental. L'algorithme reçoit un feedback de l'analyse des données et guide l'utilisateur vers le meilleur résultat. L'apprentissage par renforcement diffère des autres types d'apprentissage supervisé,

car le système n'est pas formé avec un ensemble de données exemple. Au lieu de cela, le système apprend plutôt par le biais d'une méthode d'essais et d'erreurs. Par conséquent, une séquence de décisions fructueuses aboutit au renforcement du processus, car c'est lui qui résout le plus efficacement le problème posé.

4. Apprentissage en profondeur

L'apprentissage en profondeur est une méthode spécifique d'apprentissage automatique qui intègre des réseaux neuronaux en couches successives afin d'apprendre des données de manière itérative. L'apprentissage en profondeur est particulièrement utile lorsque vous tentez de détecter des tendances à partir de données non structurées.

Les réseaux neuronaux complexes d'apprentissage en profondeur sont conçus pour émuler le fonctionnement du cerveau humain, de sorte que les ordinateurs peuvent être entraînés pour faire face à des abstractions et des problèmes mal définis. La plupart des enfants de cinq ans distiguent facilement le visage de leur instituteur de celui de l'agent chargé de leur faire traverser le passage piéton. En revanche, l'ordinateur doit fournir un travail considérable pour identifier chaque visage. Les réseaux neuronaux et l'apprentissage en profondeur sont souvent utilisés dans les applications de reconnaissance d'image, de communication orale et de vision numérique.

I.4 Les différents types d'algorithmes

I.4.1 La régression linéaire

Une régression linéaire est un modèle de ML supervisé, avec x en entrée et y en sortie elle est de la forme $y = w_1x + w_0$ ou w_0 et w_1 sont des valeurs réelles à apprendre. On définit **w** comme le vecteur $[w_0, w_1]$, et définir :

$$f(x) = w_1x + w_0 \qquad (1)$$

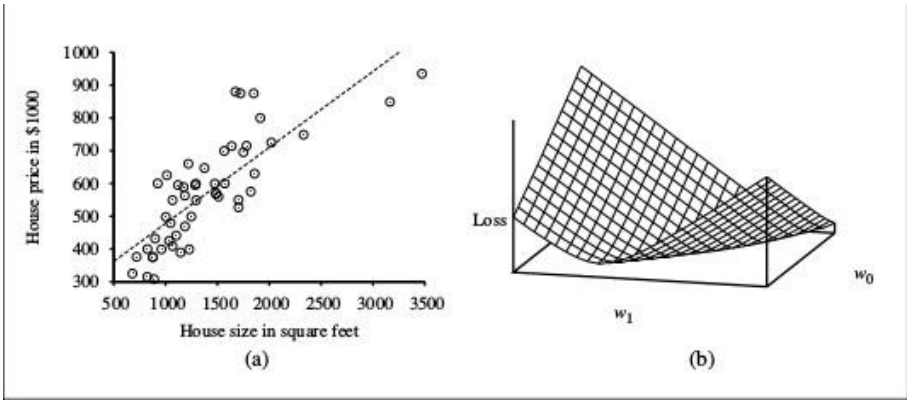


Figure 2 : modèle de régression linéaire

–

La figure 2 montre un exemple d'ensemble d'apprentissage de *n* points sur le plan *x*, *y*, chaque point représente la surface et le prix d'une maison en vente.

La tâche de trouver ***f*** qui convient le mieux à ces données s'appelle la régression linéaire. Pour trouver *f* il suffit de trouver les valeurs [w_1, w_0] qui minimise l'erreur empirique.

$$loss(f) = \sum_{i=1}^{n} (y_i - (w_1 xi + w_0))^2 \qquad (2)$$

I.4. 2 Les k plus proches voisins

L'algorithme des K-Nearest Neighbors (KNN) (K plus proches voisins) est un algorithme de classification supervisé. Chaque observation de l'ensemble d'apprentissage est représentée par un point dans un espace à *n* dimensions ou *n* est le nombre de variables prédictives. Pour prédire la classe d'une observation, on cherche les *k* points les plus proches de cet exemple. La classe de la variable cible, est celle qui est la plus représentée parmi les *k* plus proches voisins. Il existe des variantes de l'algorithme ou on pondère les *k* observations en fonction de leur distance à l'exemple dont on veut classer (C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, et E. Delp, 2006), les observations les plus éloignées de notre exemple seront considérées comme moins importantes.

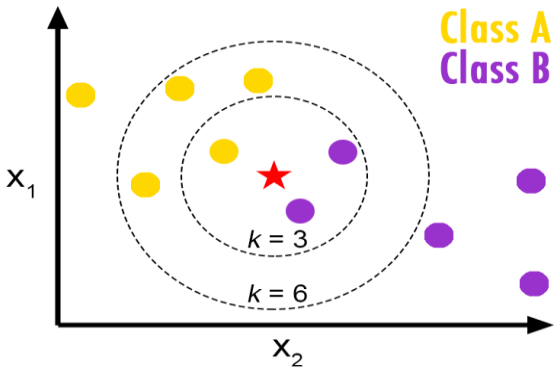


Figure 3: Exemple de classification k plus proche voisin

I.4.3 Le classifieur naïf de Bayes

Le classifieur naïf de Bayes est un algorithme supervisé probabiliste qui suppose que l'existence d'une caractéristique pour une classe, est indépendante de l'existence d'autres caractéristiques, raison pour laquelle on utilise l'adjectif «naif». Une personne peut être considérée comme un homme s'il pèse un certain poids et mesure une certaine taille. Même si ces caractéristiques sont liées dans la réalité, un classifieur bayésien naïf déterminera que la personne est un homme en considérant indépendamment ces caractéristiques de taille et de poids

malgré des hypothèses de base extrêmement simplistes, ce classifieur conduit à de très bons résultats dans beaucoup de situations réelles complexes. En 2004, un article a montré qu'il existe des raisons théoriques derrière cette efficacité inattendue (C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, et E. Delp, 2006). Toutefois, une autre étude de 2006 montre que des approches plus récentes (arbres renforcés, forêts aléatoires) permettent d'obtenir de meilleurs résultats(Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang, 2016).

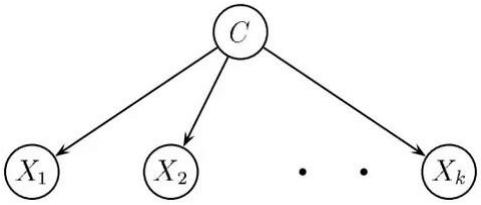


Figure 4 : Le classifieur naïf de Bayes est basé sur le théorème de Bayes avec une indépendance (ditenaïve) des variables prédictives.

I.4.4 Les forets aléatoires

Combiner un ensemble de classifieurs individuels faibles pour former un unique système de classification appelé Ensemble de classifieurs a suscité un intérêt grandissant de la communauté scientifique.L'efficacité des combinaisons de classifieurs repose principalement sur leur capacité à tirer parti des complémentarités des classifieurs individuels, dans le but d'améliorer autant que possible les performances en généralisation de l'ensemble.

Parmi les différentes approches de construction d'ensembles de classifieurs, l'algorithme des forêts aléatoires, composé d'un ensemble de classifieurs élémentaires de type arbres de décision, le but de l'algorithme est de conserver les avantages des arbres de décision tout en éliminant leurs inconvénients et particulièrement leur vulnérabilité au sur-apprentissage. C'est un algorithme qui peut être utilisé aussibien pour la classification que pour la régression.

L'algorithme repose sur trois idées principales :

- Pour M observations de l'ensemble d'apprentissage, chacune décrite par n variables prédictives, on crée B nouvel échantillon de même taille M par tirage avec remise. Cette technique s'appellele **boostrap**. Chacun des B échantillon servira à l'apprentissage d'un arbre de décision.
- Pour n caractéristiques, un nombre $k < n$ (généralement $\sqrt{n}$) est tiré aléatoirement de sorte qu'à chaque nœud de l'arbre, un sous-ensemble de k caractéristiques soit tiré

aléatoirement, parmi lesquelles la meilleure est ensuite sélectionnée pour le partitionnement.

— Pour classer une nouvelle observation, on procède par vote majoritaire. On fait passer cette observation par les B arbres et sa classe c'est la classe majoritaire parmi les B prédictions

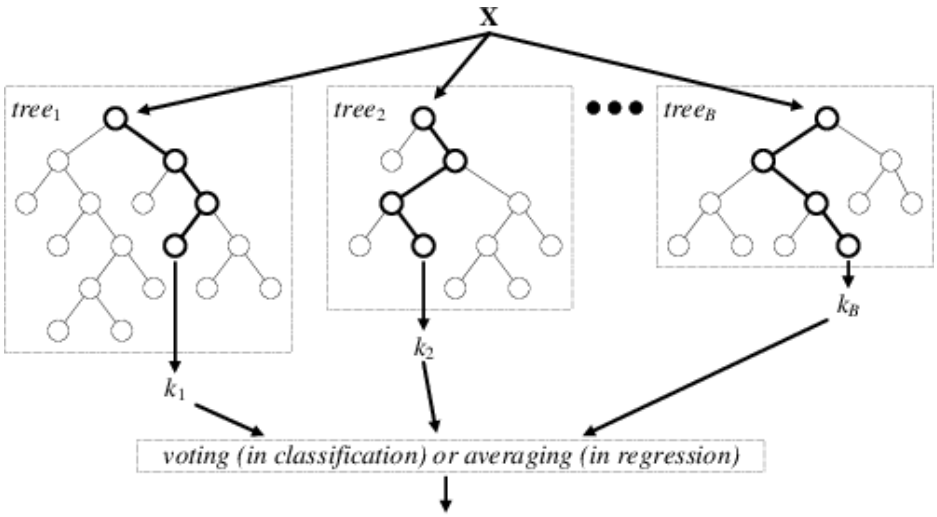


Figure 5 : Création de B boostrap à partir des exemples d'apprentissage (C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, et E. Delp, 2006)

I.4.5. Les machines à vecteur de support

Les Support Vector Machine (SVM) (machines à vecteur de support) sont des algorithmes de classification binaire non linaire très puissant.

Le principe des SVM consiste à construire une bande séparatrice non linéaire de largeur maximale qui sépare deux ensembles d'observations et à l'utiliser pour faire des prédictions. l'astuce des SVM pour y parvenir consiste à utiliser une transformation ϕ non linéaire qui envoie les points $\mathbf{x}^{(1)}$, ..., $\mathbf{x}^{(M)}$ de l'espace original à n dimensions (n est le nombre de variable prédicatives) vers des nouveaux points $\phi(\mathbf{x}^{(1)})$, ..., $\phi(\mathbf{x}^{(M)})$ dans un espace de dimension plus grand que n ou ils seront plus facile à séparer(H. P.Moravec.1977).

Les SVM sont des classificateurs qui reposent sur deux idées clés :

La première idée consiste à trouver un séparateur linéaire de largeur maximale, c'est la notion de marge maximale. La marge est la distance entre la frontière de séparation et les échantillons les plus proches. Ces derniers sont appelés vecteurs supports. Le problème est de trouver cette frontière séparatrice optimale.

Dans le cas où le problème est linéairement séparable, le choix de l'hyperplan séparateur n'est pas évident. Il existe en effet une infinité d'hyperplans séparateurs, dont les performances en phase d'apprentissage sont identiques, mais dont les performances en phase de test peuvent être très différentes. Pour résoudre ce problème, il a été montré [8], qu'il existe un unique hyperplan optimal, défini commel'hyperplan qui maximise la marge entre les échantillons et l'hyperplan séparateur.

Il existe des raisons théoriques à ce choix. Vapnik a montré (Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yuet-ing Zhuang, 2016) que la capacité des classes d'hyperplans séparateurs diminue lorsque leur marge augmente.

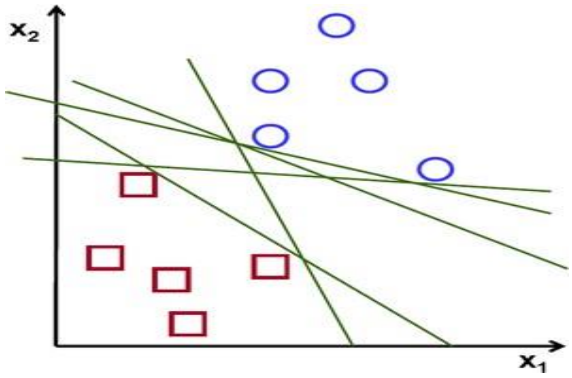


Figure 6 : On cherche un hyperplan qui divise les observations en deux catégories.

I .5. l'apprentissage profond (deep learning)

Le Deep Learning est un nouveau domaine de recherche du ML, qui a été introduit dans le but de rapprocher le ML de son objectif principal : l'intelligence artificielle. Il concerne les algorithmes inspirés par la structure et le fonctionnement du cerveau.

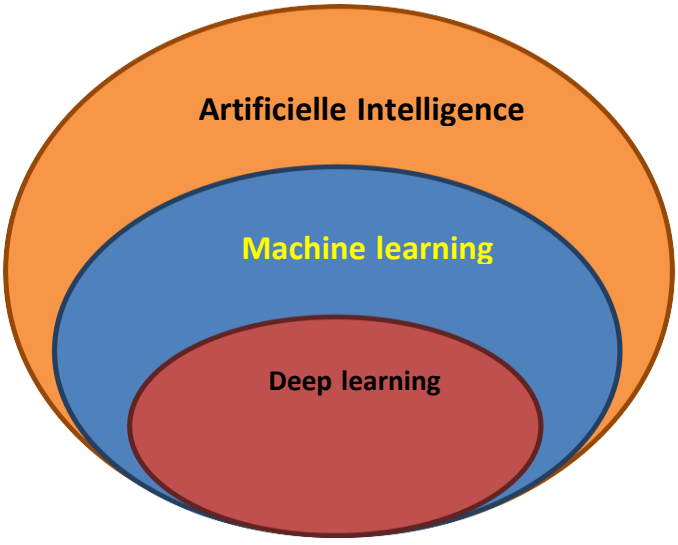


Figure 7 : La relation entre l'intelligence artificielle, le ML et le deep learning

L'apprentissage profond (« *deep learning* ») est un ensemble de techniques d'apprentissage automatique qui a permis des avancées importantes en intelligence artificielle dans les dernières années. Dans l'apprentissage automatique, un programme analyse un ensemble de données afin de tirer des règles qui permettront de tirer des conclusions sur de nouvelles données. L'apprentissage profond est basé sur ce qui a été appelé, par analogie, des « réseaux de neurones artificiels », composés de milliers d'unités (les « neurones ») qui effectuent chacune de petites opérations simples. Les résultats d'une première couche de « neurones » servent d'entrée aux calculs d'une deuxième couche et ainsi de suite. Par exemple, pour la reconnaissance visuelle, des premières couches d'unités identifient des lignes, des courbes, des angles… des couches supérieures identifient des formes, des combinaisons de formes, des objets, des contextes…, Les progrès de l'apprentissage profond ont été possibles notamment grâce à l'augmentation de la puissance des ordinateurs et au développement de grandes bases de données (« *big data* »).

I.5.1. Pour quoi le deep learning ?

Les algorithmes de ML décrits dans la première partie fonctionnent bien pour une grande variété de problèmes. Cependant ils ont échoues à résoudre quelques problèmes majeurs de l'IA telle que la reconnaissance vocale et la reconnaissance d'objets.

Le développement du deep learning fut motivé en partie par l'échec des algorithmes traditionnels dans telle tâche de l'IA. Mais ce n'est qu'après que de plus grandes quantités de données ne soit disponibles grâce notamment au Big Data et aux objets connectés et que les machines de calcul soient devenues plus puissantes qu'on a pu comprendre le potentiel réel du Deep Learning.

Une des grandes différences entre le Deep Learning et les algorithmes de ML traditionnelles c'est qu'il s'adapte bien, plus la quantité de données fournie est grande plus les performances d'un algorithme de Deep Learning sont meilleurs. Contrairement à plusieurs algorithmes de ML classiques qui possèdent une borne supérieure a la quantité de données qu' ils peuvent recevoir des fois appelée "plateau de performance", les modèles de Deep Learning n'ont pas de telles limitations (théoriquement) et ils sont même allés jusqu'à dépasser la performance humaine dans des domaines comme l'image processing.

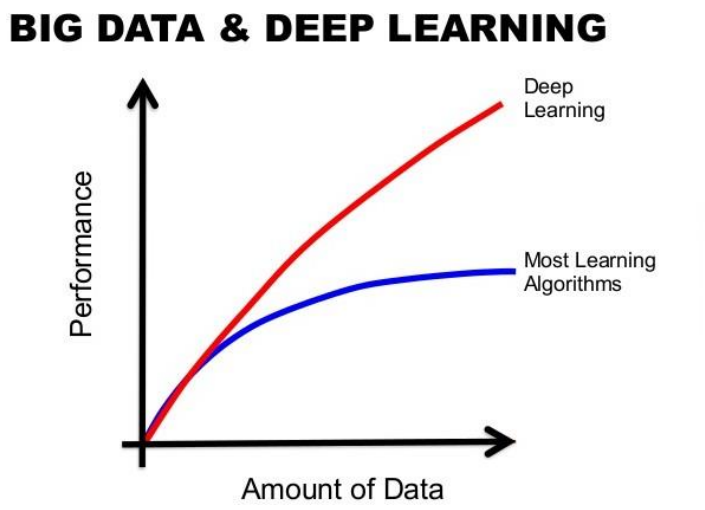


Figure 8: La différence de performance entre le Deep Learning et la plupart des algorithmes de ML en fonction de la quantité de données

Autre différence entre les algorithmes de ML traditionnelles et les algorithmes de Deep Learning 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 Deep Learning cette étape est exécutée automatiquement par l'algorithme.

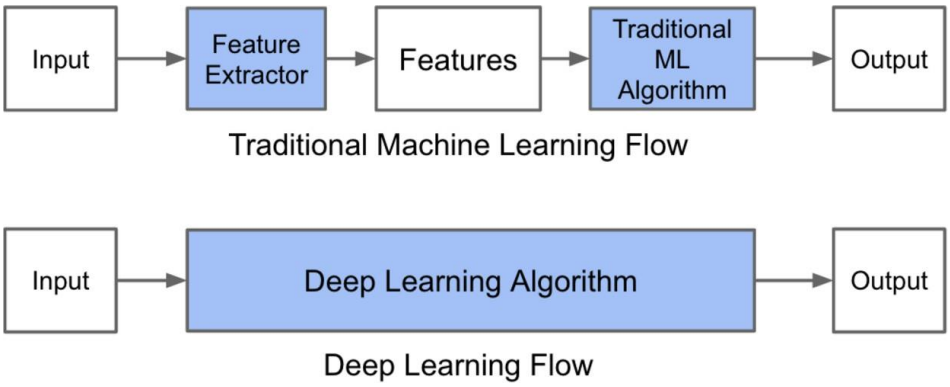


Figure 9 : Le procède du ML classique comparé à celui du Deep Learning

I.5.2 les déférents types de modèle d'apprentissage profond

Il existe un grand nombre de variantes d'architectures profondes. La plupart d'entre elles sont dérivées de certaines architectures parentales originales. Il n'est pas toujours possible de comparer les performances de toutes les architectures, car elles ne sont pas toutes évaluées sur les mêmes ensembles de données.

1. Le réseau de neurone artificiel

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 (Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Don-ahue, Raymond Mooney, Trevor Darrell, and Kate Saenko.2015)

La figure suivante montre une représentation d'un neurone réel et d'un neurone artificiel.

