

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

UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU



FACULTE DU GENIE ELECTRIQUE ET D'INFORMATIQUE  
DEPARTEMENT D'INFORMATIQUE

## Mémoire de Fin d'Etudes de MASTER ACADEMIQUE

Domaine : **Mathématiques et Informatique**

Filière : **Informatique**

Spécialité : **Conduit de Projets Informatique (CPI)**

*Présenté par*

**Soria Ouhocine**

**Nora Ikhlef**

Thème

## Détection de visage basée sur MapReduce dans le domaine de vidéosurveillance.

*Mémoire soutenu publiquement le 13/07/ 2016 devant le jury composé de :*

**Président : M<sup>r</sup> Mhand kerbiche**

**Encadreur : M<sup>r</sup> Ahmed Dib**

**Examineur : M<sup>r</sup> Mouhamed said Habet**

**Examineur : M<sup>r</sup> Loukman Chebouba**

# *Remerciement*

*Tous d'abord, nous remercions le bon dieu de nous avoir guidés vers le chemin  
de la lumière et du savoir.*

*Nous tenons à remercier vivement notre dévoué promoteur M' A.DIB pour son  
encadrement et pour sa contribution à travers sa disponibilité, son temps et son  
expérience.*

*Nous remercions vivement les membres de jury qui nous feront l'honneur  
d'évaluer notre travail.*

*Nous remercions tous ceux qui croient à la science et au savoir.*

# Sommaire

# Sommaires

## Chapitre 1 :

1	Introduction .....	01
2	Système de vidéo de surveillance.....	01
3	Technologie .....	05
3.1	Architecture des systèmes de vidéosurveillance .....	05
3.1.1	Composantes d'un système de vidéo surveillance .....	05
3.1.1.1	Acquisition .....	05
3.1.1.2	Traitement .....	07
3.1.1.3	Archivage .....	07
3.1.1.4	Affichage .....	08
3.1.2	évolution des systèmes de vidéo de surveillance.....	08
3.1.2.1	première génération : le tout analogique.....	08
3.1.2.2	deuxième génération : le système hybride.....	09
3.1.2.3	troisième génération : le tout numérique IP .....	10
3.1.3	vidéo de surveillance IP .....	10
3.1.4	architecture d'un réseau de vidéosurveillance intelligente .....	12
3.1.4.1	architecture centralisée .....	13
3.1.4.2	architecture distribuée .....	13
4	Domaines d'application .....	13
4.1	Gouvernement et sécurité publique.....	14
4.2	Education .....	15
4.3	Commerce de détail .....	15
4.4	Transports .....	15
4.5	Milieu bancaire .....	16
4.6	Autres .....	16
5	Indexation et recherche des vidéo surveillanc.....	16
5.1	Indexation et recherche .....	17
6	Conclusion .....	19

## Chapitre 2 :

1	Introduction :.....	20
2	Présentation de MapReduce .....	20
2.1	Définition de MapReduce .....	20
2.2	L'algorithme .....	20
2.3	Comment écrire un programme MapReduce .....	21
2.4	Les composants d'un processus MapReduce .....	24
2.5	Domaines d'application .....	25
2.6	Exemple illustratif (wordcount) .....	26
2.6.1	Monde de fonctionnement .....	26

2.6.2 Intégration ou utilisation de MapReduce avec des framework de cloud computing .....	29
2.6.2.1 Hadoop .....	29
2.6.2.2 Définition de Hadoop et ces composants .....	29
2.6.2.3 Architecture de Hadoop.....	31
2.6.2.4 MapReduce dans Hadoop .....	34
a. Etape du découpage des données (split) .....	35
b. Etape du Map .....	35
c. Etape Reduce .....	35
d. Etape Combiner (facultative) .....	35
3 Conclusion .....	36

## Chapitre 3

1 Introduction .....	37
2 CBVIR (Content Based Video Indexing and Retrieval).....	37
2.1 Description générale d'un système CBVR.....	37
2.1.1 Indexation .....	37
2.1.2 Recherche.....	38
2.2 Les domaines d'application de l'indexation et la recherche de la vidéo... ..	39
3 Système de détection de visage humain.....	40
3.1 Système de détection.....	40
3.2 Présentation.....	40
3.3 Caractéristique du visage humain.....	40
3.3.1 Domaine d'application.....	41
3.4 Détection des visages humains et apprentissage du système.....	42
3.4.1 Méthodes d'apprentissage (approches statistiques).....	42
4 Enquête sur les méthodes de détection du visage humain.....	44
4.1 Les méthodes basées sur la connaissance.....	45
4.2 Les méthodes basées sur des caractéristiques invariantes.....	46
4.3 Les méthodes d'appariement de modèles.....	46
4.4 Les méthodes basées sur l'apprentissage de modèles.....	46
4.5 Algorithme de Viola et Jones.....	47
4.6 Réseau de neurones.....	47
5 Conclusion.....	47

## Chapitre 4 :

1.Introduction.....	48
2.structure de système .....	48

2.1. HDFS .....	48
2.2. namenode .....	49
2.3 Namenode secondaire .....	50
2.4 Dtatanode .....	50
2.5 MapReduce .....	50
3. algorithme de détection de visage utilisé .....	51
3.1. le détecteur de viola et jones .....	51
3.2 le détecteur de viola et jones .....	51
3.3 apprentissage détecteur du viola et jones .....	51
3.4 algorithme adaboost de viola et jones pour la détection du visage.....	52
3.4.1 Détection en cascade.....	52
4. algorithme de detection de visage Avec mapReduce .....	52
4.1. algorithme avec Map.....	52
4.2 algorithme avec reduce .....	53
5. realisation et environnement.....	53
5.1. Dans eclipse .....	53
5.2. dans hadoop .....	54
6. Plateforme utilisé .....	54
6.1. virtuelBox .....	55
6.2. apache Hadoop 2.7.1 .....	55
6.3. Ubuntu 14.04.....	55
6.4. OpenImaj.....	56
7. conclusion .....	57