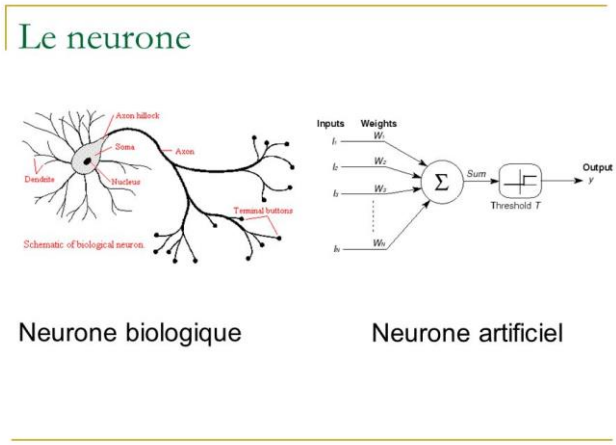


Figure 10: le neurone biologique et artificielle

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.

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 :

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

La figure suivante montre la représentation graphique de la fonction Sigmoïde

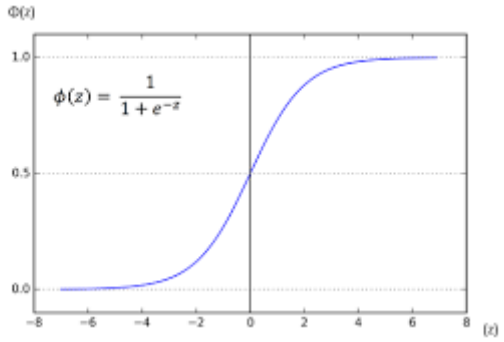


Figure 11 : la représentation graphique de fonction d'activation **Sigmoïde**

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 suivante montre représentation graphique de la fonction Tanh.

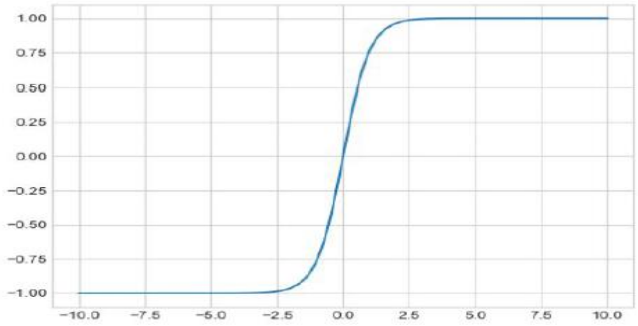


Figure 12 : la représentation graphique de fonction **Tanh**

3. La fonction ReLu

La fonction ReLu est une transformation 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), la représentation graphique de la fonction ReLu est montrée dans la figure suivante.

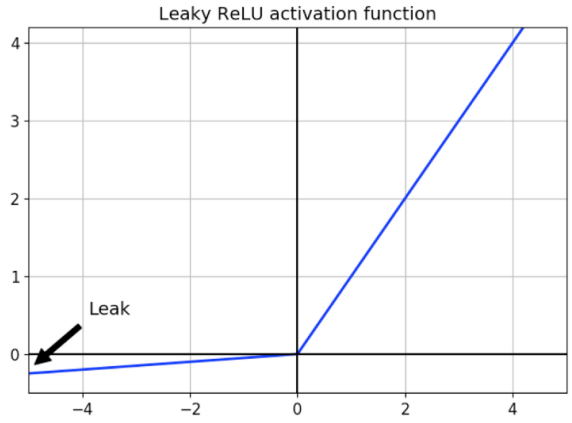


Figure 13 : représentation graphique de fonction **ReLu**

4. la fonction softmax

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 suivante :

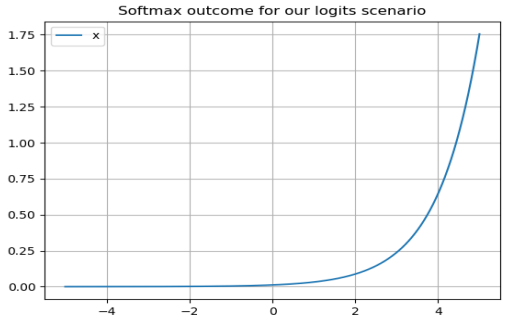


Figure 14 : représentation graphique de Fonction Softmax

b. Rétropropagation du gradient

La rétropropagation du gradient (*backpropagation*) est la méthode la plus utilisée pour L'adaptation des poids, 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.

Les domaines d'application des réseaux neuronaux sont souvent caractérisés par une relation entrée-sortie de la donnée d'information :

- La reconnaissance d'image
- Les classifications de textes ou d'images
- Identification d'objets
- Prédiction de données
- Filtrage d'un set de données

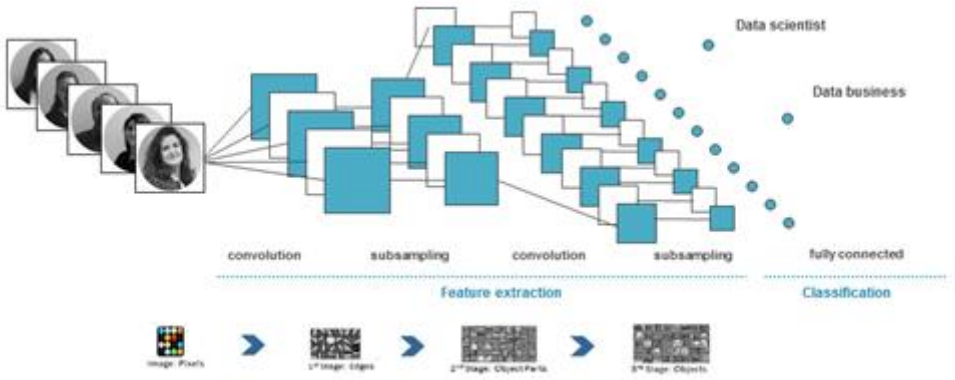


Image 15 : Réseaux neuronaux classiques pour reconnaissance d'image.

2. L'architecture d'un réseau neurones

Un réseau de neurones peut prendre des formes différentes selon l'objet de la donnée qu'il traite et selon sa complexité et la méthode de traitement de la donnée. Les architectures ont leurs forces et faiblesses et peuvent être combinées pour optimiser les résultats. Le choix de l'architecture s'avère ainsi crucial et il est déterminé principalement par l'objectif

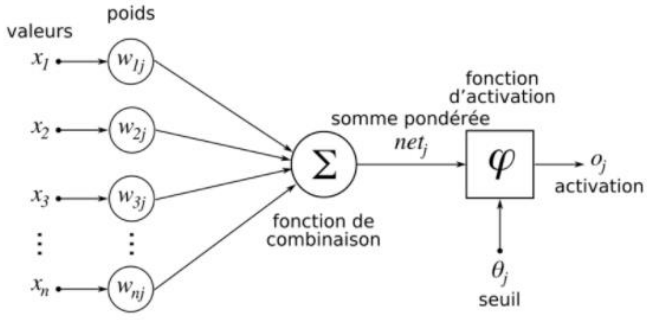


Figure 16 : architecture des réseaux neurones

Les architectures de réseaux neuronaux peuvent être divisées en 4 grandes familles :

- Réseaux de neurones Feed fowarded
- Réseaux de neurones récurrent (RNN)
- Réseaux de neurones à résonance
- Réseaux de neurones auto-organisés

3. Les types de réseaux neurones

Citant par la suite les différents types de réseaux de neurones.

a. Les réseaux de neurones feed-forwarded

En effet, feed-fowarded (propagation avant) signifie tout simplement que la donnée traverse le réseau d'entrée à la sortie sans retour en arrière de l'information.

Typiquement, dans la famille des réseaux à propagation avant, on distingue les réseaux monocouches (perceptron simple) et les réseaux multicouches (perceptron multicouche) Le perceptron simple est dit simple parce qu'il ne dispose que de deux couches ; la couche en entrée et la couche en sortie.

Le réseau est déclenché par la réception d'une information en entrée. Le traitement de la donnée dans ce réseau se fait entre la couche d'entrée et la couche de sortie qui sont toutes reliées entre elles. Le réseau intégral ne dispose ainsi que d'une matrice de poids. Le fait de disposer d'une seule matrice de poids limite le perceptron simple à un classificateur linéaire permettant de diviser l'ensemble d'informations obtenues en deux catégories distingues.

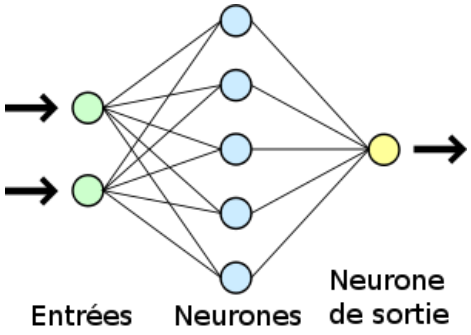


Figure 17: Les réseaux feed-forwarded

Le perceptron multicouche se structure de la même façon. L'information entre par une couche d'entrée et sort par une couche de sortie. À la différence du perceptron simple, le perceptron multicouche dispose entre la couche en entrée et la couche en sortie une ou plusieurs couches dites « cachées ». Le nombre de couches correspond aux nombres de matrices de poids dont disposent le réseau. Un perceptron multicouche est donc mieux adapté pour traiter les types de fonctions non-linéaires.

b. Les réseaux de neurones récurrents (RNN)

Les Réseaux de Neurones récurrents traitent l'information en cycle. Ces cycles permettent au réseau de traiter l'information plusieurs fois en la renvoyant à chaque fois au sein du réseau.

La force des Réseaux de neurones récurrents réside dans leur capacité de prendre en compte des informations contextuelles suite à la récurrence du traitement de la même information. Cette dynamique auto-entretient le réseau.

Les Réseaux de neurones récurrents se composent d'une ou plusieurs couches. Le modèle de *Hopfield* (réseau temporel) est le réseau de neurones récurrent d'une seule couche le plus connu.

Les Réseaux de neurones récurrents à couches multiples revendiquent quant à eux la particularité de posséder des couples (entrée/sortie) comme les perceptrons entre lesquels la donnée véhicule à la fois en propagation en avant et en rétro propagation.

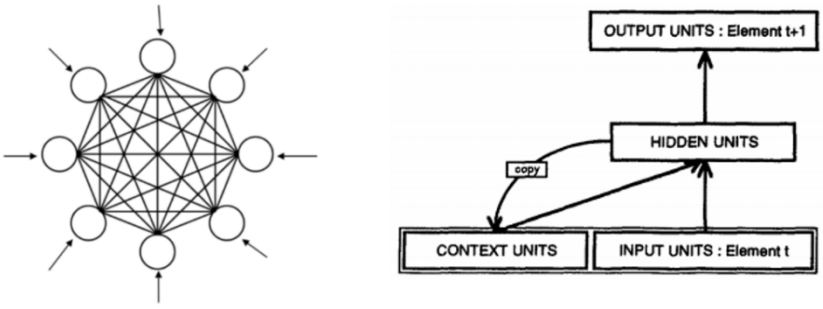


Figure 18 : les réseaux neurone RNN

Réseau de neurones récurrent dont les neurones comportent plusieurs mécanismes internes (une cellule mémoire, une porte d'entrée, une porte de sortie et une porte d'oubli)

permettant de tenir compte à la fois des dépendances (courtes et longues) dans les séquences de données.

c. Réseau de neurone récurrent à mémoire court et long terme LSTM

Les réseaux de mémoire à long court terme généralement simplement appelés « LSTM » sont un type spécial de RNN, capables d'apprendre des dépendances à long terme. Ils ont été introduits par Hochreiter et Schmidhuber (1997) et ont été affinés et popularisés par de nombreuses personnes dans les travaux suivants.

Ils fonctionnent extrêmement bien sur une grande variété de problèmes et sont maintenant largement utilisés.

Les LSTM sont explicitement conçus pour éviter le problème de dépendance à long terme. Se souvenir des informations pendant de longues périodes est pratiquement leur comportement par défaut, Tous les réseaux de neurones récurrents ont la forme d'une chaîne de modules répétitifs de réseau de neurones. Dans les RNN standard, ce module répétitif aura une structure très simple, telle qu'une seule couche tanh.

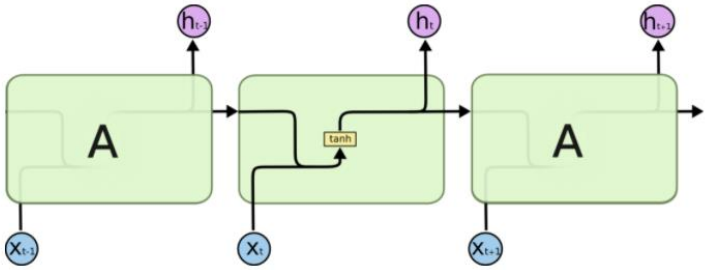


Figure 19 : Le module répétitif dans un RNN standard contient une seule couche.

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 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 ci-dessus 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 (Yingbo Li and Bernard Merialdo. 2010).

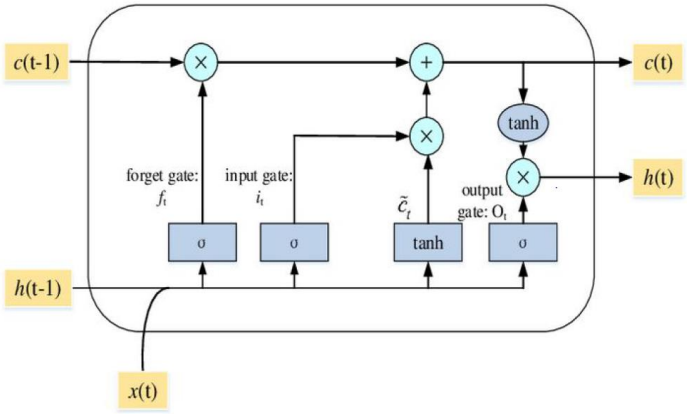


Figure 20 : Illustration d'un bloc de mémoire LSTM avec une cellule

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.

d. Gated Recurrent Unit (GRU)

L'unité récurrente fermée GRU (Gated Recurrent Unit) a été introduite en 2014 par Cho et Al [8] pour résoudre le problème de disparition du gradient rencontré par les réseaux récurrents classiques mais aussi pour proposer une architecture avec moins de paramètres à entrainer par rapport à une LSTM. À l'instar de LSTM, l'unité GRU est l'élément de base d'une architecture GRU. Une passe avant de l'unité GRU est modélisé par les équations (1) :

$$z_t = \sigma(W_z.x_t + U_Z.h_{t-1} + b_Z) \qquad \textbf{(1)}$$

$$\begin{aligned} r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) \\ \tilde{h}_t &= \tanh(W_h x_t + U_R h(r_t * h_{t61}) + b_h) \qquad \textbf{(2)} \\ h_t &= (1 - z_t) * h_{t-\&} + z_t * \tilde{h}_t \end{aligned}$$

Où σ est la fonction sigmoïde, z_t est le vecteur d'activation de la porte de mise à jour, r_t le vecteur d'activation de la porte de réinitialisation, h et le vecteur candidat et h_t est le vecteur output de l'unité GRU. W et U sont des poids, b est le vecteur biais (poids et biais sont à entrainer durant le processus d'apprentissage) et le symbole * pour le produit de Hadamard.

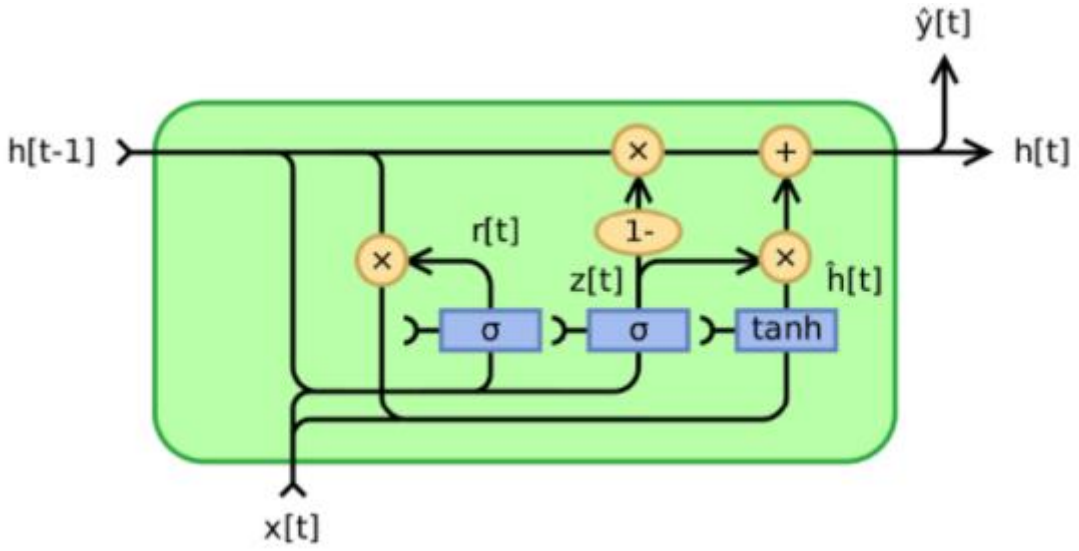


Figure 21: Unité de base GRU.

e. Les réseaux de neurones à résonance

L'appellation du réseau neuronal fait encore une fois référence à son fonctionnement. En effet, au sein des réseaux de neurones à résonance, l'activation de tous les neurones est renvoyée à tous les autres neurones au sein du système. Ce renvoi provoque des oscillations, d'où la raison du terme résonance.

Il va sans dire que ces réseaux de neurones peuvent prendre différentes formes avec des degrés de complexité plutôt élevés. Pour aller plus loin, je vous invite à vous intéresser à la Mémoire Associative Bidirectionnel qui permet d'associer deux informations de natures différentes ou encore le modèle ART (Adaptative Resonance Theory) qui fait interagir une information contextuelle avec la connaissance que l'on a déjà pour identifier ou reconnaître des objets.

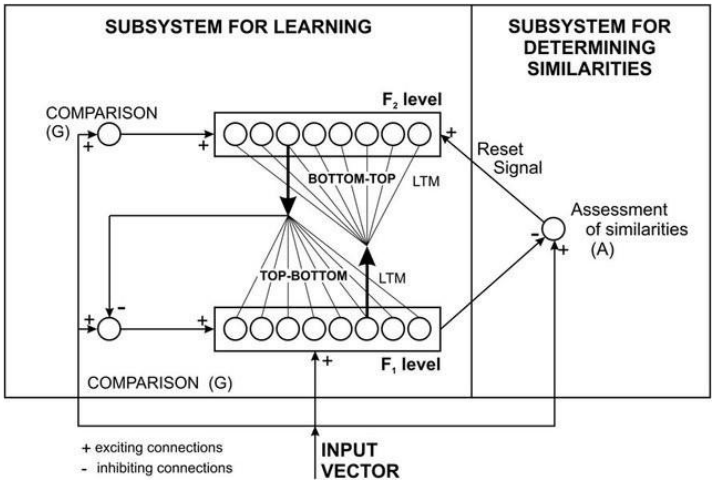


Image 22 : Les réseaux de neurones à résonance

f. Les réseaux de neurones auto-organisés

Les Réseaux de neurones auto-organisés sont surtout adaptés pour le traitement d'informations spatiales. Par des méthodes d'apprentissage non-supervisé, les réseaux neuronaux auto-organisés sont capables d'étudier la répartition de données dans des grands espaces comme par exemple pour des problématiques de clusterisation ou de classifications.