# Tables des figures

## Tables des figures

### Chapitre 1 :

Figure1 : schéma d'un système de vidéo de surveillance .....	1
Figure2 : caméras de surveillance .....	2
Figure 3 : Multiplexeur vidéo .....	2
Figure 4 : l'intérieur d'un enregistreur vidéo numérique .....	3
Figure 5 : pupitre de commande PTZ .....	3
Figure 6 : un centre du contrôle d'une ville.....	4
Figure 7 : entrée/sortie d'un enregistreur vidéo numérique.....	4
Figure 8 [4] : l'architecture générale de l'indexation et de la recherche de vidéo de vidéosurveillance .....	18

### Chapitre 2 :

Figure 1 : fonctionnement du MapReduce.....	21
Figure 2 : Architecture basique de MapReduce.....	21
Figure 3 : les composants d'un processus MapReduce.....	25
Figure 4 : exemple de wordcount.....	28
Figure 5 : architecture de HDFS.....	31
Figure 6 : architecture de Hadoop.....	32
Figure 7 : position de MapReduce dans Hadoop.....	34

### Chapitre 3 :

Figure 1 : phase d'indexation par le contenu.....	39
Figure 2 : phase de recherche par le contenu.....	39
Figure 3 : Détection de visage par classification.....	42
Figure 4 : Classification d'une région de l'image.....	43
Figure 5: principe de réseau de neurone pour la détection du visage.....	47

### Chapitre 4 :

Figure 1 : exemple de caractéristiques pseudo-Haar.....	51
Figure 2 : architecture de cascade.....	52
<b>Figure 3</b> : l'image avant l'exécution du programme java dans eclipse.....	53
<b>Figure 4</b> : l'image après l'exécution de programme java dans eclipse.....	54
<b>Figure 5</b> : lancement d'exécution le fichier .jar dans Hadoop.....	54
<b>Figure 6</b> : le résultat de l'exécution de fichier .jar (MapReduce).....	56

Figure 7 : premier interface a l'ouverture d'Ubuntu.....58

# Introduction générale

Dans ces dernières années, le volume de données exploité dans des entreprises a considérablement augmenté. Emanant de source diverse (transaction, systèmes d'information automatisés, réseaux sociaux), elles sont souvent susceptibles de croître très rapidement. Lorsqu'on parle de manipulation de données volumineuses, on pense généralement à des problématiques sur la manipulation de ces données volumineuses et sur leur rapidité de traitement qu'on trouve généralement dans la vidéosurveillance avec ce nombre de déploiements de leur systèmes qui augmente régulièrement et constitués par de plus en plus de caméras, les opérations de vidéo surveillance ont des difficultés grandissantes d'exploitation dues aux grandes quantités de données vidéo stockées chaque jour.

Et pour cela on s'intéresse à des méthodes parallèles pour la recherche des visages et la détection des visages dans les vidéos de surveillance.

Pour cela les programmeurs ont fait une solution à ce problème qui nommée MapReduce, qui est un cadre à l'aide duquel nous pouvons écrire des applications pour traiter d'énormes quantités de données, en parallèle, sur de grandes grappes de matériel de base d'une manière fiable.

Pour traiter ces vidéos de surveillances il faut des frameworks qui manipulent ces grandes quantités de données telles que le framework Hadoop MapReduce qui est une infrastructure logicielle qui permet d'écrire des tâches traitant d'importantes quantités de données. Les données d'entrée sont divisées en blocs indépendants qui sont ensuite traitées en parallèle sur les nœuds d'un cluster. Le concept fondamental d'Hadoop réside dans la manipulation, le traitement et l'analyse de très grands jeux de données (qui se chiffrent en péta-octets), données qui sont alors automatiquement réparties dans ces espaces de stockage et des lots de traitement sur un ensemble de clusters de serveurs à bas coûts. D'un seul serveur à plusieurs milliers de machines, Hadoop est une solution évolutive qui embarque une capacité de tolérance aux pannes. Détection des défaillances et automatisation lui confèrent une excellente résistance. Derrière Hadoop se trouvent deux technologies importantes : MapReduce et le HDFS (le système de fichiers d'Hadoop).

Pour ce faire, et pour mener à bien notre travail nous avons opté pour la démarche articulée autour des chapitres suivants :

Dans le premier chapitre on a vu les systèmes de vidéos de surveillance et cette différente technologie ainsi que l'importance de l'indexation et la recherche dans vidéos.

Et dans le deuxième chapitre on a défini le Framework Hadoop et le principe de MapReduce et ces importances pour la manipulation des données volumineuses.

Dans le troisième chapitre on a présenté le système de détection de visage et les méthodes utilisées pour faire détecter les visages.

Finalement dans le quatrième chapitre: proposition d'algorithme scalable pour la détection de visage.