Le modèle le plus connu de ce type de réseaux de neurones est sans doute la carte auto-organisatrice de Kohonen :

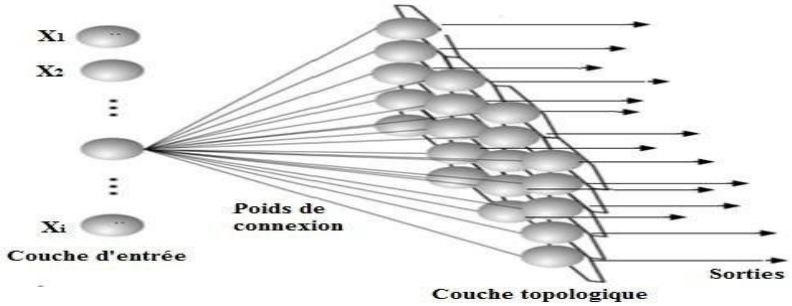


Image 23 : les réseaux de neurones auto-organisés

g. Les réseaux de neurones convolutifs (CNN) :

Réseaux de neurones (convolutifs Convolutional Neural Network (CNN))sont un type de réseau de neurones spécialisés pour le traitement de données ayant une topologie semblable à une grille. Les exemples comprennent des données de type série temporelle, qui peuvent être considérées comme une grille 1D en prenant des échantillons à des intervalles de temps réguliers et des données de type image, qui peuvent être considérées comme une grille 2D de pixels.

Les réseaux convolutifs ont connu un succès considérable dans les applications pratiques. Le nom « réseau de neurones convolutif » indique que le réseau emploie une opération mathématique appelée convolution. La convolution est une opération linéaire spéciale. Les réseaux convolutifs sont simplement des réseaux de neurones qui utilisent la convolution à la place de la multiplication matricielle dans au moins une de leurs couches. Ils ont de larges applications dans la reconnaissance de l'image et de la vidéo, les systèmes de recommandation et le traitement du langage naturel (C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, et E. Delp, 2006)

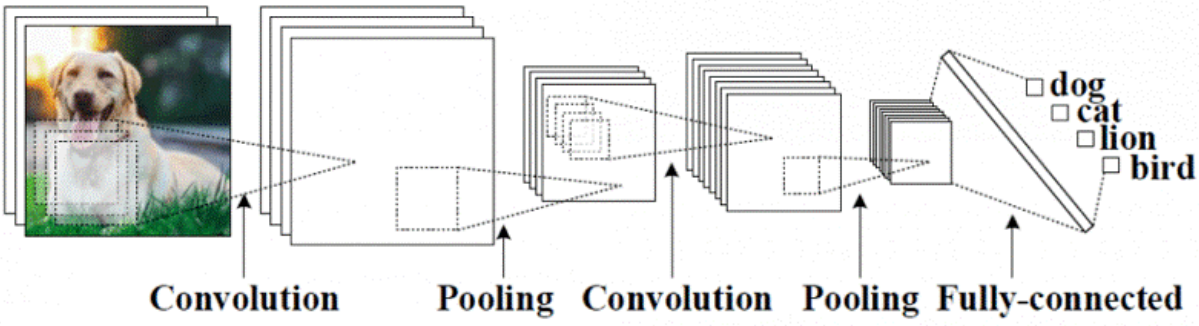


Figure 24 :Les réseaux de neurones convolutifs.

- Architectures de réseaux de neurones profonds :

L'architecture des réseaux profonds dérivés de certaine architecture originale, nous allons choisir les réseaux de neurones convolutifs (CNNs)**.**

- Principe d'architecture d'un CNN :

Les réseaux de neurones convolutifs sont à ce jour les modèles les plus performants pour classer des images**.** Désignés par l'acronyme CNN, de l'anglais Convolutional Neural Network, ils comportent deux parties bien distinctes. En entrée, une image est fournie sous la forme d'une matrice de pixels. Elle a deux dimensions pour une image aux niveaux de gris.

La couleur est représentée par une troisième dimension, de profondeur 3 pour représenter les couleurs fondamentales [Rouge, Vert, Bleu].

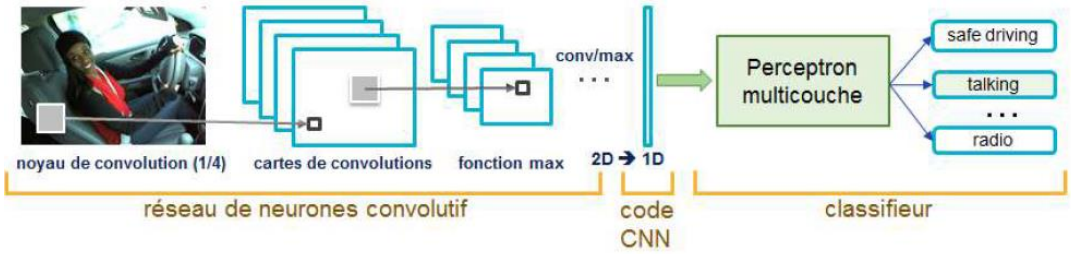


Figure 25: les principes d'un CNN

Ce code CNN en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée de couches entièrement connectées Le rôle de cette partie est de combiner les caractéristiques du code CNN pour classer l'image.

La sortie est une dernière couche comportant un neurone par catégorie. Les valeurs numériques obtenues sont généralement normalisées entre 0 et 1, de somme 1, pour produire une distribution de probabilité sur les catégories (C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, et E. Delp, 2006) (Y. F. Ma, L. Lu, H. J. Zhang, et M. Li, 2002)

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.

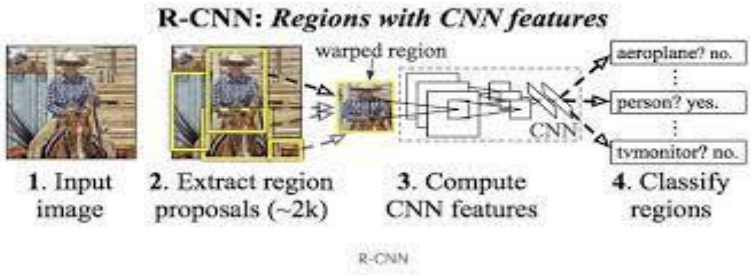


Image 26 : CNN pour l'extraction des objets

Une architecture CNN est formée par un empilement de couches de traitement comme il est illustré dans la figure suivante :

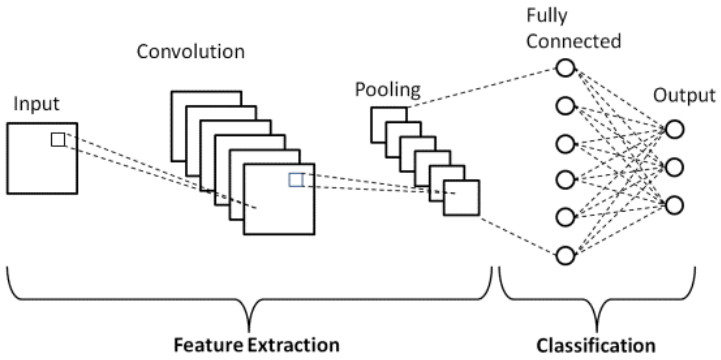


Figure 27 : architecture d'un CNN

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.

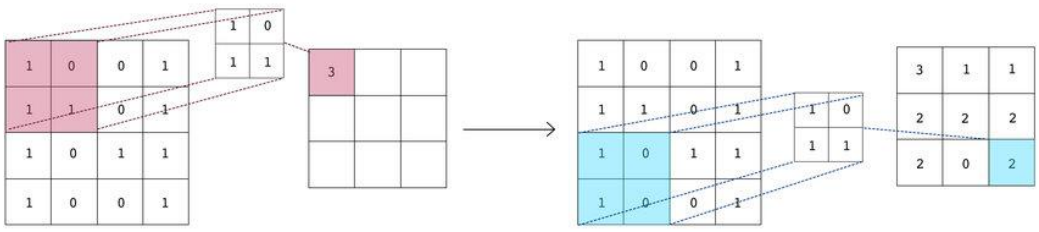


Figure 28: Exemple de convolution avec un filtre de 2x2 appliqué à une image 4x4x1 [9]

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.

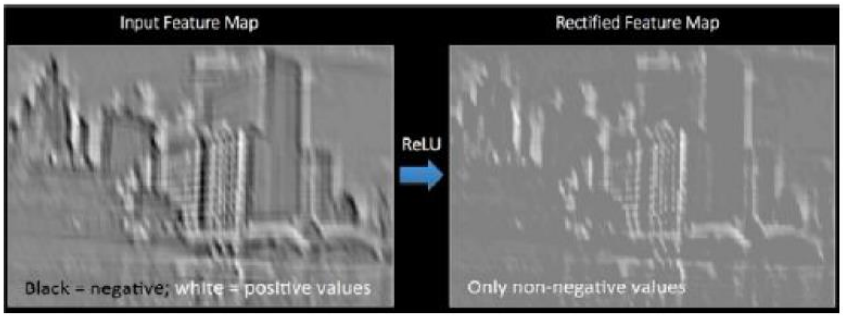


Figure 29 : Application de la fonction d'activation ReLU [8]

Couche Pooling (sous-échantillonnage)

Une autre composante importante d'un CNN., 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 suivante 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. (Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z. Zhou. 2010)

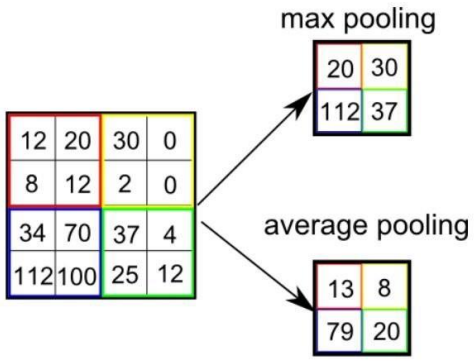


Figure 30 : Exemple de Pooling maximale et moyenne des opérations de Pooling.

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 (Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z. Zhou. 2010)

I.5.3 Les différents modèles de CNN

Nous avons focalisé sur trois 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(Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z. Zhou. 2010) 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), l'avantage de cette dernière 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).

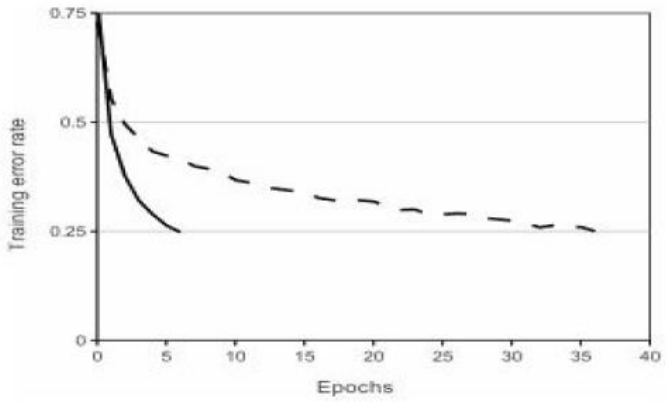


Figure 31: Taux d'apprentissage [9]

- **Plusieurs GPU.**

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 génèrent 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(Jagreet Kaur Gill , 2020)

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

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 (Subhashini Venugopalan, Marcus Rohrbach, 2015) Jeffrey Don-ahue, Raymond Mooney, Trevor Darrell, and Kate Saenko.2015)

- **L'architecture globale de AlexNet**

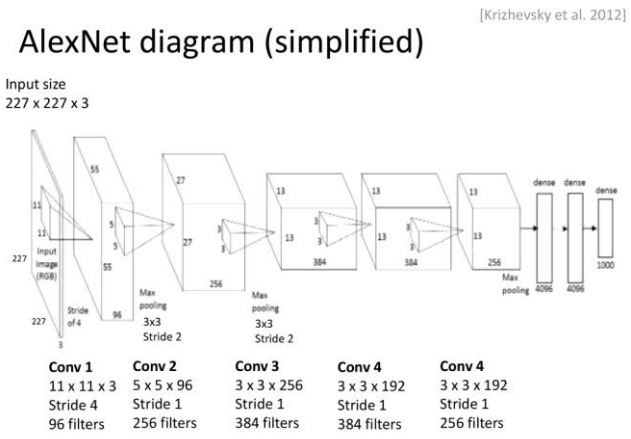


Figure 32 : Illustration de l'architecture de Alex-Net

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 précédente), 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

Google Net (ou Inception V1) a été proposé par des chercheurs de Google (avec la collaboration de diverses universités) en 2014 dans le document de recherche intitulé « Going Deeper with Convolutions ». Il a permis une diminution significative du taux d'erreur par rapport aux précédents lauréats AlexNet (gagnant de l'ILSVRC 2012), Cette architecture utilise des techniques telles que les convolutions 1×1 au milieu de l'architecture et la mise en commun globale moyenne.

Cette architecture 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 et trouve le meilleur poids lors de l'entraînement du réseau et sélectionne naturellement les caractéristiques appropriées.

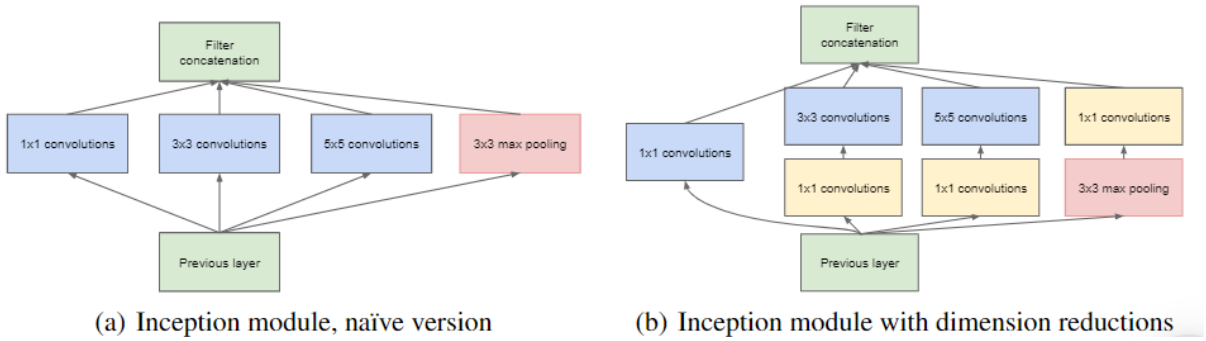


Figure 33 : Inception avec la réduction de la dimensionnalité [24]

3. Le modèle Inception version3

Inception v3 est une architecture convolutif profonde largement utilisée pour les tâches de classification, et a fait ses débuts en tant que module pour Googlenet. Il s'agit de la troisième édition du réseau de neurones convolutifs Inception de Google, initialement présenté lors du défi de reconnaissance ImageNet.

L'architecture de inception version3

L'architecture d'un réseau Inception v3 se construit progressivement, étape par étape, comme expliqué ci-dessous :

1. Convolutions factorisées : cela aide à réduire l'efficacité de calcul car cela réduit le nombre de paramètres impliqués dans un réseau. Il contrôle également l'efficacité du réseau.

2. Des circonvolutions plus petites : remplacer les circonvolutions plus grandes par des circonvolutions plus petites conduit certainement à un entraînement plus rapide. Disons qu'un filtre 5 × 5 à 25 paramètres ; deux filtres 3 × 3 remplaçants une convolution 5 × 5 n'ont que 18 paramètres (3*3 + 3*3) à la place.

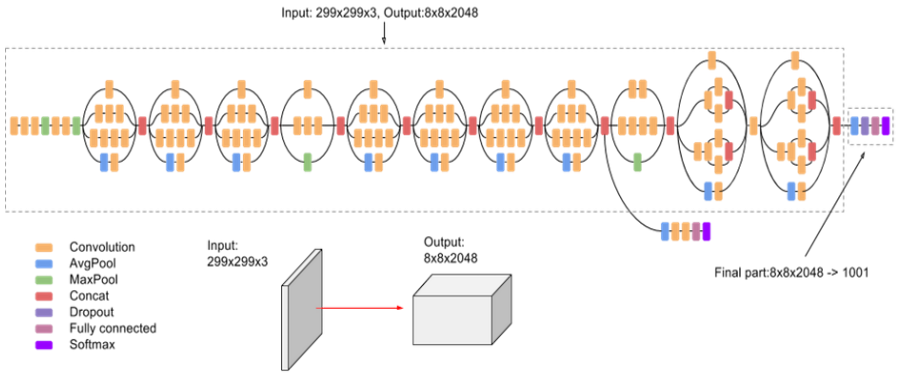


Figure 34 : Architecture globale d'Inception v3

4. Resnet

Un réseau neuronal résiduel appelé "ResNet" est un réseau de neurone artificiel avec une architecture profonde contenant un nombre de couches limités, développé par Microsoft en 2015. Il s'assemble sur des constructions obtenues à partir des cellules pyramidales du cortex cérébral. Les réseaux neuronaux résiduels accomplissent cela en utilisant des raccourcis ou des "sauts de connexion" pour se déplacer sur différentes couches.

Les réseaux de neurones ResNet emporte la compétition annuel ILSVRC avec un taux d'erreur de 3.6% ce qui est considéré comme meilleur que la précision au niveau humain. Ils font des sauts de deux ou trois couches contenant des normalisations par lots et des non-linéarités entre les deux, profitent également d'une matrice de poids supplémentaire pour apprendre les poids de saut dans certains cas. Le terme utilisé pour décrire ce phénomène est "Highwaynets". Les modèles constitués de plusieurs sauts parallèles sont appelés "Densenets". Les réseaux non résiduels peuvent également être appelés réseaux simples lorsqu'on parle de réseaux neuronaux résiduels.

L'une des principales raisons de sauter des couches est d'éviter les gradients de disparition et autres problèmes similaires. Comme le gradient est rétropropagé vers les couches précédentes, ce processus répété peut rendre le gradient extrêmement petit. Généralement en utilisant les activations des couches précédentes jusqu'à ce que la couche adjacente apprenne des poids particuliers. Pendant l'entraînement, ces poids s'ajustent aux couches en amont et agrandissent la couche sautée précédemment. Dans le cas le plus simple, ce sont les poids utilisés pour relier les couches adjacentes qui entrent en jeu.

Aussi le saut élimine les complications du réseau, le rendant plus simple, en utilisant très peu de couches pendant la phase initiale de formation. Cela accélère l'apprentissage par

dix fois, en minimisant l'effet des pentes qui disparaissent. Pourquoi ? Parce qu'il n'y a pratiquement pas de couches à traverser. Après cela, le réseau finit par remettre les couches qualifiées en place tout en apprenant l'espace des caractéristiques.

ResNet a une profondeur de deux couches utilisées dans les petits réseaux tels que ResNet18, ResNet34 ou de trois couches dans ResNet50, ResNet101, ResNet152.