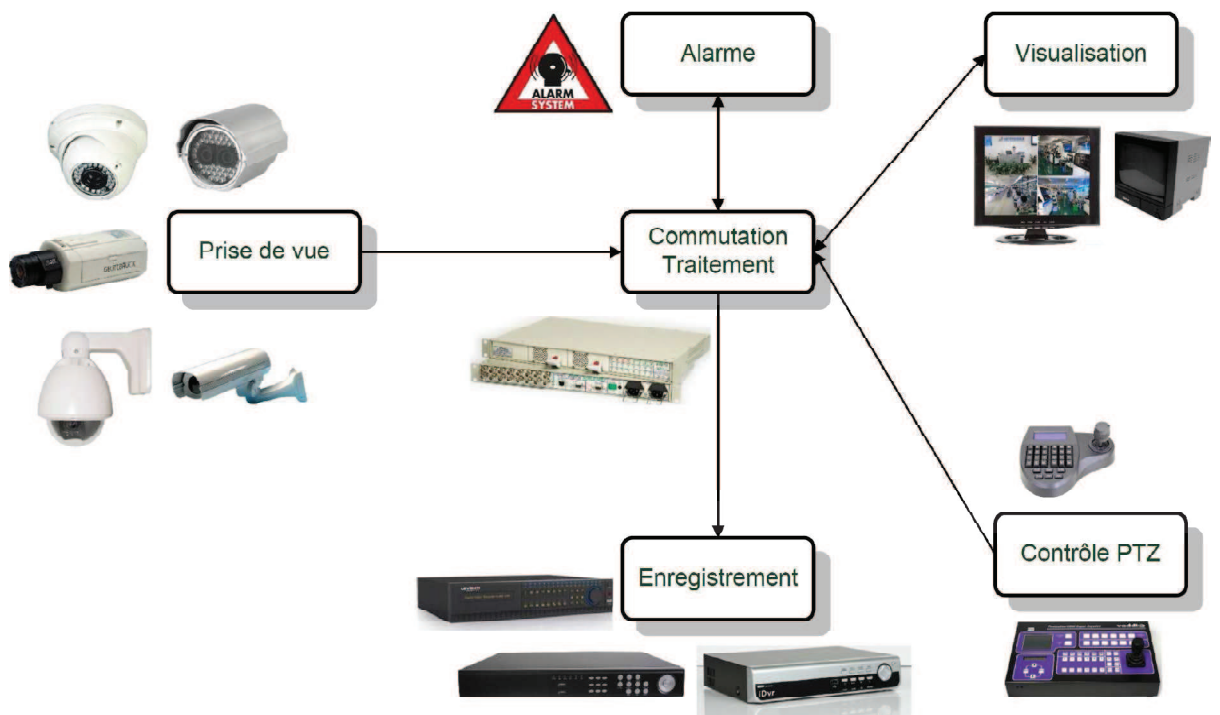
# Chapitre 1

## 1. Introduction :

De nos jours, la vidéosurveillance est omniprésente et on la retrouve dans de nombreux secteurs d'activité (banque, transport, industrie, grande distribution, etc.) ou lieux de vie (villes, immeuble de bureau, équipements collectifs, etc.). La plupart des responsables souhaitent accroître la sécurité en protégeant les biens et les personnes par la vidéosurveillance. Aujourd'hui ; le développement technologique permet d'obtenir de très bons résultats lorsque on sait exprimer un besoin et lui faire correspondre un matériel efficace et évolutif afin de pérenniser l'installation. Mais nous sommes confrontés à une pléthore de produits venant de tout horizon.

## 2. Système de vidéo de surveillance [1] :

La vidéo de surveillance, parfois désignée par le sigle anglais CCTV (Closed-Circuit TéléVision), est un système de caméras et de transmission d'image, disposé dans un espace public ou privé pour le surveiller. Les images obtenues avec ce système, peuvent être traitées automatiquement et/ou visionnées puis archivées ou détruites. La surveillance a pour but de contrôler les conditions de respect de la sécurité, de la sûreté ou de l'exécution d'une procédure particulière. Selon ses partisans, un tel système permet de prévenir la criminalité (vols, agressions, etc.) et d'opérer un contrôle social (mouvements de foule, etc.). Ses détracteurs lui reprochent son inefficacité, son coût et son atteinte à la vie privée.



**Figure1** : schéma d'un système de vidéo de surveillance.

Ce schéma représente les différentes fonctions d'un système de vidéo de surveillance. Celles-ci peuvent être assurées par un ou plusieurs appareils (un DVR peut remplir les fonctions de commutation et d'enregistrement).

Les différentes fonctions peuvent être décrites comme suit :

✚ **Prise de vue :** permet l'acquisition des images à l'aide de caméras plus ou moins perfectionnées. Certaines situations conduisent à utiliser des caméras avec des spécifications bien particulières (étanche, grand-angle, IR, etc.).



Figure2 : caméras de surveillance.

✚ **Commutation /traitement :** cette fonction permet le « dispatching » des signaux vidéos aux fonctions d'enregistrement et de visualisation. Elle permet aussi d'effectuer un traitement, à posteriori, de l'image (Luminosité/contraste, détection de mouvement, encombrement sortie de secours, etc.).



Figure 3 : Multiplexeur vidéo.

- ✚ **Enregistrement** : l'enregistrement des images se fait principalement de manière numérique. L'enregistrement 24 heures sur 24 sur une période glissante est limité à 30 jours par la loi.



Figure 4 : l'intérieur d'un enregistreur vidéo numérique.

- ✚ **Contrôle PTZ** : permet le contrôle de la position des caméras possédant la fonctionnalité PTZ (Pan-Tilt Zoom). Ce sont, généralement, des caméras de type dômes.



Figure 5 : pupitre de commande PTZ.

- ✚ **Visualisation** : un ou plusieurs moniteurs permettent la visualisation des images de vidéo de surveillance. Des installations, plus ou moins complexes, peuvent aussi permettre de diffuser le flux vidéo au sein même du lieu vidéo-surveillé pour une dissuasion des éventuels contrevenants.



**Figure 6 :** un centre du contrôle d'une ville.

✚ **Alarme :** les systèmes de vidéo de surveillance peuvent de communiquer avec des systèmes de détection d'intrusion. Une détection de mouvement sur un flux vidéo peut, par exemple, ouvrir un contrat exactement comme détecteur IRP (Infra Rouge Passif). L'échange d'informations va dans les 2 sens : les systèmes de vidéo de surveillance peuvent utiliser les signaux en provenance d'un détecteur pour déclencher un enregistrement.



**Figure 7 :** entrée/sortie d'un enregistreur vidéo numérique.

### 3. Technologie :

#### 3.1. Architecture des systèmes de vidéosurveillance [2] :

##### 3.1.1. Composantes d'un système de vidéo surveillance

Cette section présente de façon sommaire, les différentes composantes matériel et logicielles des systèmes de vidéo de surveillance. Une description plus détaillée de l'infrastructure des systèmes de vidéo-surveillance, ainsi que des principes guidant les choix matériels, peut être retrouvé dans de nombreux ouvrages et guide de références.

##### 3.1.1.1. Acquisition

Il existe une panoplie de modèle de cameras répondant a différents besoin de surveillance. Elles sont analogique ou numérique et peuvent être motorisée ou non. Plus spécifiquement, on retrouve les type de cameras suivants :

✚ **Fixe** : pointée dans une direction unique, elle couvre une zone définie (une entrée, une portion de stationnement, etc....). c'est la caméra de surveillance traditionnelle. Elle constitue un excellent choix lorsqu'on désire que la présence de la caméra, ainsi que ça direction de surveillance, soient visible.

✚ **PTZ (Pan-Tilt-Zoom)** : Motorisée, elle peut être actionnée, manuellement ou automatiquement on des mouvements panoramique/inclinaison/zoom. Elle sert a suivre des objets ou des individus ce déplaçant dans la scène ou à zoomer sur des régions d'intérêt (par exemple, sur une plaque d'immatriculation).

✚ **Dôme** : recouverte d'un caisson hémisphérique, ce qui la rend discrète et, dans certaine modèle résistant au vandalisme et aux intempéries. Elle peut être fixe ou mobile. Les versions motorisées, couvrent une zone très large, grâce à leur balayage horizontale de 360° et de 180° à la verticale. Bien qu'en « tour de garde », elle puisse remplacer dix cameras fixes en balayant l'aire à surveiller, elle n'observe qu'une seul direction à la fois.

✚ **Mégapixel** : offre une résolution plus élevée que les cameras standards, allant de 1 à 16 mégapixel. Elle permet soit de capter une image plus détaillée, soit de couvrir un plus large champ visuel réduisant le nombre de caméras nécessaire pour couvrir une aire à surveiller. Lorsqu'utilisée avec un grand angle, elle possède un espace de visualisation allant généralement de 140° à 360°. Offrant la possibilité de zoomer de façon logicielle dans l'image, elle peut ainsi devenir une alternative à la caméra PTZ mécanique qui entraîne l'usure des pièces. Sa résolution élevée contribue à l'amélioration de la performance des algorithmes de détection et de reconnaissance exigeant un haut niveau de détails, telles que la lecture de plaques d'immatriculation et la reconnaissance de visage.

✚ **Infrarouge et thermique** : sensible au rayonnement infrarouge (IR), elle est capable de produire une image de bonne qualité dans le noir pour une surveillance nocturne. De nuit, elle filme en noir et blanc, mais elle peut produire une image couleur le jour. Certaines caméras infrarouges sont équipées de leur propre source de lumière IR, allumée lorsque le niveau d'éclairage chute sous un certain seuil. Des projecteurs IR séparés (lampe ou LED) peuvent aussi être utilisés. Les caméras thermiques enregistrent le rayonnement de chaleur des objets. Elles ne requièrent aucune source d'illumination.

✚ **Panoramique** : grâce à une optique spéciale, elle offre 360° de visibilité avec une seule caméra. Elle permet un PTZ virtuel dans l'image. Les principales technologies panoramiques pour la surveillance sont le fisheye, la lentille à miroirs et la lentille panomorphe. Toutefois, la résolution de ces caméras est souvent insuffisante pour des analyses nécessitant un niveau de détail élevé.

✚ **Transmission** : la vidéo captée par les caméras de surveillance doit être transmise aux systèmes d'enregistrement, de traitement et de visionnement. Cette transmission peut se faire par câble (câbles coaxiaux ou à fibre optique, fils de cuivre torsadés) ou à travers l'air (signaux infrarouges, transmission radioélectrique).

La vidéo filaire prédomine largement dans les systèmes de vidéo de surveillance. Elle offre une puis grande bande passante et une meilleure fiabilité que les connections sans fil, à un coût inférieur. Cependant, la transmission vidéo sans fil s'impose parfois comme solution, par exemple dans le cas de surveillance de grands périmètre où l'installation de câblage

s'avérerait trop coûteuse, ou lorsque les zones à surveiller sont impossibles à rejoindre par câble. Qu'il transite par fil ou sans fil, le signal vidéo peut être analogique ou numérique. Néanmoins, les réseaux informatiques (LAN, WAN ou internet) sont plus en plus utilisés pour transporter la vidéo grâce au protocole IP. Les caméras IP peuvent connecter directement sur ces réseaux, tandis que les flux vidéo émergeant de caméras analogiques doivent, au préalable, être numérisés par un encodeur, aussi appelé serveur vidéo, pour passer par les réseaux IP.

✚ **Compression :** la vidéo numérisée représente une grande quantité de données à transmettre et à archiver. L'envoi d'une séquence vidéo peut nécessiter jusqu'à 165 mégabits de bande passante et la vidéo d'une seule caméra pour une journée peut occuper sept Giga-octets d'espace disque. C'est pourquoi la vidéo de surveillance doit être compressée grâce à des codecs ( ), algorithmes permettant de réduire la quantité de données en supprimant les redondances, par image ou entre les trames d'une séquence, ainsi que les détails imperceptibles à l'œil humain. Selon le type de compression, l'usage du processeur requis pour l'exécution du codec est plus ou moins intensif. Un compromis s'impose donc entre le taux de compression et les ressources du processeur qui sont accaparées.

Il existe deux grands groupes de standards internationaux de compression : JPEG, créés par le joint Photographic Experts Group, et MPEG, élaborés par le MovingPhotographic Experts Group. Dans le premier groupe, on trouve les formats JPEG pour les images fixes, et MJPEG pour les séquences vidéo. Le second groupe comprend les formats MPEG-1, MPEG-2, MPEG-4 et H.264. À l'heure actuelle, MJPEG et MPEG-4 sont les standards les plus répandus en vidéo de surveillance. Toutefois, avec les améliorations en qualité et efficacité (taux de compression, latence, résistance à l'erreur) qu'il apporte, H.264 devrait bientôt remplacer MPEG-4. En effet, sans affecter la qualité de l'image, l'encodeur H.264 permet de réduire la taille de celle-ci de plus de 80% par rapport à la compression MJPEG, et de 50% par rapport à la compression MPEG-4.