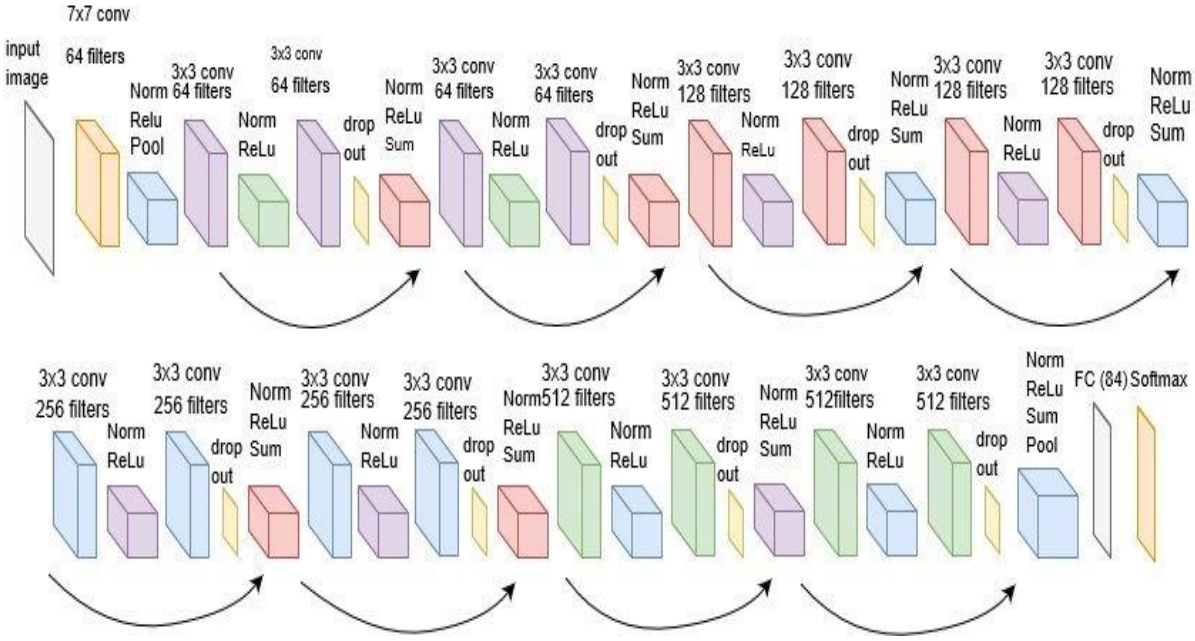


Figure 35 : Représentation du réseau ResNet

5. les auto-encodeurs

Auto-encodeur : Les auto-encodeurs sont des *algorithmes* d'*apprentissage non supervisé* à base de *réseaux de neurones* artificiels, qui permettent de construire une nouvelle représentation d'un jeu de *données*. Généralement, celle-ci est plus compacte, et présente moins de *descripteurs*, ce qui permet de *réduire la dimensionnalité* du jeu de données. L'architecture d'un auto-encodeur est constitué de deux parties : l'***encodeur*** et le ***décodeur.***

L'encodeur est constitué par un ensemble de couches de neurones, qui traitent les données afin de construire de nouvelles représentations dites "encodées". À leur tour, les couches de neurones du décodeur, reçoivent ces représentations et les traitent afin d'essayer de reconstruire les données de départ. Les différences entre les données reconstruites et les données initiales permettent de mesurer l'erreur commise par l'auto-encodeur. L'entraînement consiste à modifier les paramètres de l'auto-encodeur afin de réduire l'erreur de reconstruction mesurée sur les différents exemples du jeu de données.

La plupart du temps, on ne s'intéresse pas à la dernière couche du décodeur, qui contient uniquement la reconstruction des données initiales, mais plutôt à la nouvelle représentation créée par l'encodeur.

L'architecture la plus simple d'un auto-encodeur est semblable à une perceptron multicouches. Cependant, en fonction des données traitées, on peut utiliser différentes topologies de réseaux de neurones. Par exemple, des couches convolutives afin d'analyser des images ou des couches de neurones récurrentes pour traiter des séries temporelles ou des séquences.

À noter qu'à la différence d'un grand nombre de réseaux de neurones, les auto-encodeurs peuvent être entraînés de manière non-supervisée, ce qui permet d'appliquer ces méthodes à des jeux de données non annotés.

La figure suivante schématise un auto-encodeur simple, dont l'encodeur (encoder) traite des images (inputs), afin de les représenter comme des points dans un espace à deux dimensions (encoded représentation), puis décode cette représentation (decoder), afin de retrouver les données de départ (output) [9].

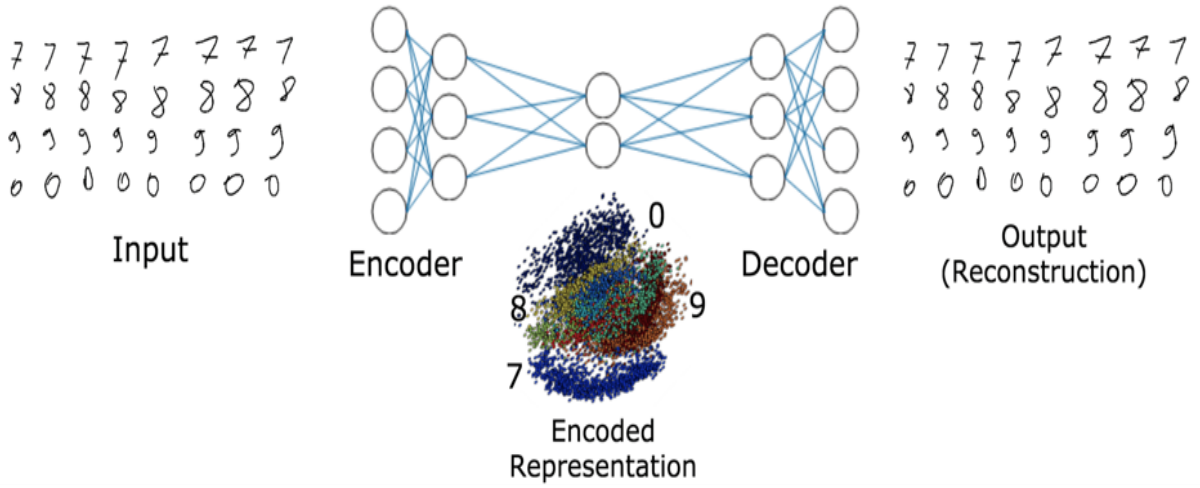


Figure 36 : présentation des encodeurs

6 . Réseau antagoniste génératif (GAN)

Les réseaux antagonistes génératifs (ou GAN, generative adversarial networks) appartiennent à une catégorie de technique d'apprentissage automatique dans laquelle deux réseaux sont placés en compétition dans un scénario de jeu à somme nulle. Généralement, les

GAN sont non supervisés et apprennent par eux-mêmes à imiter n'importe quelle répartition de données.

Les deux réseaux neuronaux qui composent un GAN sont appelés générateur et discriminateur. Le générateur est un type de réseau neuronal convolutif qui crée de nouvelles instances d'un objet. Le discriminateur est un type de réseau neuronal déconvolutif qui détermine l'authenticité de cet objet ou son appartenance à un jeu de données.

Ces deux entités sont en compétition pendant la phase d'apprentissage où les pertes se confrontent les unes aux autres afin d'améliorer les comportements, ce mécanisme étant appelé rétropropagation.

L'objectif du générateur est de produire une sortie passable sans être pris en faute tandis que celui du discriminateur est d'identifier les contrefaçons. A mesure que la double boucle de rétroaction se déroule, le générateur produit une sortie de meilleure qualité et le discriminateur identifie mieux les contrefaçons.

Les GAN sont de plus en plus connus comme une forme évoluée d'apprentissage automatique. Des chercheurs et des développeurs ont expérimenté l'utilisation de GAN pour produire des copies, même imparfaites, d'œuvres célèbres telles que la Joconde et des portraits de personnes qui n'existent pas.

Fonctionnement d'un GAN

Pour créer un GAN, on doit commencer par déterminer la sortie finale souhaitée et compiler un jeu initial de données d'apprentissage fondé sur ces paramètres. Ensuite, ces données sont envoyées de manière aléatoire dans le générateur jusqu'à ce qu'il obtienne une précision minimum dans la production des sorties.

Puis les images générées sont introduites dans le discriminateur accompagnées des points de données réelles provenant de la conception d'origine. Le discriminateur filtre les informations et renvoie une probabilité entre 0 et 1 pour représenter l'authenticité de chaque image (1 pour une image réelle et 0 pour une image contrefaite).

7. Transformer

Le Transformer est un modèle de Deep Learning (donc un réseau de neurones) de type seq2seq qui a la particularité de n'utiliser que le mécanisme d'attention et aucun réseau récurrent ou convolutionnel. Le Transformer qui est un modèle séquence à séquence dont l'architecture à la particularité de n'utiliser aucun réseau récurrent. En effet, le Transformer assure l'interdépendance des mots grâce au mécanisme d'attention. L'architecture du Transformer a inspiré l'implémentation de plusieurs modèles qui font partie, aujourd'hui, des incontournables du NLP.

Un modèle seq2seq est un modèle qui prend en entrée une séquence (une suite d'éléments du même type) et renvoie une séquence en sortie. L'exemple par excellence pour ce type de modèle est la traduction de texte.

Figure 37: exemple de modèle de transformer (traduction du texte)

Pour faire des modèles séquence à séquence avant la venue du Transfomer, il fallait faire recours au fameux LSTM (ou GRU) qu'on utilisait dans une architecture Encoder-Decoder.

Dans une architecture Encoder-Decoder, la partie « Encodeur » crée une représentation vectorielle d'une séquence de mots. Le « Décodeur », lui, retourne une séquence de mots à partir d'une représentation vectorielle.

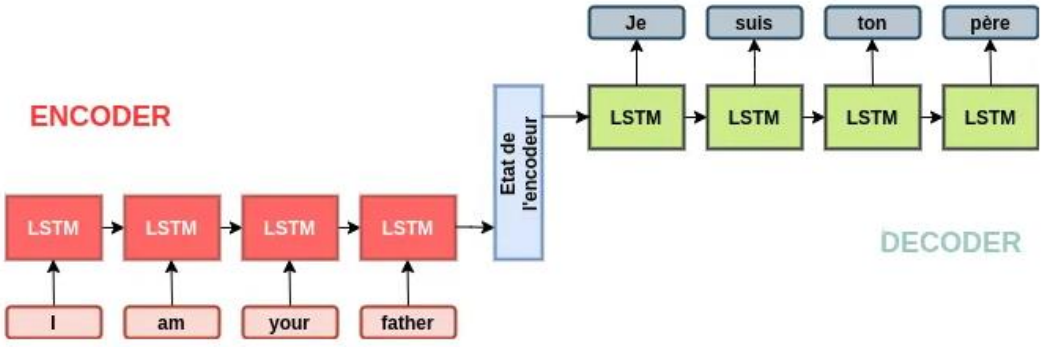


Figure 38 : le modèle transformer (K. Kumar, D. D. Shrimankar, and N. Singh, 2017)

Le rôle du LSTM est de prendre en compte l'interdépendance des mots. Mais ce modèle a une limitation : il est relativement lent à entraîner et très peu parallélisable.

L'idée du Transformer est de conserver l'interdépendance des mots d'une séquence 34en n'utilisant pas de réseau récurrent mais seulement le mécanisme d'attention qui est au centre de son architecture.

- ## Architecture du Transformer

L'architecture du Transfomer a hérité du pattern Encoder-Decoder. La partie « encodage » contient 6 encodeurs montés l'un après l'autre. La partie « décodage » consiste en 6 décodeurs également montés l'un après l'autre mais prenant chacun, comme entrée supplémentaire, la sortie du 6^{e} encodeur.

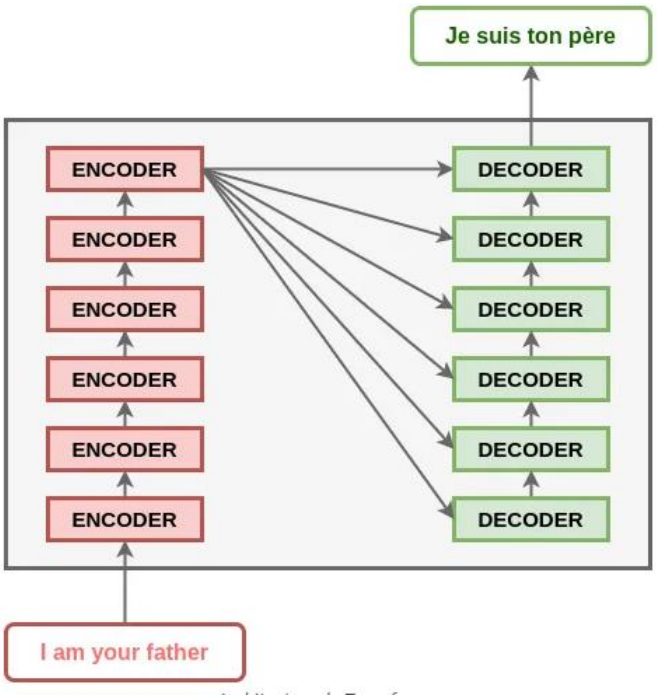


Figure 39 : Architecture du Transformer (L. Wang, X. Fang, Y. Guo, et Y. Fu. 2016)

Les encodeurs (resp. décodeurs) ont tous la même structure et leur nombre (six) est totalement arbitraire (peu importe le nombre d'encoders-decoders, le principe reste le même).

L'entrée d'un encodeur est la sortie du précédent. L'entrée du premier encodeur est vecteur d'embedding. Également l'entrée d'un décodeur est la sortie du décodeur précédent ainsi que les mots déjà encodés. Le dernier décodeur est connecté à un bloc « Réseau de

neurones linéaire + Softmax ». Le rôle de ce bloc est de permettre d'identifier à quels mots du vocabulaire correspondent les sorties du dernier encodeur.

Les blocs élémentaires du Transformer, vous l'aurez compris, sont les « encoders » et les « décodeurs ». Passons ces deux éléments à la loupe.

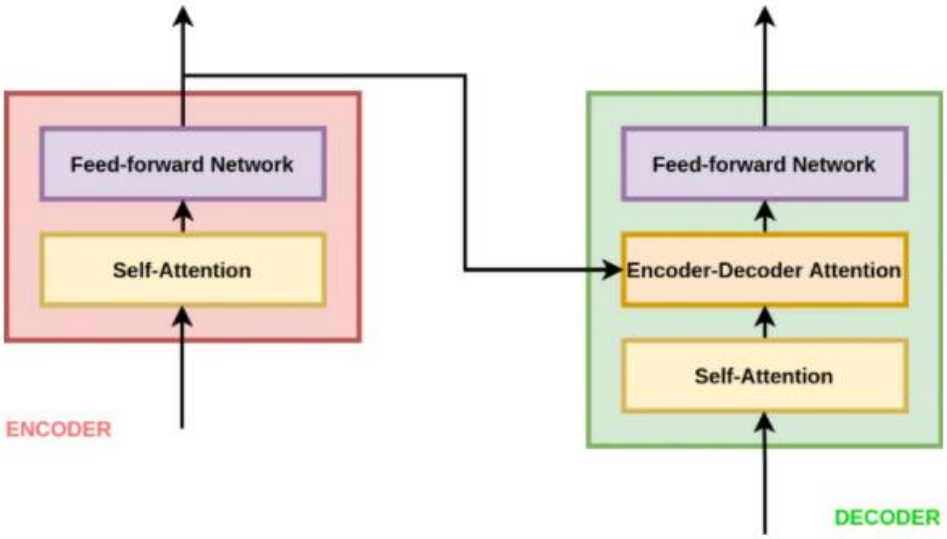


Figure 40 : Fonctionnement du transformer (L. Wang, X. Fang, Y. Guo, et Y. Fu. 2016)

L'encodeur consiste en deux blocs (qui sont tous deux réseaux de neurones) : Une couche dite de « Self-attention » et un réseau à propagation avant (ou Feed-forward Neural Network). Si le second est un réseau de neurones assez connu, le premier l'est un peu moins. La couche de Self-attention est l'élément central de l'architecture du Transformer. Son rôle est de faire garder l'interdependance des mots dans la représentation des séquences. Nous verrons le mécanisme d'attention plus en détail un plus bas. Le décodeur est également composé d'un bloc de Self-attention et d'un Feed-forward mais il contient en plus une couche « Encoder-Decoder Attention » qui a pour but de permettre au décodeur de réaliser le mécanisme d'attention entre la séquence d'entrée (encodée) et la séquence de sortie (en train d'être décodée).

I.6 Conclusion

Dans ce chapitre, nous avons cité les définitions de 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.

Chapitre II : concepts de base de données vidéo

II.1 Introduction

Nous avons assisté à une croissance dramatique de vidéo dans divers scénarios de la vie quotidienne dans nos jours, telles que les vidéos sportifs, des vidéos de grand public, des vidéo de caméra de surveillance.

Nous intéressons dans notre travail par les vidéos de surveillance, il existe plusieurs types fondamentaux de résumés vidéo.

Ce chapitre est consacré pour la description de la structure de vidéo, nous citons les principales méthodes qui constituent les deux grands types de résumés vidéo tels que la sélection d'images représentative (image clé) et résumé dynamique résultant d'une sélection de segments extraits de la vidé.

Nous présentons par la suite les deux catégories de signal vidéo analogique et vidéo numérique et quelques descripteurs des caractéristiques d'image.

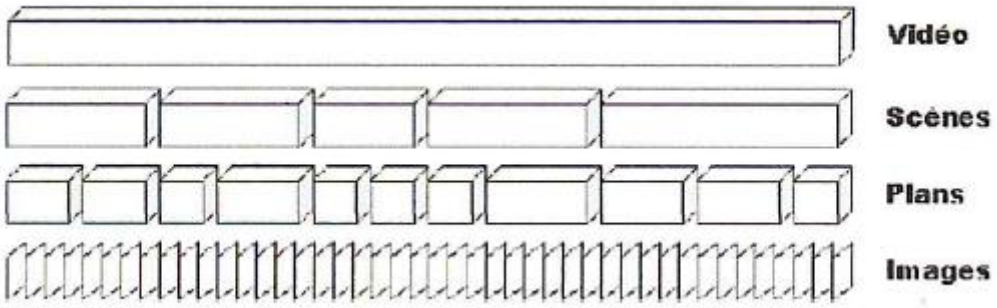
II.2 Structure de la vidéo

La vidéo est une succession d'images affichées à une fréquence de 25 images par seconde accompagnées d'une bande son, chaque image est décomposé en ligne horizontales et chaque ligne étant une succession de point (pixels) 625 lignes (567 effectives) et 720 pixels par ligne pour le PAL [1]. On caractérise la fluidité d'une vidéo par le nombre d'image par seconde (frame rate), exprimé en FPS.

La figure 41 présente la structure d'une vidéo, l'image mobile complète d'une vidéo peut être discrétisée en une séquence d'image finie (un nombre fixe d'image), chaque image nommée « frame » qui est l'unité de base de la vidéo.

Chaque image dans la vidéo à un numéro (index), tous les images de la vidéo ont la même taille et le temps entre chaque image est égal, généralement 1/25 à 1/30 images de seconde.

Les documents vidéo sont hiérarchiquement structurés en séquences, scènes plan et images (voir figure 41).



Figures 41 : Structure hiérarchique d'une vidéo

1. La scène

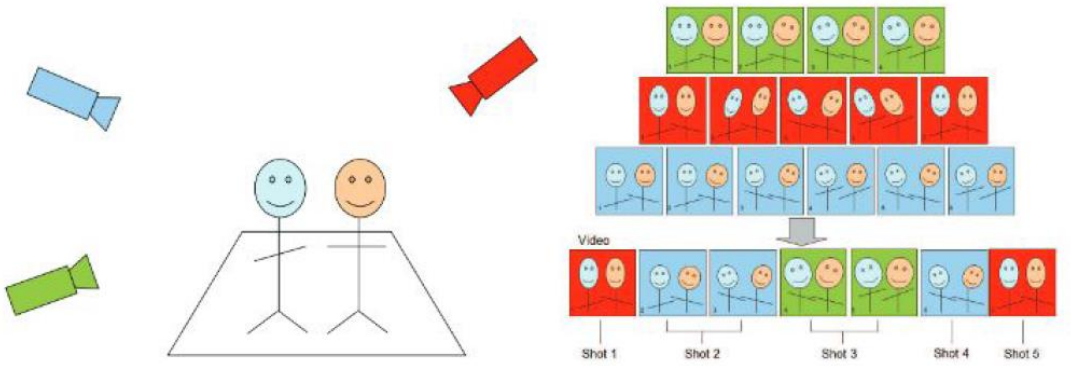
Une scène est une suite de plans qui sont cohérents d'un point de vue narratif. En d'autres termes, une scène est une collection de plans qui transmettent différentes vues d'un même événement ou d'un même objet et qui contiennent les mêmes objets d'intérêt (C.Schmid, R.Mohret C.Bauckhage, 2000) Comme illustre la figure 41.