### 3.1.1.2. Traitement :

Les systèmes de gestion vidéo opèrent les traitements des images de vidéo de surveillance, tel que la gestion des différents flux vidéo, le visionnement, l'enregistrement, l'analyse et la recherche dans les séquences enregistrées. Il existe quatre grandes catégories de **systèmes** de gestion vidéo.

✚ **Enregistreur vidéo numérique (DVR) :** appareil qui dispose d'un disque dur interne pour l'enregistrement numérique de la vidéo et d'un logiciel intégré de traitement de la vidéo. Il n'accepte que les flux provenant de caméras analogiques, qu'il numérise. Les modèles récents permettent de visionner la vidéo à distance sur ordinateur. Encore très répandus, ils laissent toutefois peu à peu leur place aux systèmes supportant la vidéo IP de bout en bout.

✚ **Enregistreur vidéo hybride (HDVR) :** similaire à l'enregistreur numérique, mais accepte à la fois le branchement de caméras analogique et IP. Il est possible de rendre hybride plusieurs modèles d'enregistreurs vidéo numériques par l'ajout d'un logiciel.

✚ **Enregistreur numérique réseau (NVR) :** conçu pour les architectures réseaux IP de vidéo de surveillance, il ne peut traiter que les signaux vidéo provenant de caméras IP ou d'encodeurs.

✚ **Logiciel de vidéo de surveillance IP :** solution purement logicielle de gestion de la vidéo sur un réseau IP. Dans le cas de système de surveillance comportant peu de caméras, un navigateur Web peut suffire à gérer la vidéo. Pour de plus gros réseaux de vidéo de surveillance, un logiciel dédié de gestion vidéo doit être utilisé. Celui-ci s'installe sur un ordinateur personnel ou un serveur. Plus complexe à installer, en raison des configurations nécessaires du serveur, il offre une plus grande flexibilité pour le choix et l'ajout de composants au réseau de vidéo de surveillance. Les logiciels de vidéo de surveillance IP représentent une tendance forte en gestion vidéo, surtout dans les infrastructures comportant un grand nombre de caméras. Les plateformes ouvertes permettent d'intégrer facilement des caméras et composants matérielles de différents fabricants.

### 3.1.1.3. archivage :

La période d'archivage des séquences vidéo varie selon les besoins de surveillance, allant de quelques jours à quelques années. En moyenne, les organisations conservent les preuves vidéo entre 30 et 90 jours. Le déploiement de vastes réseaux de caméras et l'utilisation de vidéo de surveillance à haute résolution fait exploser les demandes pour les systèmes de stockage. Bien que le coût des supports d'enregistrement ait considérablement baissé dans les dernières années, l'archivage représente souvent une part importante des dépenses d'infrastructure en vidéo de surveillance, en raison de la quantité toujours croissante de données vidéo à stocker.

Les solutions de stockage sont de deux types :

1. **Interne :** les disques durs intégrés aux enregistreurs vidéo numériques ou aux serveurs représentent la forme d'archivage la plus répandue. Elle peut offrir jusqu'à quatre téraoctets d'espace. Certaines caméras IP disposent même d'une carte mémoire ou d'un disque USB permettant d'enregistrer des heures, voire des jours de vidéo. Les solutions internes d'archivage conviennent bien pour les systèmes de vidéo de surveillance de taille modeste, comprenant jusqu'à 50 caméras.

2. **Rattaché :** l'archivage se fait sur des appareils externes aux enregistreurs ou serveur vidéo. De type NAS (Network Attached Storage) ou SAN (Storage Area Network), ces systèmes offrent un espace de stockage partagé entre les différents clients du réseau. Sur un système de stockage en réseau NAS, un fichier est archivé sur un même disque dur, alors qu'avec le réseau de stockage SAN, un fichier peut être sauvegardé en fragments répartis sur plusieurs supports de stockage. Ces solutions d'archivage rattachées sont particulièrement avantageuses pour les grands réseaux de vidéo de surveillance comportant un grand nombre de caméras. Bien que plus onéreuses que les systèmes internes d'archivage, ces solutions rattachées sont supérieures en termes d'extensibilité, de flexibilité et de redondance.

### 3.1.1.4. affichage :

Une grande partie de la vidéo captée par la caméra de surveillance n'est jamais regardée. Elle est simplement archivée, au cas où un visionnement soit nécessaire suite à un incident. Traditionnellement, la vidéo de surveillance a principalement servi comme

outil d'enquête. Toutefois, dans plusieurs cas de surveillance, des agents de sécurité visionnent, en temps réel, les images provenant des caméras de surveillance. Sans nécessairement regarder toute la vidéo captée, les agents peuvent faire une revue périodique des différentes sources vidéo.

La vidéo de surveillance peut être visionnée sur différents appareils. Dans de petites installations, il est possible de regarder la vidéo directement de l'enregistreur, simultanément à son enregistrement. Plus généralement, les images seront regardées à distance, sur un ordinateur ou, de façon mobile, sur un téléphone ou dispositif portable. Les grands centres d'opérations de sécurité, supervisant des centaines de caméras, utilisent souvent un mur d'écrans vidéo. Ceux-ci offrent une grande surface de visionnement et permettent d'afficher différents flux vidéo.

### 3.1.2. évolution des systèmes de vidéo de surveillance :

La transition numérique en vidéo de surveillance s'est opérée en plusieurs étapes. Amorcée avec l'apparition de l'enregistreur numérique, elle se poursuit vers une conversion totale à l'infrastructure IP, où la vidéo est transmise sur réseau intranet ou Internet de la caméra à l'écran de visionnement. Dans ce passage, l'on retrouve plusieurs systèmes hybrides, intégrant composantes analogiques et numériques.