2. Le plan (*shot*)/ Séquence

Gargi et ses camarades (Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang, 2016) 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 (Y. F. Ma, L. Lu, H. J. Zhang, et M. Li, 2002).



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

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

Figure 42 : Une scène vidéo (Y. F. Ma, L. Lu, H. J. Zhang, et M. Li, 2002)

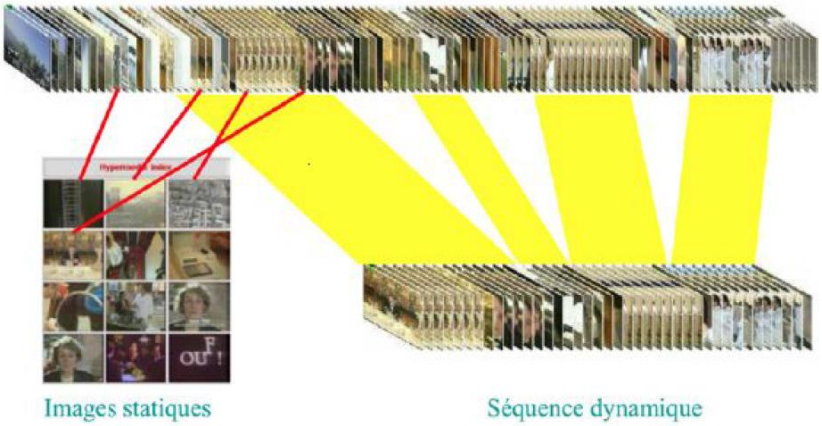


Figure 43 : les deux types de résumé vidéo (H. P.Moravec.1977)

II.3 Signal de vidéo

Le signal vidéo permet la transmission d'une séquence d'image à un diapositif d'affichage sous une forme électrique, on distingue deux grand types de signal vidéo une vidéo analogique et un autre numérique.

A. signal analogique

Le signal analogique est constitués de son qui change constamment dans une instant donnée peut prend une valeur comprise entre le minimum et le maximum autorisé (figure 44)

Les images vidéo affichages lui sont transmises sous forme de signal par 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.

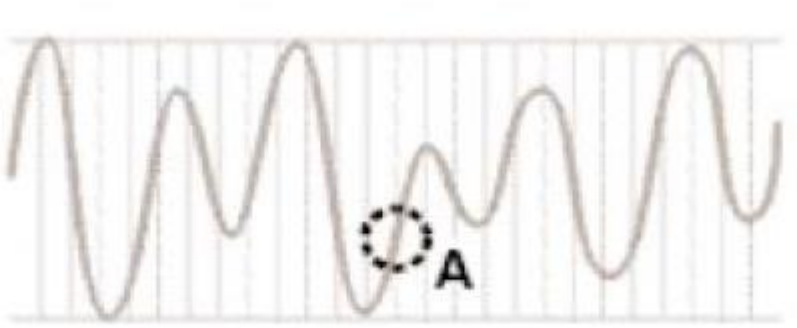


Figure 44 : Un signal analogique [2]

B. Signal numérique

Les signaux numériques, sont transmis sous forme de points sélectionnés par intervalles sur la courbe (figure 45).

Figure 45 : Illustration d'un signal numérique (H. P.Moravec.1977)

L'ordinateur utilise le 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 être interprétée à la réception comme un ensemble de nombres représentatifs de l'information émise à l'origine.

Figure 46 : Un signal numérique type binaire(H. P.Moravec.1977)

Le signal numérique 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.

II.4 Nombre d'image par seconde (frame rate)

Le système SVH permet 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'oeil de percevoir un phénomène d'animation. Pour créer ce phénomène dans la bande vidéo, un ombre d'images par seconde est exigé, en général 25 ou 30 images par seconde.

La résolution d'image (qualité de vidéo) dépond sur le contenu de chaque image et le nombre des images, elle correspond au nombre des éléments individuels constituent l'image (pixels) affiché à l'écran, elle exprimée sous la forme du nombre du pixel utilisées sur l'axe vertical (exemple : 640× 480 ou 720 × 480) donc la résolution la plus élevé permet d'obtenir une image de bonne 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 (Subhashini Venugopalan, Marcus Rohrbach, 2015) Jeffrey Don-ahue, Raymond Mooney, Trevor Darrell, and Kate Saenko.2015).

I.5 Extraction des caractéristiques

Afin de faire l'extraction de caractéristiques nous devons passer par l'étape d'extraction d'image clés.

I.5.1 Extraction d'images clés

Dans l'extraction d'images clés, une image clé est reçue et produite en étant identique à l'image originale qui est un sous-court et devrait être différent du cadre en dehors de la vidéo au même moment. L'image clé d'une vidéo est considérée comme l'une des images d'une vidéo fournissant le meilleur résumé du contenu vidéo. L'extraction de clé a joué un rôle critique dans le résumé vidéo où la plupart des techniques courantes utilisées dans la méthode d'extraction d'images clés sont basées sur des caractéristiques statistiques de macro-blocs de Streaming vidéo MPEG, basé sur l'activité Shot et enfin basé sur l'analyse de mouvement.

Le choix des caractéristiques qui expriment le contenu des images est devenu un problème très important dans la conception des résumés, ces caractéristiques sont généralement les couleurs, les textures et les formes, Il existe deux approches dans la quel permettent la résolution du problème d'extraction des caractéristiques.

La première consiste à la construction des descripteurs à la construction 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 mise en œuvre, et le membre réduit d'observation que l'on obtient, cependant que l'inconvénient majeur de ces derniers est la perte de l'information de localisation des éléments de l'image.

La deuxième 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 (Behrooz Mahasseni, Michael Lam, et Sinisa Todorovic. 2017).

I.6 Résumé vidéo

Fondamentalement, le résumé vidéo peut être défini comme une technique ou un mécanisme pour produire une vidéo plus courte que l'originale et traiter des vidéos qui contiennent redondance afin de les rendre plus intéressantes et précieuses. De plus, fournir aux utilisateurs un résumé visuel synthétique utile de la séquence de la vidéo qui peut être soit en forme d'une image en mouvement (séquences vidéo) ou d'images (images clés) (Potapov et al.,

2014). Comme mentionné précédemment, l'augmentation rapide du volume de contenu vidéo téléchargé depuis le Web outre l'exigence d'effort pour télécharger et traiter des vidéos, révèlent le besoin de nouvelles technologies efficaces et efficientes qui pourraient gérer la grande quantité de données dans le contenu vidéo.

En ce qui concerne la navigation, l'utilisateur peut acquérir suffisamment d'informations dans le minimum de temps possible puisqu'un bon résumé d'une vidéo permet à l'utilisateur de recueillir un maximum d'informations pour la séquence de la vidéo cible plus courte durée.

Des vidéos qui contiennent des résumés générés automatiquement, donne à l'utilisateur la capacité de naviguer et de parcourir un large éventail d'archives de vidéos pour lui permettre de faire des décisions efficaces lors de la sélection, de la consommation et du partage des vidéos ou de les supprimer vidéo (Rehman et Saba, 2014).

De plus, avec cette technologie l'utilisateur peut utiliser le produit final pour partager, digérer ou même profiter du contenu de la vidéo résumée qui à son tour, enregistre et améliore le stockage efficace du contenu vidéo, économise la bande passante lorsqu'il s'agit du téléchargement de la vidéo en plus de l'heure de visionnage du résumé est enregistré. Il y a de nombreux domaines touchés par le développement des techniques de résumé vidéo, tels que l'apprentissage en ligne, les vidéos personnelles, les actualités, les sports, les films, entre autres (Furini et al, 2010).

Il y a trois phases du processus d'abstraction vidéo. La première étape est la analyse des informations vidéo, suivie d'une sélection du clip qui est significatif et Enfin étant la synthèse de sortie (Zhang et al., 2015).

Pour faire l'analyse des informations vidéo, il est crucial que les fonctionnalités pertinentes,les motifs ou les structures soient identifiés dans les composants audio, textuels et visuels comme légendes. Dans la figure (47), les phases du processus d'abstraction vidéo sont présentées.

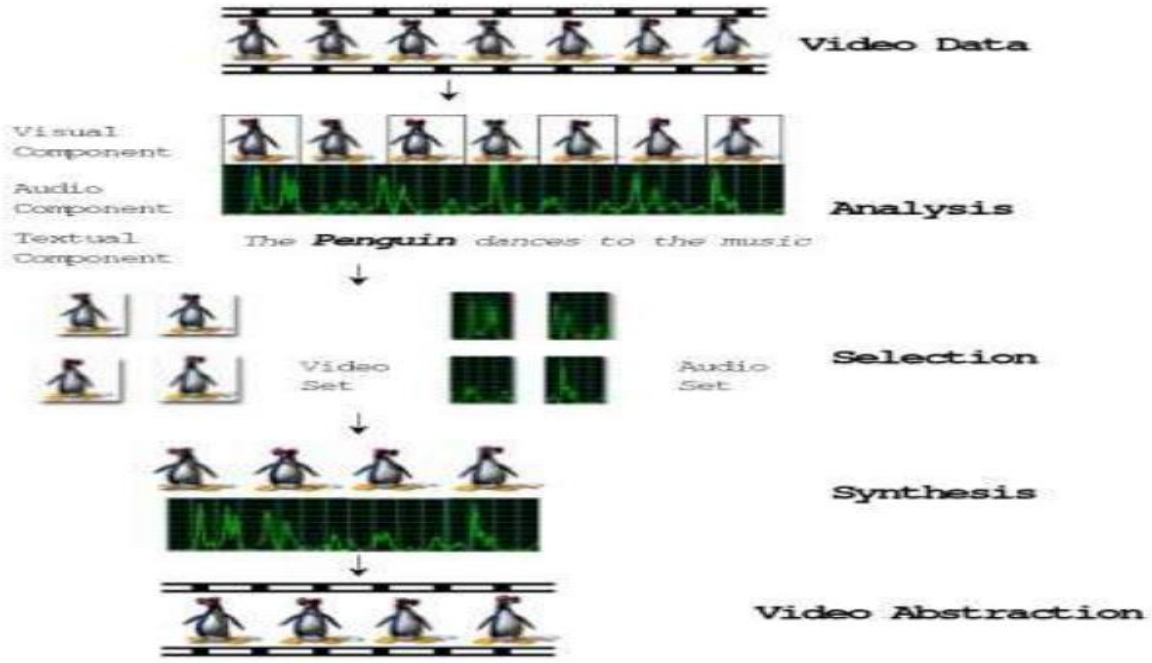


Figure 47 : Schéma d'abstraction vidéo (Zhang et al, 2015)

Il existe deux modes dans lesquels le résumé vidéo peut être représenté, ceux-ci sont le skimming vidéo dynamique et le Storyboard (résumé vidéo statique) (Saba et Altameem, 2013). Le résumé vidéo statique parle généralement du storyboard utilisé pour résumer vidéo sous forme d'images statiques ; cette méthode permet d'extraire des images clés des prises de vue vidéo originales. Après cela, les images clés sélectionnées sont mélangées ou disposées en deux. Il existe deux modes dans lesquels le résumé vidéo peut être représenté ; ceux-ci sont le skimming vidéo dynamique et le Storyboard (résumé vidéo statique) (Saba et Altameem, 2013). Le résumé vidéo statique parle généralement du storyboard utilisé pour résumer vidéo sous forme d'images statiques ; cette méthode permet d'extraire des images clés des prises de vue vidéo originales. Après cela, les images clés sélectionnées sont mélangées ou disposées en deux espaces dimensionnels. En comparaison, le résumé vidéo dynamique aide à choisir le plus important et de petites portions pertinentes et dynamiques, ceux-ci sont appelés écrémés vidéo et ont à la fois vidéo et audio et sont utilisés dans la génération du résumé de la vidéo originale (Sigari et al., 2015).

Dans la section suivante, les techniques et les types de résumé vidéo ont été discuté en profondeur.

II.6.1 Types et techniques de résumé vidéo

Dans cette section, trois principaux types de techniques du résumé vidéo seront discutées, ces sont les techniques Résumé vidéo à vue unique, de résumé statique et de clustering résumé (Lee et al, 2012), et classification de l'échantillonnages.

A. Résumé vidéo dynamique

En résumé dynamique de vidéos, les extraits des segments audio et visuels sont extraits de la vidéo originale en se basant sur la notion d'écrémage vidéo pour raccourcir des vidéos contenant des scènes et des données importantes de la vidéo originale (Knox et al., 2014).

Dans ce cas, l'utilisateur peut recevoir une vue abstraite de l'intégralité de la vidéo, qui est connu comme l'histoire vidéo. Dans cette technique d'écrémage, un segment de l'original la vidéo est prise contenant l'audio et la partie résumé vidéo de l'original.

Les types courants de résumé vidéo sous cette technique sont le modèle de mouvement et le Décomposition en valeur singulière (SVD) (Mayberry et al., 2014). La méthode sémantique l'analyse peut également être appliquée dans cette technique d'écrémage. Alors que la plupart des écrémages techniques sont basées sur l'information visuelle, d'autres méthodes ou technologies ont été considéré pour utiliser et mettre en œuvre à la fois des informations linguistiques et audio.

Dans l'abstraction vidéo d'écrémage, les schémas vidéo du film sont présentés. Les approche essaie d'obtenir le contenu d'une vidéo à partir des progrès de la sémantique humaine compréhension et histoire globale (Lu et al., 2013).

Au début, la propriété d'histogramme à deux dimensions est implémentée pour permettre segmentation de la vidéo en plans. Après cela, les règles générales des techniques courantes et un scénario spécial de production de vidéo sont appliqués/utilisés pour permettre le déroulement d'une histoire à saisir quant au degré d'avancement entre les scénarios et l'histoire globale (Lee et al, 2012). Le processus de la technique d'écrémage vidéo est illustré à la figure suivante :

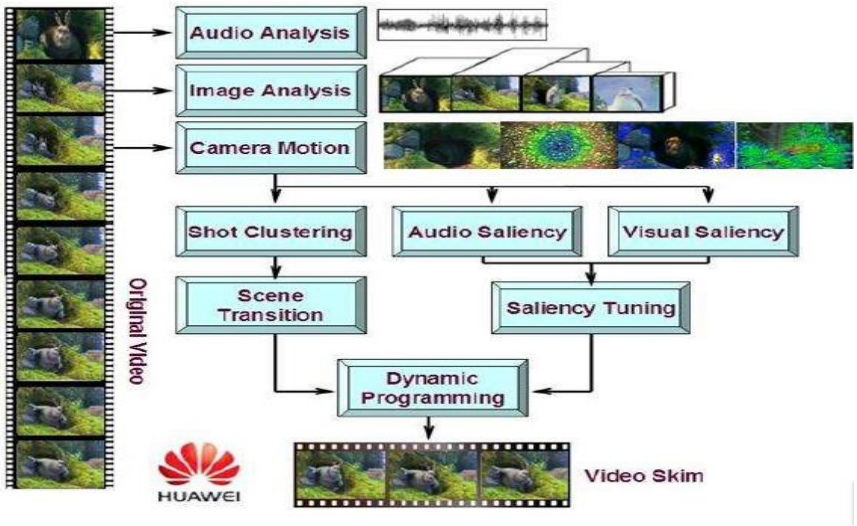


Figure 48 : Processus de technologie d'écrémage vidéo (Kokaram, 2013)

Pour que l'approche d'écrémage soit réussie, plusieurs informations comprenant la transcription, l'analyse de la parole, du son et de l'image vidéo doit être utilisée à partir de diverses sources. Un bien exemple est le survol automatique des vidéos du journal télévisé, il faut considérer que le les vidéos sont constituées de transcriptions textuelles (Sigari et al., 2015). Premièrement, les résumés des textes sont acquis, cela peut être fait en utilisant la technique d'écrémage de l'écrémage de texte classique, après laquelle les vidéos correspondantes à ce texte sont acquises à partir de la vidéo originale. Par ainsi ce faisant, des vidéos de survol sont obtenues représentant un court synopsis vidéo original. L'objectif de le survol vidéo dans l'exemple ci-dessus consiste à acquérir et à intégrer la compréhension de l'image et la langue de la vidéo originale qui est faite en s'assurant que l'original l'information est extraite (Knox et al, 2914). Ces informations sont constituées de mots-clés audio, objets pertinents spécifiques et structures vidéo correspondantes, comme illustré à la figure 49 ci-dessous consistant en un survol vidéo.

Figure 49 : Résumé vidéo utilisant la technique de résumé vidéo dynamique

(Saba et Atameem, 2013)

Par rapport à la méthode statique, l'écrémage prend en charge la reconnaissance d'objets dans le contenu vidéo, la représentativité de l'objet est suffisante pour remplacer l'original contenu de la vidéo.

B. Résumé vidéo statique

La technique statique qui peut également être appelée storyboard statique ou image fixe résumé de résumé vidéo, aussi communément appelé le R-frame est une méthode utilisée pour résumer les images de la vidéo.

Les méthodes sont impliquées dans l'extraction d'images clés par pré-échantillonnage la séquence vidéo originale de manière aléatoire ou uniforme. L'extraction de l'image clé est essentiel pour la estion de contenu vidéo qui consiste à sélectionner une ou plusieurs images à représenter le contenu vidéo original et l'utiliser dans la génération de contenus vidéo (Jadhav et Jadhav, 2015). Une structure hiérarchique de séquence d'une vidéo est montrée dans la figure 50 ci-dessous.

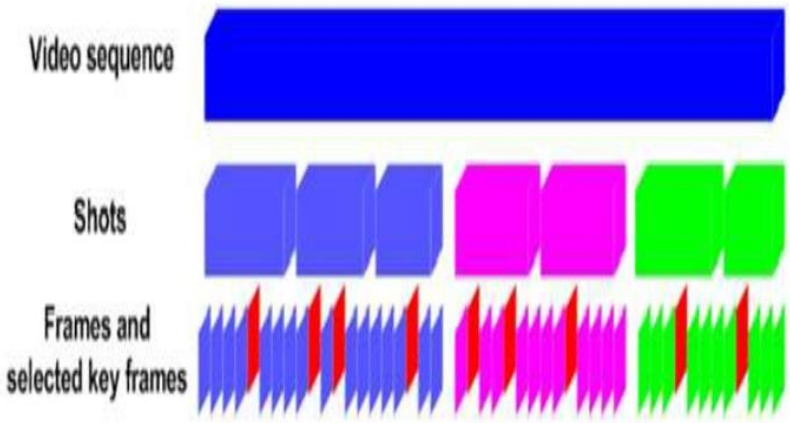


Figure 50 : Structure hiérarchique de la séquence vidéo (Jadhay&Jadhay, 2015).

L'inconvénient de cette méthode est que tous les plans apparaissent avec le même niveau d'importance pour l'utilisateur, donc cette vidéo de sortie résumée apparaît encombrante surtout pour les longues vidéos.

Les principales étapes de la technique de résumé vidéo d'images clés (résumé statique) sont (Lu et al., 2013) :

- Tout d'abord, extrayez l'image vidéo de la séquence d'images de la vidéo originale.
- La deuxième étape consiste à regrouper la trame vidéo sur la base des différents clustering algorithme, où la détection de tir est nécessaire à partir de la déduction de contenu.
- Enfin, les images clés sont sélectionnées

La figure 51 montre l'ensemble du processus de récapitulation des images clés

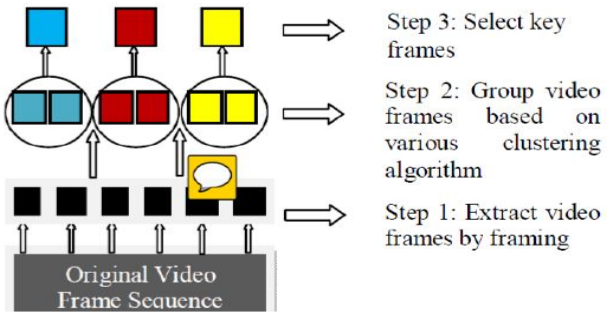


Figure 51 : Étapes de la technique de récapitulation des images clés

(Almeida et al., 2012).

Cette technique est classée de trois manières différentes (Gasic et al., 2013) :

1. Classification sur la base d'un échantillonnage
2. Classification sur la base de la segmentation de scène ou
3. Classification sur la base de la segmentation des plans.

C. Classification de l'échantillonnage

Principalement dans cette classification, les images clés qui ont un contenu similaire sont redondants puisqu'ils sont choisis aléatoirement et uniformément sous-échantillonnés, le contenu vidéo n'est pas pris en compte et le résumé produit n'est pas représentatif de toute la vidéo originale les pièces. (Almeida et al, 2013).

1. Classification de segmentation de scène

Dans cette seconde méthode de classification, la classification des images clés se fait par l'utilisant de la détection de scène, toutes les parties sont incluses dans les scènes qui ont un lien sémantique dans le vidéo originale ou dans le même temps ou dans le même espace (Almeida et al., 2013). Le seul l'inconvénient de cette méthode est que la méthode ne considère pas la trame séquentielle position.

2. Classification des segments de tir

Cette méthode de résumé d'images clés d'une vidéo les images clés adoptées sont extrait au contenu vidéo. La première image est extraite en tant qu'image clé de la prise de vue ou la première image de la prise de vue. et dernier cadre. Cette méthode de résumé des données est la plus efficace dans les cas où il y a petite variation de contenu ou plans fixes (Almeida et al., 2012). Cependant, ils font ne fournit pas une représentation suffisante du tir lorsqu'il y a des mouvements forts.

3. Résumé vidéo des techniques de regroupement

Dans cette méthode, une synthèse est appliquée en fonction des clusters appelés, sur laquelle un pré-échantillonnage est exécuté pour réduire le nombre de trames en fonction d'un taux d'échantillonnage (par exemple une image par seconde), après cette extraction de caractéristiques différentes techniques sont appliquées, y compris la couleur, la texture et la forme. Ici après un regroupement algorithme en cours d'application pour obtenir le cluster des trames d'entrée données. Enfin un cadre est sélectionné dans chaque cluster pour faire la sortie vidéo résumée L'étape qui montre comment la technique de clustering est appliquée est illustrée ci-après :

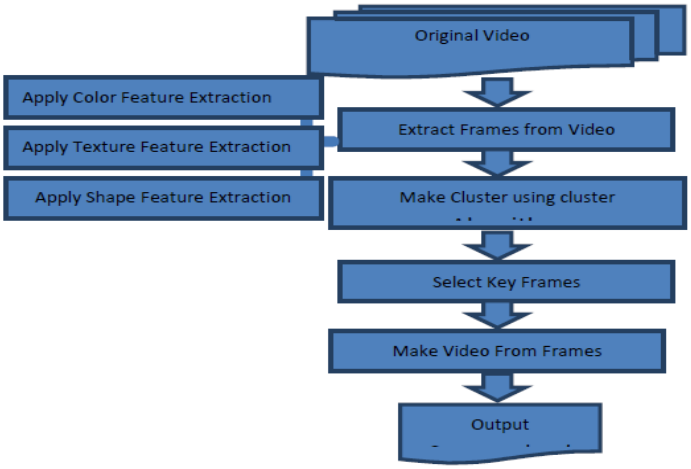


Figure 52 : Techniques de clustering Résumé vidéo

Approche VGRAPH

L'approche VGRAPH est la plus couramment utilisée dans la génération de données statiques. résumé vidéo. Pour commencer, un pré-échantillonnage de la vidéo originale est effectué ; c'est le premier étape. Ensuite, l'utilisation d'un cadre de couleur permet de segmenter la vidéo pré-échantillonnée en plans.

Après cela, l'élimination des trames de bruit et la sélection des trames représentatives de chaque coup. Enfin l'extraction des images clés se fait en utilisant la stratégie du plus proche

cadre voisin; le cadre le plus proche est construit à partir de la caractéristique de texture qui est extraits d'images représentatives de plans (Potion et al., 2014). Les étapes sont démontré dans la figure 53 ci-dessous

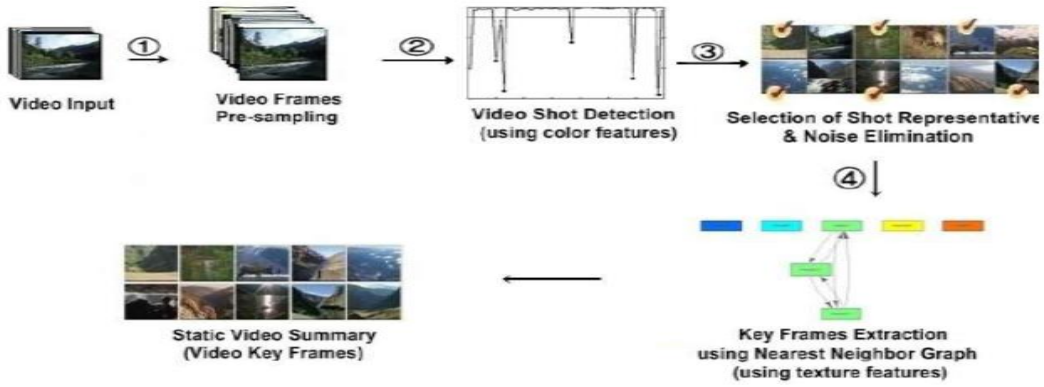


Figure 53 : Processus d'approche VGRAPH (Potapov et al, 2014).

II.7 Travaux connexes

Dans les travaux précédents, les chercheurs ont fait de la littérature connexe où ils ont proposé différentes techniques de résumé vidéo ; les travaux connexes se concentreront sur les résumés vidéo à vue unique et vidéo multi vues.

II.7.1 Résumé vidéo à vue unique

Zhang et al, en 2016 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 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.7.2 Résumé vidéo en multi vues

Nous présentons plusieurs approches sur la génération des résumé vidéo multi vues.

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

Le problème de la synthèse multi-vu vidéo introduisent Fu et al en 2010 (Nura Aljaafari, 2018) 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.

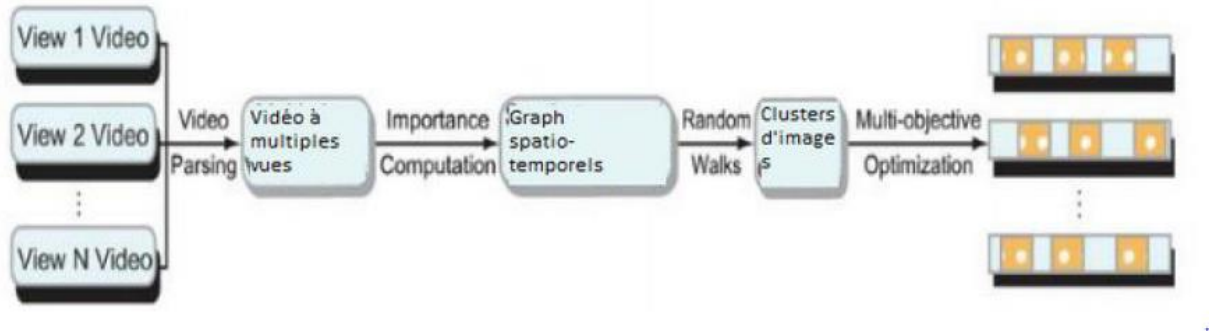


Figure 52 : Méthode de résumé vidéo multi-vu de Fu et al

Le résultat de la synthèse est généré par la résolution d'un problème d'optimisation multi-objectif basée 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-vus sont proposés par le storyboard multi-vues et le tableau d'événements présentés à la figure 53. Dans la figure 53, Le *story-board* assemble en série des plans multi-vus centrés sur les événements dans un ordre temporel.

Figure 54 : Résumé du Vidéo Story Board Multi-Vues (Nura Aljaafari, 2018)

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

Yingbo Li et Bernard Merialdo en 2010 (A. Krizhevsky, I. Sutskever, and G. E. Hinton, 2012) 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 (Szegedy, C, Liu, W, Jia, 2015) 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*).

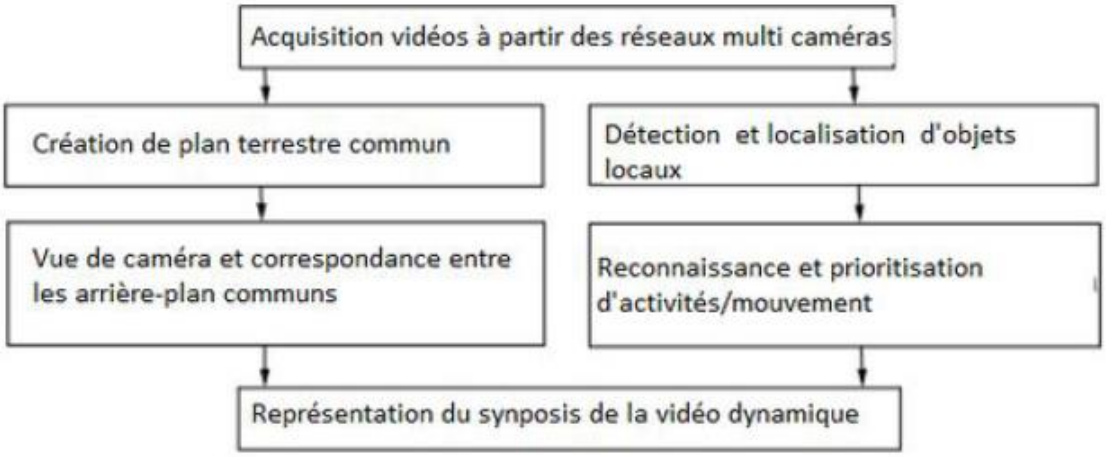


Figure 55 : Un Framework pour le synopsis vidéo multi-vues (Szegedy, C Vanhoucke, V 2016)

E. Cadre d'apprentissage métrique multi-vu

Linbo Wang et al, en 2016 [30] utilise l'apprentissage à noyaux multiples dans leurs méthode pour la résolution du 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 (Bharath Raj, 2018) 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 noeud 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 56pas élagué et présente donc une variance élevée. Le cadre proposé est donné par la figure 56.

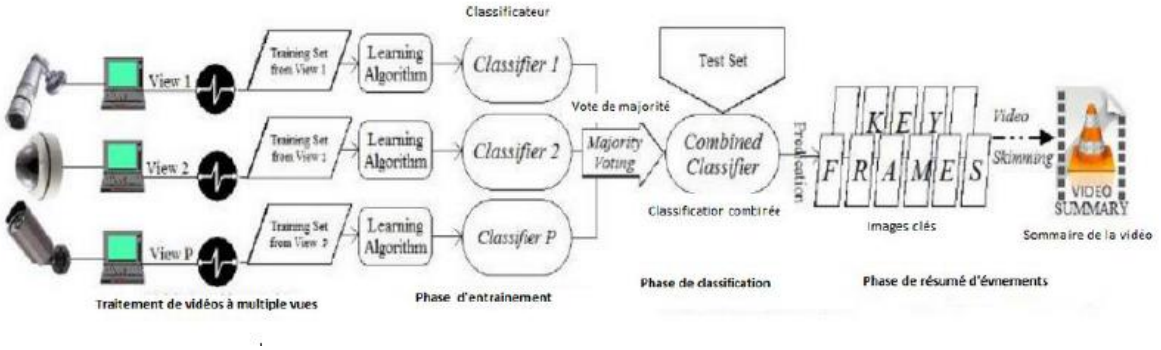


Figure 56 : Mise en sac de l'événement (E. Bressert. SciPy and NumPy, 2012)

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

II.8 Conclusion

Le chapitre est consacré pour discuter en profondeur des types et des techniques de vidéo Résumé, en plus des ouvrages du résumé vidéo.

Chapitre III : Approche proposée

III.1 Introduction

De nos jours, les réseaux de caméras de surveillance peuvent être trouvés presque partout. Le volume de données collectées par un réseau de capteurs de vision déployés dans des contextes variés allant de la sécurité à la surveillance environnementale, qui répond clairement aux exigences de grandes quantités de données.(voir la figure 57)

Les défis liés à l'évaluation et au traitement de telles quantités de données vidéo sont évidents chaque fois qu'un événement nécessite de parcourir de grandes archives vidéo pour trouver des événements pertinents.

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

Une solution à ce défi est de construire automatiquement un résumé vidéo, qui répond à cette demande en offrant un aperçu général et rapide du matériel audiovisuel de la vidéo originale, ainsi qu'en affichant les parties intéressantes.

Nous présenterons et discuterons en détail les principes de fonctionnement de notre solution basée sur le deep learning pour créer automatiquement des résumés vidéo basés sur plusieurs vues de la même scène dans ce chapitre.

III.2 Vue globale de l'approche

Pour la génération du résumé vidéo un aperçu global de notre approche est illustre dans la figure ci-dessous. (Figure 57)

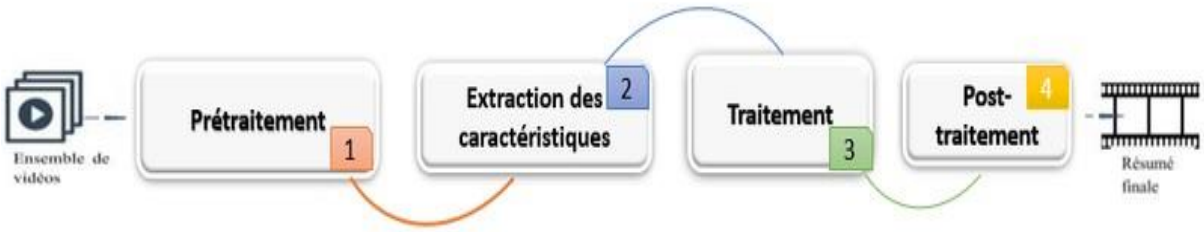


Figure 58 : Illustration de schéma globale de notre approche

Les différentes phases que constitue notre approche vont être définies par la suite :

1. Phase de prétraitement.
2. Phase d'extraction des caractéristiques.
3. Phase d'extraction des séquences frames.
4. Phase de post-traitement.

III.2.1 Phase de prétraitement

Afin qu'on puisse faire la création automatique du résumé, dans notre approche il est nécessaire de passé par la phase de prétraitement des vidéos d'entrées. Pendant cette phase on fait l'extraction des frames de chaque vidéo, ensuite redimensionner. Cette phase permet d'optimiser le temps de calcul.

Le redimensionnement des frames ne se laisse pas faire au hasard, nous utilisons 4 modèles pour poursuivre la phase à venir chaque modèle (AlexNet, GoogleNet, InceptionV3, ResNet50) accepte une dimension d'image en entrée spécifique montrer dans le tableau ci-dessous :

Modelé	AlexNet	GoogleNet	InceptionV3	Resnet50
Dimension d'image en entrée spécifique	(224,224)	(227,227)	(299,299)	(224,224)

Tableau 1 : les dimensions convenables pour chaque modèle.

Une simple image contient un nombre N de pixels, lorsqu'elle est colorée N se multiplie par 3 (N*3) correspondant aux 3 couleurs rouge R, vert V et bleu B donnant ainsi un grand chiffre envoyé dans le réseau de neurone. Avec une large séquence d'image les données à envoyer dans le réseau de neurones deviennent volumineux, lors de l'empilement couche et neurone le nombre de paramètres du réseau va exploser et le nombre de calculs va croitre de manière exponentielle. La solution alors est de transformer l'image colorée en une image avec des niveaux de gris.

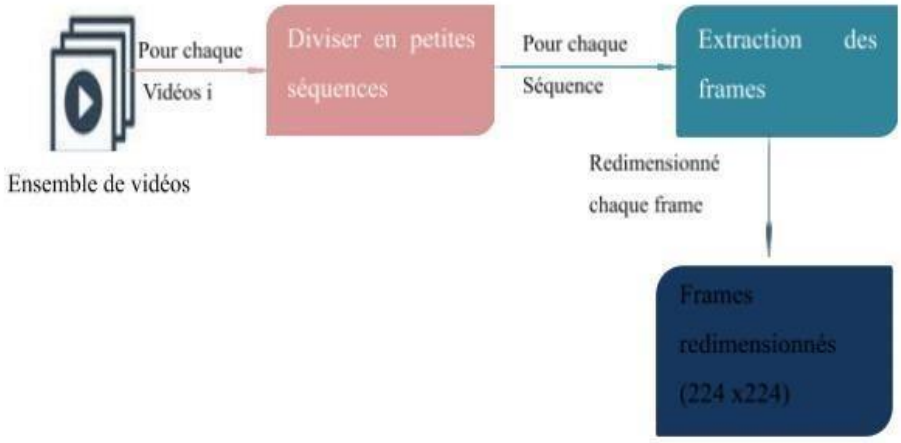


Figure 59: schéma globale de la phase prétraitement

Ainsi nos vidéos de départ seront transformés en un large dataset remplit de frames prêts à être traiter.

II. 2.2 Phase d'extraction des caractéristiques profondes

Dans cette phase l'information brute présente sur les images des segments vidéo seront extraite et analyser, pour identifier les éléments qui les constituent. En vision par ordinateur, rôle principal de ces caractéristiques est de transformer l'information visuelle sous formes de vecteurs caractéristiques.

L'extraction des caractéristiques consiste en des transformations mathématiques calculées sur les pixels. Les caractéristiques visuelles permettent généralement de mieux rendre compte de certaines propriétés visuelles de l'image,

Le schéma global de cette phase est illustré ci-dessous. (Figure 59)

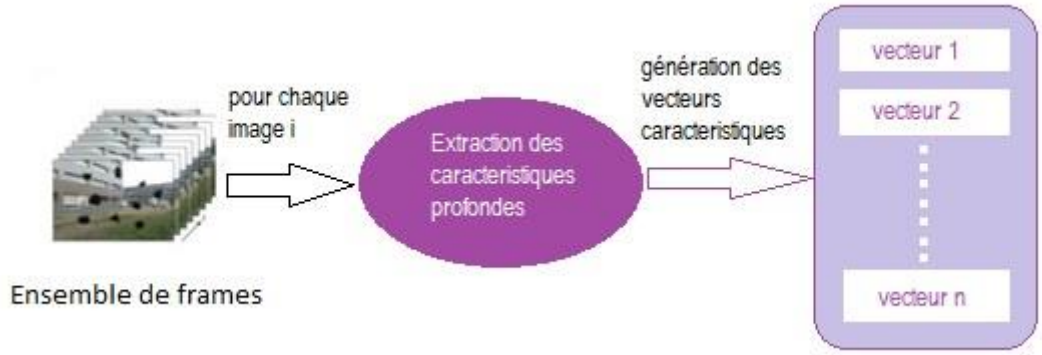


Figure 60 : Schéma globale de la phase d'extraction des caractéristiques profondes

L'extraction des caractéristiques profondes nécessite une grande puissance de calcul, nous proposons d'effectuer cette étape sur Google Colab afin de minimiser le temps.