#### 3.1.2.1. première génération : le tout analogique

Dans le réseau TVCF analogique traditionnel, des caméras analogiques sont connectées par câbles coaxiaux (un câble par caméra) aux écrans de surveillance et, pour des fins d'archivage, à un magnétoscope qui enregistre la vidéo sur cassette. Un multiplexeur peut être utilisé pour grouper les flux vidéo de plusieurs caméras en un seul signal composite qui est transmis au magnétoscope ou à un moniteur analogique. Il permet d'afficher quatre, neuf ou 16 signaux vidéo sur un même écran, ou d'enregistrer ceux-ci sur un même système d'archivage. Aucune compression n'est effectuée sur les signaux vidéo. Avec un magnétoscope conçu à cet effet, l'enregistrement à taux de trame (frame rate) réduit permet d'économiser de l'espace sur la bande vidéo, selon les besoins de surveillance.



#### **Avantage :**

- les systèmes analogiques sont très fiables.
- simple à utiliser, ils ne requièrent pas de compétences informatiques.



#### **Inconvénient :**

- la qualité de la vidéo est inférieure à celle des systèmes numériques.
- il faut changer les cassettes fréquemment (aux trois jours ou plus).
- nécessite un nettoyage et un entretien régulier des magnétoscopes.
- la qualité de la vidéo enregistrée se détériore avec le temps.
- Ne permet pas le visionnement à distance, comme sur les réseaux numériques.

#### 3.1.2.2. deuxième génération : le système hybride

Le remplacement du magnétoscope à cassette par un enregistreur numérique (DVR) représente la première étape de la transition numérique en vidéo de surveillance. Apparus dans les années 90, les enregistreurs numériques archivent la vidéo sur des disques durs.

Comportant souvent plusieurs entrées vidéo, l'enregistreur numérique remplace à la fois le multiplexeur et le magnétoscope analogique.

Les modèles récents sont équipés d'un port Ethernet, permettant la connexion à un réseau et l'accès à distance à la vidéo, soit en temps réel, soit à partir de l'enregistrement. Les transmissions vidéo se font sous protocole IP à partir de l'enregistreur numérique. Ils permettent la transition vers un système hybride de vidéo de surveillance sur réseau IP, tout en conservant les caméras analogiques. L'enregistreur numérique hybride (HDVR), quant à lui, peut recevoir à la fois les flux vidéo de caméras analogique et IP. Il représente une solution efficace pour moderniser un système de vidéo de surveillance en profitant des avantages des nouvelles caméras IP, tout en conservant les caméras analogiques existantes en place. Alors qu'on estime que 95% des caméras sont encore analogiques, les enregistreurs numériques et les enregistreurs hybrides sont des technologies très répandues.

#### **Avantage :**

- aucune cassette à changer.
- la vidéo archivée est de meilleure qualité, sans détérioration avec le temps.
- possibilité de chercher rapidement dans les enregistrements vidéo.
- surveillance vidéo et opération du système à distance à partir d'un PC.

#### **Inconvénient :**

- concentration des tâches de numérisation, compression vidéo, enregistrement et réseautage dans la même machine.
- l'enregistrement numérique est un appareil propriétaire, ce qui augmente les coûts de maintenance et de mise à jour.
- le nombre d'entrées vidéo de l'enregistreur numérique (souvent un multiplexeur de 16) contraint l'ajout de caméras.

Les encodeurs vidéo, aussi appelés serveurs vidéo, sont utilisés pour convertir les signaux provenant de caméras analogiques et les transmettre en flux IP sur un réseau via un commutateur. Tout en conservant les caméras analogiques, ils permettent un passage presque complet à l'infrastructure réseau pour la vidéo de surveillance, puisque que la vidéo est constamment transmise sous le protocole IP à travers le réseau. Les encodeurs vidéo peuvent être utilisés avec les enregistreurs vidéo réseaux (NVR). Ceux-ci ne peuvent traiter et enregistrer que des flux vidéo IP. Ils sont offerts sous une plateforme ouverte (un ordinateur muni d'un logiciel de gestion vidéo) ou dans un matériel propriétaire dédié. Sous cette dernière forme, l'enregistreur vidéo réseau se compare à un enregistreur numérique hybride, excepté qu'il nécessite l'usage d'encodeurs pour opérer avec des caméras analogiques.

#### **Avantage :**

- Utilisation d'ordinateurs et de matériel standards de réseau pour l'enregistrement et la gestion de la vidéo.
- possibilité d'enregistrer à l'extérieur du site de surveillance (par exemple, centralisation des enregistrements).
- architecture distribuée qui offre flexibilité, extensibilité (peut ajouter une caméra à la fois) et redondance (en cas de bris ou pannes).

**Inconvénient :**

- gourmand en bande passante, si l'enregistrement est fait hors du site de surveillance (par exemple, de façon centralisée).
- si le réseau tombe en panne, l'enregistrement peut être interrompu.
- si l'enregistrement centralisé n'est pas requis, l'utilisation d'enregistreur vidéo réseau est souvent plus coûteuse que celle d'enregistreurs numériques.
- nécessitent des calculs complexes pour déterminer le nombre de flux vidéo pouvant être supportés par le serveur, la quantité d'espace disque nécessaire à l'enregistrement, le taux de trames, le niveau de compression et d'autres facteurs liés aux capacités du réseau.