Pour construire ces vecteurs caractéristiques des vidéos, nous avons choisi une architecture profonde obtenue en entrainant un réseau convolutif tridimensionnel, nous avons injecter les frames obtenus de la phase du pré-traitement dans plusieurs modèles afin de pouvoir plutard faire une étude comparative sur les quatre modèles CNN suivants : AlexNet, GoogleNet, Inception V3, ResNet50.

Nous obtenons ainsi plusieurs vecteurs de caractéristiques pour chaque réseau convolutif tridimensionnel.

III.2.3 Phase d'extraction des séquences frames

Apres l'extraction de toutes les caractéristiques profondes, nous avons utilisé un autre type de réseau de neurones récurrent, le LSTM-bidirectionnel (Voir détails dans le chapitre 1)

Nous avons empilé deux couches de LSTM l'une sur l'autre pour créer un LSTM bidirectionnel dans l'architecture proposée, ce qui facilite l'apprentissage des changements à long terme. Les caractéristiques profondes extraites en utilisant différents modèles de CNN sont par la suite propagées vers RNN pour déterminer si une séquence de trames est informative ou non.

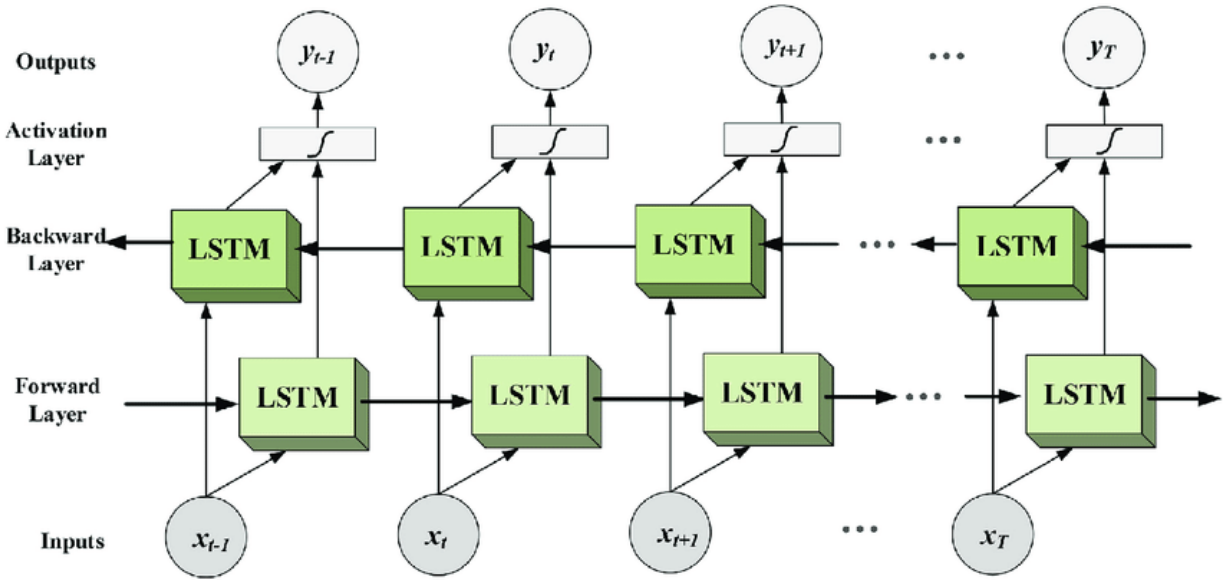


Figure 61 : Représentation d'un réseau lstm bidirectionnel

Afin de prendre que les séquences pertinentes, chaque vecteur de caractéristiques représentant l'image se tient en face d'une ligne composé de 1 ou de 0 pour dire respectivement qu'une image est pertinente ou pas comme le montre la figure ci-dessous, autrement dit les images de cette manière seront mises en face des labels.

Pour faire l'entrainement de chacun de nos modèles les données rentrerons dans un lstm, pas un lstm simple mais un lstm bidirectionnel car nous auront besoin que nos données fassent un passage vers l'avant traversant des neurones calculons ainsi les poids entres chaque nœud et sont prochain, dès qu'il parvient à la fin de notre réseau il fera rencontre avec les données étiquetées puis retourne par la suite en arrière afin de modifier le poids W calculer lors de passage en avant, optimisant ainsi nos modèles pour prédire de meilleur résultat.

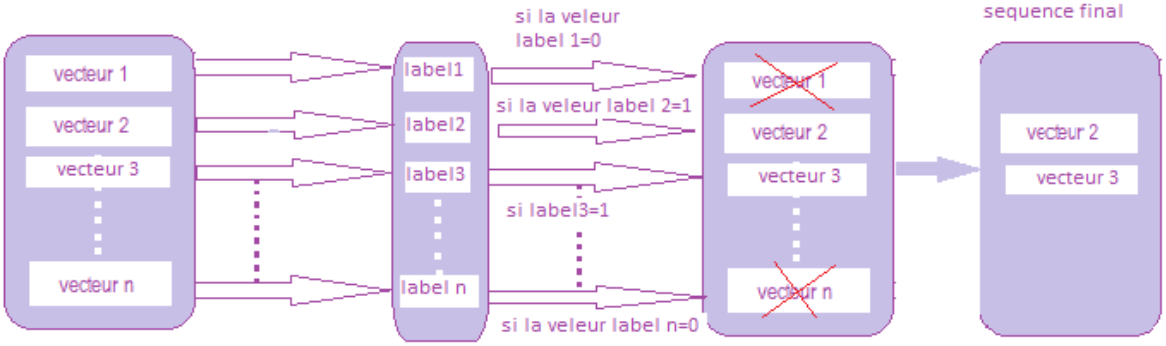


Figure 62 : Représentation de la phase d'extraction des séquences

Les données notées auparavant comme données inconnu, seront dorénavant des données dont nous connaissons déjà la réponse cible, nous les appelons ainsi des données étiquetées.

Enfin, nous avons obtenus 4 modèles LSTM qui peuvent identifier les séquences comme informatives ou non informatives, aidant à la création du résumé final.

III.2.4 Phase de post-traitement

Apres l'extraction des séquences frames importantes, nous passons à la prochaine et dernière étape de production de notre résumé finale des vidéos originaux, en se basant sur un réseau de neurone LSTM bidirectionnel qui est une extension du réseau de neurone RNN, le choix de ce réseau de neurone bidirectionnel est dû à l'exécution de deux manière l'une du passé au futur et l'autre du futur au passé, concevant ainsi les informations des entrées qui lui ont déjà été transmises en utilisant l'état masqué.

Dans cette phase nous entrainons nos quatre modèles sur avec les séquences frames obtenus jusqu'à ce que nos modèles puissent être capable

Il est possible que certaines séquences visuellement similaires soient prises en considération dans le résume final, ceci cause un problème lors de la sélection des séquences dont la probabilité est maximale pour la classe d'informativité.

Pour obtenir un résumé représentatif les images les mieux ajustées sont prises en compte pendant de l'étape de post-traitement, les images ci-dessous présentes les frames du résumé final avec la probabilité maximale.

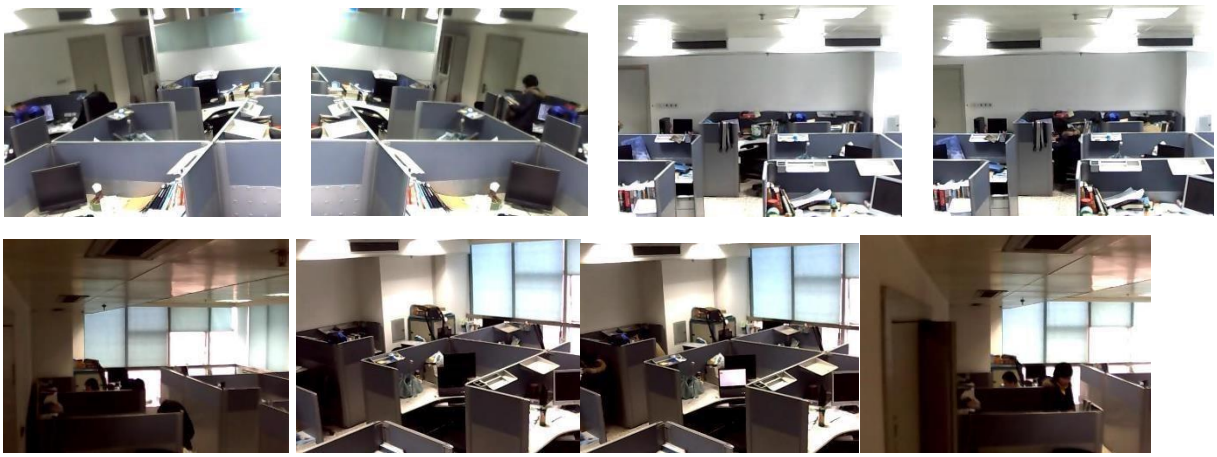


Figure 62 : Illustration des images de résumé final

III.3 Conclusion

Nous avons détaillé notre approche du résumé vidéo dans ce chapitre, qui est basée sur l'apprentissage profond et utilisé une architecture neuronale basée sur des réseaux de neurones convolutif pour extraire les caractéristiques profondes de chaque frame d'une séquence et les transmettre à un autre réseau de neurones LSTM pour acquérir des probabilités d'information et générer un résumé vidéo dynamique.

Chapitre IV : Test et résultats

IV.1 Introduction

Nous avons défini notre approche de la création automatique de résumé vidéo, ainsi que tous les concepts qui l'accompagnent, et nous sommes maintenant prêts à la mettre à l'épreuve. Ce chapitre décrit l'environnement matériel et logiciel dans lequel nous avons travaillé, ainsi que le jeu de test sur lequel nous avons travaillé et les métriques que nous avons utilisées. Enfin, nous terminerons ce chapitre par un résumé des résultats des tests.

IV.2 Environnement matériel

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

- **Marque :** Asus VivoBook
- **Processeur :** Intel(R) Core (TM) i7-8550U CPU @ 1.80GHz 1.99 GHz
- **Carte graphique :** Intel (R) UHD Graphics 620
- **Mémoire :** 8,00 Go
- **System d'exploitation** : Windows 10 Professionnel, 64 bits

IV.3 Environnement logiciel

Python 

Python est un langage de programmation interprété multi-paradigme. Il favorise la programmation impérative structurée, et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions, il est ainsi similaire à Perl, Ruby, Scheme, Smalltalk et Tcl.

Le langage Python est placé sous une licence libre proche de la licence BSD et fonctionne sur la plupart des plates-formes informatiques, des supercalculateurs aux ordinateurs centraux, de Windows à Unix en passant par Linux et MacOs, avec Java ou encore .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser. Il est également apprécié par les pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation plus aisée aux concepts de base de la programmation

Quelques avantages du langage python :

- Python peut être **étendu à d'autres langues**.
- Python est simple et facile (**Facile à apprendre, comprendre, et coder**).
- Proche du langage C.
- Python est pour tout le monde (Le code Python peut s'exécuter sur n'importe quelle machine, que ce soit Linux, Mac ou Windows.
- Python est gratuit (libre et open source)
- Pas de perte de temps pour déclarer les types, variables, ….
- Types de données complexes intégrés (listes, …).

Numpy 

NumPy est un package utiliser pour les calculs scientifiques en Python. Il est idéal pour les opérations liées à l'algèbre linéaire, aux transformations de Fourier, ou au crunching de nombres aléatoires. Il peut être utilisé en guise de container multi-dimensionnel de données génériques. De plus, il s'intègre facilement avec de nombreuses bases de données différentes.

OpenCv 

Opencv est une bibliothèque graphique libre, initialement développée par Intel, spécialisée dans le traitement d'images en temps réel. La société de robotique Willow Garage et la société ItSeez se sont succédé au support de cette bibliothèque. Depuis 2016 et le rachat de ItSeez par Intel, le support est de nouveau assuré par Intel. Cette bibliothèque est distribuée sous licence BSD.

Elle met à disposition de nombreuses fonctionnalités très diversifiées permettant de créer des programmes en partant des données brutes pour aller jusqu'à la création d'interfaces graphiques basiques

Scipy 

Scipy est une bibliothèque pour les calculs techniques et scientifiques. Elle regroupe des modules pour les tâches de science des données et d'ingénierie telles que l'algèbre, l'interpolation, le FFT, ou le traitement de signaux et d'images.

OS

Le module os en Python permet d'interagir avec les fonctionnalités du système d'exploitation et d'accéder aux informations. De plus, le module os nous permet de travailler avec les fichiers et les répertoires

CSV 

Csv est un format de fichier simple utilisé pour stocker des données tabulaires, telles qu'une feuille de calcul ou une base de données. Un fichier CSV stocke des données tabulaires (chiffres et texte) en texte brut. Chaque ligne du fichier est un enregistrement de données. ... Pour travailler sur des fichiers CSV en python, il existe un module intégré appelé csv.

Pathlib

Ce module offre des classes représentant le système de fichiers avec la sémantique appropriée pour différents systèmes d'exploitation. Les classes de chemins sont divisées en chemins purs, qui fournissent uniquement des opérations de manipulation sans entrées-sorties, et chemins concrets, qui héritent des chemins purs et fournissent également les opérations d'entrées-sorties.

Time

Le module time en python offre la possibilité de lire, représenter et réinitialiser les informations de temps de nombreuses façons.

Math

Ce module est toujours disponible. Il fournit l'accès aux fonctions mathématiques définies par le standard C

Sklearn

Scikit-learn est une bibliothèque libre Python destinée à l'apprentissage automatique. Elle est développée par de nombreux contributeurs2 notamment dans le monde académique par des instituts français d'enseignement supérieur et de recherche comme Inria3.

Elle propose dans son framework de nombreuses bibliothèques d'algorithmes à implémenter, clé en main. Ces bibliothèques sont à disposition notamment des data scientists.

Elle comprend notamment des fonctions pour estimer des forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec d'autres bibliothèques libres Python, notamment NumPy et SciPy

Tensorflow

TensorFlow est une bibliothèque open source de Machine Learning, créée par Google, permettant de développer et d'exécuter des applications de Machine Learning et de Deep Learning. Découvrez tout ce que vous devez savoir à son sujet.

Keras

La bibliothèque Keras permet d'interagir avec les algorithmes de réseaux de neurones profonds et d'apprentissage automatique, notamment Tensorflow, Theano, Microsoft cognitive Toolkit ou PlaidML.

Conçue pour permettre une expérimentation rapide avec les réseaux de neurones profonds, elle se concentre sur son ergonomie, sa modularité et ses capacites d'extension. Elle a été développée dans le cadre du projet ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System). Elle a été initialement écrite par François Chollet.

Google Colab

Google Colab ou Colaboratory 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. Cool, n'est-ce pas ? Avant de présenter ce magnifique service, nous rappellerons ce qu'est un Jupyter Notebook.

IV.4 **Ensemble de donnée (Dataset)**

Dans ce qui suit, nous présentons notre jeu de données multi-vues de notre dataset représentant le dataset Office qui prend en compte un bureau de travail open space et des travailleurs.

IV.4 .1**Office :**

Il s'agit du dataset Office pris par 4 caméras de surveillances stables dans un bureau avec une angle de vue de 180°. Tandis que les caméras sont fix mais elles ne sont pas stables leurs vibrations ainsi que le changement de luminosité, et le fait que certaines vidéos sont synchronisées et d'autres ne le sont pas rendent la production d'un résumé vidéo idéal plus difficile.

- **La vue de la 1ere camera**

Le dataset du vidéo fourni par cette caméra comme le montre les images ci-dessous est le plus éclairé par rapport aux autres.

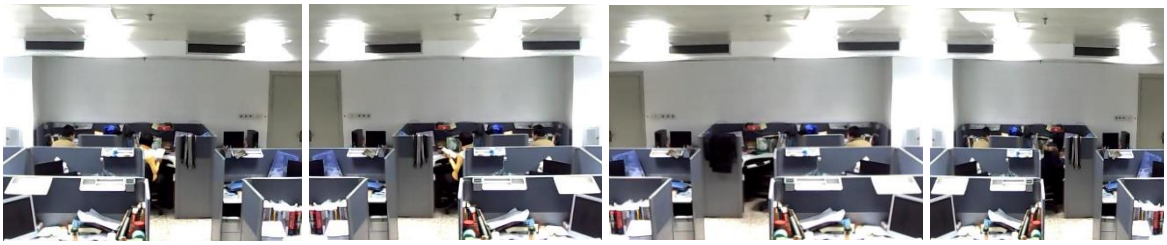


Figure 64 : illustration de la vidéo de la première vue (caméra 1)

- **La vue de la 2éme camera**

Le dataset du vidéo fourni comme le montre les images ci-dessous on remarque un éclairage diffèrent dans le même dataset et moins lumineux que le premier à cause de l'emplacement de camera captant la vidéo et des changements de climat ou bien de temps.

Figure 65 : Illustration de la vidéo de la deuxième vu (caméra 2)

- **La vue de la 3éme camera**

Dans ce dataset non seulement le changement climatique et temporel à un impact sur la difficulté de choisir les frames disant pertinentes mais aussi l'emplacement de la caméra près d'une porte nous cache quelques informations qui peuvent êtres essentiels dans la production de notre résumé vidéo.

Figure 66 : Illustration de la vidéo de la troisième vue (caméra 3)

- **La vue de la 4éme camera**

Tel que les autres vidéos fournis par les autres camera l'éclairage dans se dataset diffère tandis que la caméra est fix.

Figure 67 : Illustration de la vidéo de la quatrième vu (caméra 4)

IV.4 .2. **Les mesures d'évaluation**

Pour mesurer la qualité de notre résumé vidéo, Nous s'avons intéressé à une mesure utilisée dans plusieurs travaux liés à l'apprentissage automatique. Pour dire qu'un modèle de classification est un bon modèle nous s'intéressant à la mesure nommé la précision (accuracy).

Afin d'opté pour une meilleur compréhension et pour pouvoir interpréter nous définissons :

- **La précision du modèle (accuracy)**

En apprentissage automatique est la mesure utilisée pour déterminer quel modèle est le mieux à même d'identifier les relations et les modèles entre les variables d'un ensemble de données en fonction des données d'entrée ou d' apprentissage . Mieux un modèle peut généraliser aux données « invisibles », meilleures sont les prédictions et les informations qu'il peut produire, ce qui à son tour offre plus de valeur commerciale.

$$Précision = \frac{nombre\ de\ frame\ pertinents\ retrouvés}{\text{nombre de frame retrouvés}}$$

- **Une epoque (epoch)**

Dans l'apprentissage automatique signifie un passage complet de l'ensemble de données d'entraînement à travers l'algorithme. Ce nombre d'époques est un hyperparamètre important pour l'algorithme. Il spécifie le nombre d'époques ou de passes complètes de l'ensemble de données d'entraînement passant par le processus d'entraînement ou d'apprentissage de l'algorithme.

- **La perte (Loss)**

Une fonction de perte est utilisée pour optimiser un algorithme d'apprentissage automatique. La perte est calculée sur l'apprentisage et la validation, son interprétation est basée sur la performance du modèle dans ces deux ensembles. Il s'agit de la somme des erreurs commises pour chaque exemple dans les ensembles d'apprentissage ou de validation. La valeur de perte implique à quel point un modèle se comporte mal ou bien après chaque itération d'optimisation.

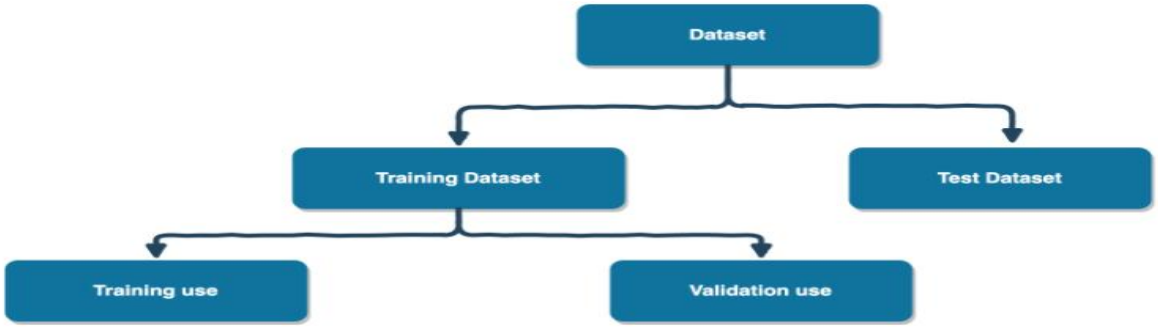


Figure 68 : Illustration de devisions du dataset en training set et test set.

L'une des combinaisons de métriques les plus couramment utilisées est la perte d'entraînement + la perte de validation au fil du temps :

- **La perte d'apprentissage :** cette valeur indique dans quelle mesure le modèle s'adapte aux données d'apprentissage
- **La perte de validation :** cette valeur indique dans quelle mesure le modèle s'adapte aux nouvelles données.

Afin d'interpréter notre modèle et de dire qu'il est satisfaisant nous surveillons les valeurs loss mentionner dans le paragraphe ci-dessus pour prédire que notre modèle subit un :

- **Surapprentissage (Overfiting)**

Un modèle est dit sur-ajuster s'il est surentraîné sur les données de telle sorte qu'il en apprenne même le bruit. Un modèle de sur-ajustement apprend chaque exemple si parfaitement qu'il classe mal un exemple invisible/nouveau.

Pour un modèle sur-ajusté, nous avons un score d'ensemble d'entraînement parfait/proche de parfait alors qu'un score de test/validation médiocre.

Le overfiting peut apparaitre lors de l'utilisation d'un modèle complexe pour un problème simple qui capte le bruit des données. Exemple : Ajustement d'un réseau de neurones à l'ensemble de données Iris, ou a cause d'avoir un petit ensemble de données, car l'ensemble d'apprentissage peut ne pas être une représentation correcte de l'univers.

- **Sous-ajustement (Underfiting)**

Un modèle est dit sous-adapté s'il est incapable d'apprendre correctement les modèles des données. Un modèle underfit n'apprend pas complètement chaque exemple dans l'ensemble de données. Dans de tels cas, nous constatons un faible score à la fois pour l'ensemble d'apprentissage et l'ensemble de test/validation.

Le underfit peut apparaitre lors l'utilisation d'un modèle simple pour un problème complexe qui n'apprend pas tous les modèles dans les données. Exemple : Utilisation d'une régression logistique pour la classification d'images.

Les données sous-jacentes n'ont pas de modèle inhérent. Exemple, essayer de prédire les notes d'un élève avec le poids de son père.

Figure 69 : Une image représentative du overfit et underfiting.

- **AdamOptimizer (Adaptive Moment Estimation)**

Adam est un algorithme d'optimisation qui peut être utilisé à la place de la procédure classique de descente de gradient stochastique pour mettre à jour les poids du réseau de manière itérative en fonction des données d'apprentissage.

IV.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.

IV.5 .1. Etude comparative de nos modèles

Afin d'interpréter nos résultats, nous sommes intéressés à une mesure souvent utilisée dans les travaux liés à l'apprentissage automatique, à savoir la précision (accuracy) est utile lorsque toutes les classes sont d'égale importance, Puisque nous procédons plusieurs modèles cette mesure nous aidera à faire une comparaison de meilleure précision.

Afin de la calculer, nous définissons les valeurs dans le tableau suivant qui montrent les résultats de précision de résumé vidéo sur notre jeu de donnée multi-vues « Office » pour chaque camera (4 vues)

- Nous avons choisi un nombre d'epoque équivalent à 400.

Le modèle	AlexNet	Googlent	Inception V3	ResNet 50
Caméra1 1ére vue	83.69%	82.18%	98.81%	82.46%
Camera 2 2éme vue	98.53%	93.53%	70.16%	99.39%
Camera3 3éme vue	99.5%	97.62%	97.32%	97.11%
Camera 4 4éme vue	80.24%	99.05%	84.92%	99.19%

Tableau 2: comparaison des performances des quatre modèles basés sur précision.

– Nous avons utilisé diffèrent valeur d'optimiser afin d'améliorer chaque modèle.

Modele	Optimizer			
AlexNet	0.000005	0.00005	0.00005	0.00002
GoogleNet	0.000004	0.0000001	0.00004	0.00004
Inception v3	0.000005	0.000002	0.00003	0.00002
ResNet	0.000005	0.000001	0.000001	0.0004

Tableau 3 : Les valeurs optimizer necessaire pour ameliorer les modeles.

Les graphs

– Dans ce cas notre modèle est basé sur AlexNet.

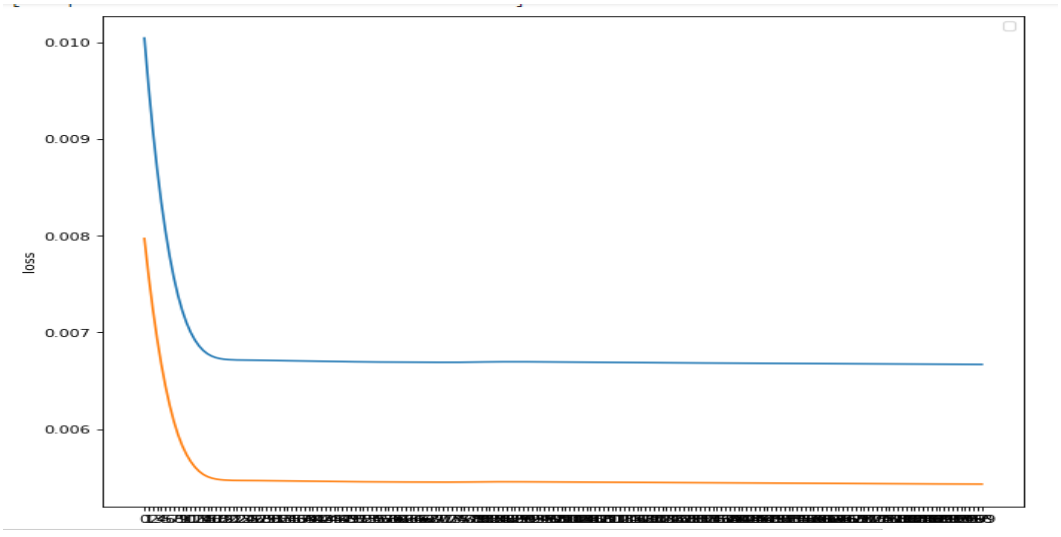


Figure 70: graph représentant loss validation et train loss dans le modèle AlexNet.

– Dans ce cas notre modèle est basé sur GoogleNet.

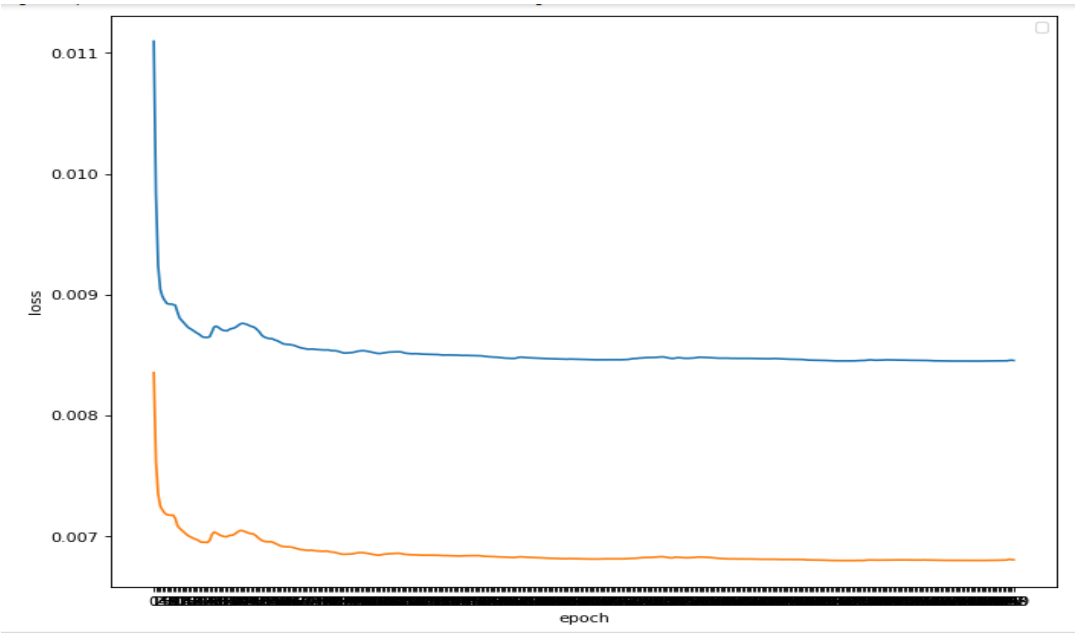


Figure 71 : graph représentant loss validation et train loss dans le modèle GoogleNet.

– Dans ce cas notre modèle est basé sur Inception V3.

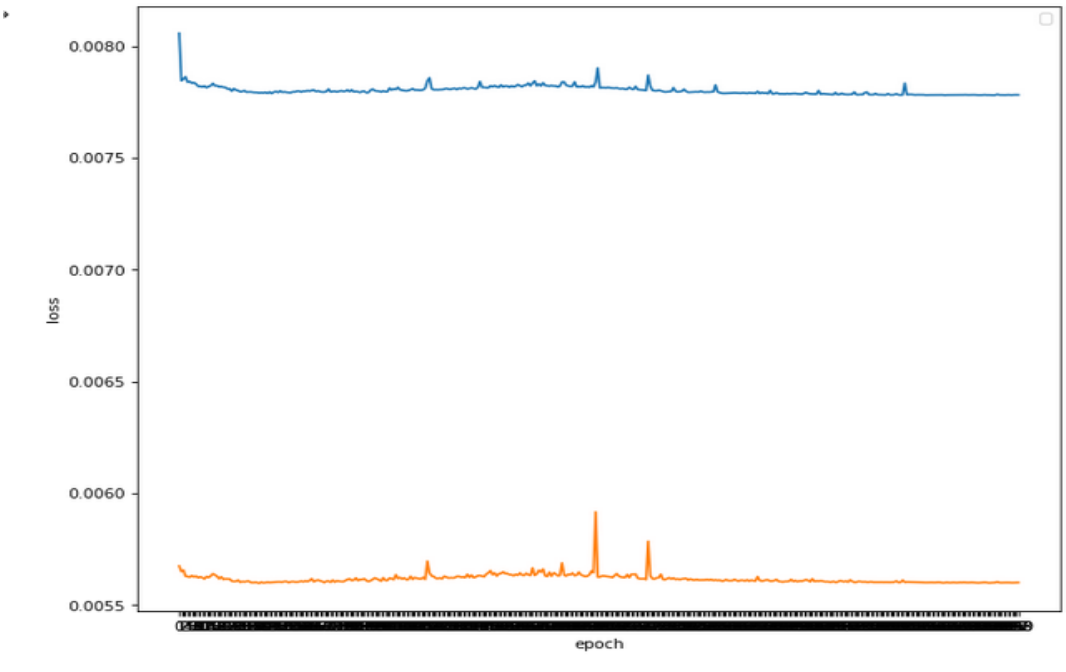


Figure 72 : graph représentant loss validation et train loss dans le modèle Inception V3.

– Dans ce cas notre modèle est basé sur ResNet50.

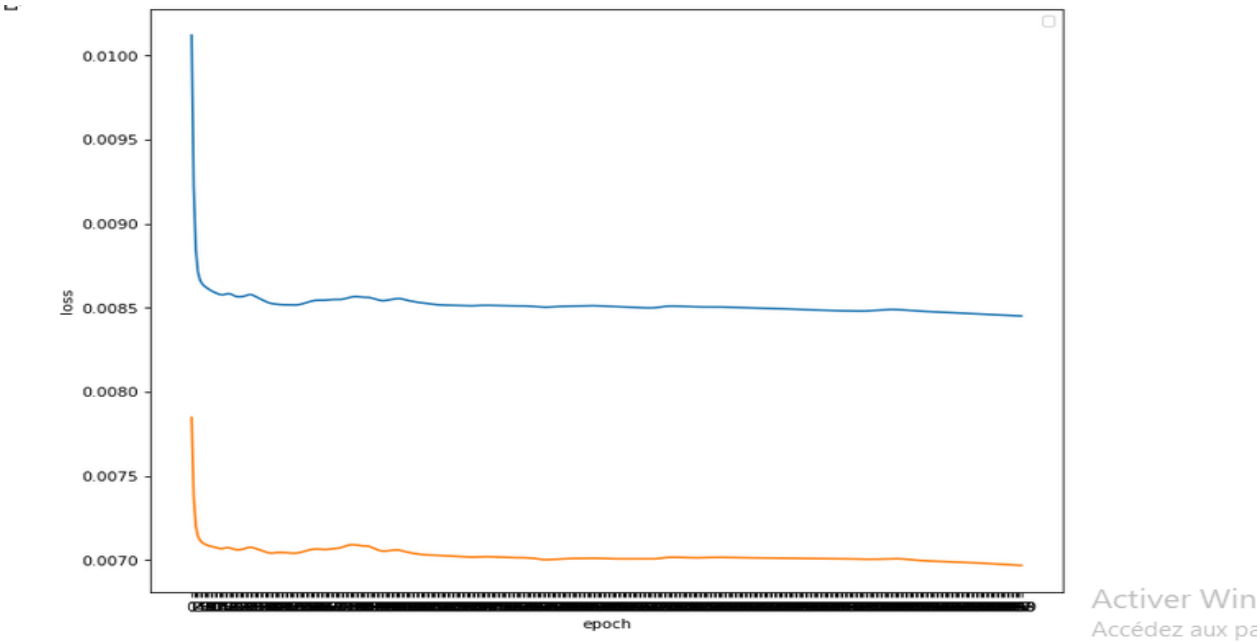


Figure 73 : graph représentant loss validation et train loss dans le modèle ResNet50.

- Les modèles AlexNet, GoogleNet, Resnet50 on montrer que nos modèles on toujours tendance à s'améliorés et à apprendre à fur et à mesure que l'entrainement se poursuit, la perte d'entrainement ainsi que la perte de validation diminue signifiant que les modèles ont toujours tendance à apprendre, les courbes dans les graphs sont prometteuses montrant ainsi que nos modèles sont entrainés convenablement et peuvent données de bons résultats.
- Malgré que la courbe de modèle Inception V3 dans certaine epoch sa capacité d'apprendre à diminuer mais s'est rapidement rattrapé durant les époque suivante, l'entrainement s'améliora ainsi, ceci est déduit en remarquant les changements de la valeur de perte.
- AlexNet stabilise après un nombre d'epoch inferieur à GoogleNet et InceptionV3.
- ResNet apporte de meilleur résultat concernant la valeur de perte par rapport aux autres modèles.
- Aucun de nos modèles ne souffre d'un overfiting ni de underfiting.

IV.5.2 Comparaison d'architecture du reseau de neurone convolutifs pour l'entrainement

Avec le paramètre 400 epoch et sur un entraînement basé sur les mêmes caractéristiques extraites nous obtenons les résultats qui sont présentés dans la figure suivante

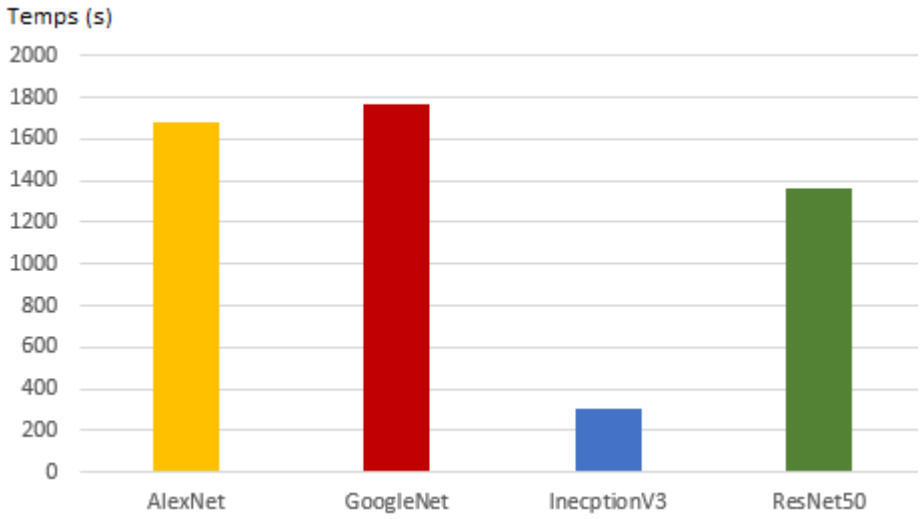


Figure 74 : Histogramme montrant la durée de traitement de chaque modèle en seconde.

D'après la figure ci-dessus, nous constatons que le modèle InceptionV3 est le plus rapide à l'exécution, Suivi de ResNet50 puis AlexNet (variances de dizaines de minute constatés).

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 ».

Conclusion générale

Conclusion générale

Les réseaux de surveillance sont presque omniprésents dans le monde d'aujourd'hui. Ces réseaux produisent des vidéos quotidiens 24 heures sur 24, avec une redondance élevée, gaspillant du stockage et compliquant l'analyse. Motivés par ces problèmes, nous avons suggéré dans notre mémoire un outil de synthèse vidéo multi-vu efficace basée sur CNN et RNN.

Nous avons utilisé une architecture CNN pour extraire les caractéristiques profondes d'une séquence d'images, puis nous avons prédit une probabilité pour chaque trame de la vidéo d'indiquer si l'image est sélectionnée ou non dans le résumé final. Ensuite nous avons utilisés une architecture neuronale basée sur les réseaux de neurones récurrents à longue « mémoire court-terme » (LSTM) bidirectionnelle qui prend les fonctionnalités spatiaux-temporelles présentes dans les images de la vidéo pour la génération dynamique du résumé finale.

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

- Nous avons utilisé le modèle CNN-RNN à calcul intensif, que nous souhaitons remplacer par un modèle d'apprentissage en profondeur ayant une précision similaire ou supérieure.
- Utiliser une stratégie d'optimisation pour travailler sur diverses tailles de vidéo, y compris des vidéos de grande taille, pour étudier les effets de la taille de la vidéo et appliquer des résumés vidéo appropriés.
- Augmenter le temps d'extraction des caractéristiques (Alexnet, Googlenet, Inception, Resnet), ce qui reste une procédure longue malgré son efficacité.

Bibliographie

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

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

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

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

[5]: 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.

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

[7]: 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.

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

[9]: 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.

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

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

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

[13]: 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.

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

[15] : 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

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

[17]: 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

[18]: 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.

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

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

[21]: 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

[22]: 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

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

[24]: 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

[25]: 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.

[26]: 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.

[27]: A. Krizhevsky, I. Sutskever, and G. E. Hinton, " *Imagenet classification with deep*

convolutional neural networks ", 2012

[28]: 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

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

[30]: 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

[31]: https://www.backstreet-surveillance.com/education-advice-tips/business-camera-placement.ht ml consulter le 05/08/2020 .

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

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

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

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

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

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

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

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

[40]: 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*

[41] : 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

[42]: 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.