**3.1.2.3. troisième génération : le tout numérique IP :**

Au sens strict, un système de vidéo de surveillance est complètement IP, lorsque toutes ses composantes sont numériques et toutes les transmissions sont effectuées sous le protocole IP. Ces réseaux comprennent donc des caméras réseau, aussi appelées IP. Il s'agit de caméras intégrant leur propre encodeur. Celles-ci sont reliées, via des commutateurs réseau, à des serveurs (ordinateurs personnels), munis d'un logiciel de gestion vidéo. L'enregistrement est fait sur un serveur ou sur des enregistreurs vidéo réseaux propriétaires. Les traitements sont effectués sur le serveur ou dans les périphériques. Toutefois, beaucoup de gens considèrent qu'un système de vidéo de surveillance dont la vidéo est transmis sous protocole IP à partir des encodeurs, constitue un système réseau IP. Dans ces systèmes, les caméras peuvent être analogiques, en autant qu'elles soient reliées à des encodeurs. Plus de détails sur la vidéo de surveillance IP et, notamment ses avantages et inconvénients, sont fournis dans la section suivante.

**3.1.3. vidéo de surveillance IP [2] :**

Depuis deux ans, le marché de la vidéo de surveillance migre de la vidéo analogique à la vidéo IP. Cet essor de la vidéo de surveillance réseau est favorisé par le perfectionnement des processeurs, la baisse des coûts de stockage et l'amélioration des algorithmes de compression.

Bien que cette transition s'observe, elle ne fait que débiter et il est difficile de prédire à quel rythme le « tout IP » prendra le dessus sur les technologies analogiques en vidéosurveillance. Les caméras constituent le facteur de résistance. En effet, il est estimé que 95% des 40 millions de caméras installées à travers le monde sont encore analogiques. De plus, encore aujourd'hui, seulement un acheteur sur quatre choisit des caméras IP. Pourtant, les caméras IP offrent des avantages techniques sur les caméras analogiques, tels qu'une meilleure qualité d'image, une résolution plus élevée et de l'intelligence embarquée. Le coût élevé des caméras IP, environ le double des caméras analogiques, reste le principal frein à l'achat. Avec les améliorations apportées par les manufacturiers aux enregistreurs vidéo et l'existence d'enregistreurs vidéo hybrides, le règne des caméras analogiques pourrait se prolonger encore quelques années.

Les réseaux IP de vidéosurveillance de dernière génération présentent les caractéristiques suivantes :

- Ils sont numériques de A à Z : caméras, réseaux, enregistrement, accès.
- Ils peuvent inclure différents types de caméras : intelligente, mégapixel, sans fil, PTZ, panoramiques, etc.
- Ils utilisent du matériel non propriétaire.
- Ils fonctionnent avec des serveurs distribués et multiplateformes.

- La sauvegarde se fait sur réseau (NVR).
- On peut y accéder à distance, n'importe où, n'importe quand, soit d'un centre de contrôle, via Internet ou un réseau LAN ou WAN, sur un cellulaire ou un assistant numérique personnel, etc.
- Ils peuvent inclure de l'analytique vidéo.

La vidéosurveillance sur réseau IP offre de nombreux avantages. Elle repose sur une infrastructure plus souple que la vidéo analogique, combinant transmission câblée et sans fil. E plus, l'infrastructure réseau est rapide et facilement extensible. Il n'y a pas de limite au nombre de caméras qui peuvent s'y ajouter. Considérant que les systèmes actuels peuvent facilement comporter entre 400 et 500 caméras, comparativement à 25 à 30 il y a quatre ans, cette facilité d'étendre le réseau de vidéosurveillance est très attrayante lorsqu'on pense déployer un nouveau système.

Pour les installations étendues comportant de grandes distances entre les caméras et les systèmes d'enregistrement, les caméras IP deviennent un choix avantageux, car le coût du câble coaxial est très élevé. C'est pourquoi les caméras IP sont surtout déployées pour la surveillance d'institutions, telles que les établissements scolaires, les campus de grandes entreprises, les municipalités et les bases militaires.

Les caméras IP apportent de nombreux bénéfices techniques. Par exemple, les caméras mégapixel IP offrent des avantages indéniables au niveau de la vidéosurveillance. Ces caméras, présentant une résolution nettement supérieure à celle des caméras analogiques, ont le potentiel de résoudre plus de crimes. Bien que son coût soit élevé, une seule caméra mégapixel peut souvent remplacer plusieurs caméras. Les caméras intelligentes, intégrant des processeurs, apportent des gains techniques aux systèmes de vidéosurveillance de nouvelle génération. Elles permettent de distribuer les calculs pour les traitements d'analytique vidéo et, ainsi, de ménager la bande passante en ne transmettant que les données pertinentes pour la sécurité.

L'architecture ouverte des réseaux de vidéosurveillance IP permet de combiner le matériel de différents fabricants. Il est possible de sélectionner les composants les plus adaptés à son application de surveillance. De plus, l'architecture IP facilite l'intégration de différents systèmes de sécurité (vidéo, alarmes de feux, contrôle d'accès, etc.).

La configuration de ces réseaux avec redondance, tant au niveau de la transmission que de l'archivage, assure une meilleure fiabilité. Dans un réseau entièrement analogique, le mauvais fonctionnement d'une caméra ou d'un enregistreur peut signifier une perte définitive de vidéo. De plus, l'archivage sur réseau, dont les coûts ne cessent de baisser, permet de conserver des années de vidéo plutôt que quelques jours ou semaines avec magnétoscopes ou enregistreurs numériques. Il suffit d'ajouter des disques durs pour augmenter la capacité de stockage.

Dans ces réseaux, la transmission numérique de la vidéo, de bout en bout du système, réduit les pertes de qualité qu'occasionnent les conversions d'analogique à numérique. De plus, par des traitements logiciels intelligents, il est possible de filtrer les événements pertinents dans la vidéo et de ne transmettre que les métadonnées les décrivant. Ceci permet d'économiser de la bande passante. Ces métadonnées servent ensuite à indexer le contenu vidéo pour permettre des recherches plus performantes dans

les séquences captées. Il est aussi possible, dans ces réseaux, de transmettre des commandes aux caméras, en fonction de la vidéo traitée. Par exemple, en réponse à un événement suspect, le logiciel de gestion vidéo pourra activer automatiquement le zoom optique ou numérique sur une caméra du réseau pour un suivi plus précis de la situation.

Le passage à la vidéosurveillance IP comporte tout de même certains inconvénients. Par exemple, dans le cas d'infrastructures analogique existantes, refaire tout le filtrage réseau peut représenter un coût élevé. Dans ce cas, l'utilisation d'encodeurs pour numériser les signaux analogiques permet de conserver les caméras en place et d'implanter un réseau hybride. De plus, la technologie des enregistreurs numériques a beaucoup progressé, afin de rester compétitive. Les enregistreurs numériques offrent désormais plusieurs avantages comparables aux technologies sur réseaux IP : accès à distance, capacité de gérer un grand nombre de caméras, intégration avec les autres systèmes (contrôle d'accès, détection d'intrusion, systèmes de points de vente, etc.), capacité analytique. Pour de petites installations de vidéosurveillance, l'utilisation de caméras analogiques avec enregistreurs numériques offre donc un excellent rendement, à des coûts moindres que la vidéo IP.

Bien que sur le plan technologique, les réseaux IP de vidéosurveillance offrent de nombreux avantages, le déploiement de ceux-ci pose certains problèmes organisationnels. Le premier réside dans l'installation. La plupart des intégrateurs et installateurs en vidéosurveillance sont issus du secteur de la sécurité physique. Or, l'installation de systèmes de vidéosurveillance sur réseau IP requiert des compétences en technologie de l'information, particulièrement en réseautique. Encore peu d'intégrateurs et installateurs de vidéosurveillance sont spécialisés dans ce domaine.

Le second problème touche à la gestion de ces systèmes. Une fois installés, qui, dans l'organisation, est responsable du système de vidéosurveillance : le personnel de sécurité physique ou les services informatiques ? Le flou qui existe encore sur cette question dans les organisations retarde bien souvent le passage à la vidéosurveillance IP.

Finalement, passer d'un réseau de télévision en circuit fermé à un système de vidéosurveillance, dont les données transitent par les réseaux Ethernet ou Internet, soulève les questions de sécurité informatique. Evidemment, ceci nécessite de mettre en place les mécanismes de sécurité adéquats pour protéger la vidéo de surveillance. Aussi, le bon fonctionnement de la vidéosurveillance devient tributaire de la fiabilité du réseau. A cet égard, l'intégration de la vidéosurveillance sur réseaux IP suivra peut-être celle de la téléphonie IP, qui est maintenant bien répandue.

### **3.1.4. architecture d'un réseau de vidéosurveillance intelligente [2] :**

Les systèmes de vidéosurveillance intelligente peuvent être déployés selon deux grands types d'architecture, soit centralisée ou distribuée. L'une comme l'autre opèrent des traitements intelligents pour extraire des données sur les images vidéo et peuvent, au besoin, commander le déplacement d'une caméra PTZ pour opérer une surveillance active.

#### **3.1.4.1. architecture centralisée :**

Dans une architecture centralisée, tous les traitements intelligents sont concentrés en un même endroit. Dans les systèmes traditionnels, c'est un serveur informatique (un

PC) qui opérera les traitements analytiques. L'analyse vidéo nécessite une grande puissance de calcul qui monopolise une part importante des ressources du processeur. Comme ils doivent aussi gérer l'encodage, l'enregistrement et le visionnement des flux vidéo, les enregistreurs numériques et serveurs ne peuvent accomplir des tâches d'analytique vidéo que sur un nombre restreint de caméras.

De plus, dans le cas des systèmes de vidéosurveillance sur réseau, la transmission de tous les flux vidéo en un point centralisé consomme beaucoup de bande passante. Le réseau informatique doit pouvoir soutenir ce trafic.

#### **3.1.4.2. architecture distribuée :**

Comme son nom l'indique, l'architecture distribuée répartit les traitements intelligents dans les différents nœuds du système de vidéosurveillance. Ainsi, les calculs nécessaires à l'analytique pourront être faits en périphérie, sur des caméras intelligentes dotées de processeurs, les encodeurs ou dans les commutateurs réseau. Dans cette architecture, plutôt que d'envoyer des flux vidéo à l'enregistreur numérique ou à un ordinateur central, seules des métadonnées extraites sur la vidéo sont transmises. L'intelligence peut aussi être partagée entre les périphériques et l'unité centrale de traitement. Cette infrastructure aide à diminuer les coûts de câblage et la bande passante requise dans un réseau de vidéosurveillance. De plus, elle offre une meilleure extensibilité, puisque l'ajout de caméras n'est pas nécessairement limité par la puissance de calcul de l'enregistreur numérique ou du serveur.

#### **4. Domaines d'application [2] :**

A son origine, la vidéo de surveillance a d'abord été utilisée par les services publics (police, transport, administrations). Elle fut ensuite adoptée par des entreprises désirant protéger des actifs stratégiques, telles que les raffineries, les centrales nucléaires, les barrages fluviaux, les usines agroalimentaires et les complexes pharmaceutiques. De nos jours, les caméras de surveillance sont présentes dans divers lieux publics et privés : immeubles, commerces, stationnements, gares, aéroports, routes, transport publics, banques, etc....

Les systèmes de vidéo de surveillance sont déployés à différents échelles. Pour la surveillance de crimes mineurs (par exemple, attaque, vols), la vidéo de surveillance est principalement utilisée pour l'enquête après le fait. Ce niveau de surveillance requiert des technologies simples, le plus souvent analogiques, qui n'intègrent pas d'intelligence vidéo. C'est le type de système qu'on peut retrouver au dépanneur du coin, dans un petit commerce ou dans une résidence privée.

La surveillance d'immeubles ou de commerces de plus grande taille nécessite souvent un réseau plus étendu de caméras. La surveillance vise principalement le contrôle des voies d'accès, des stationnements et s'il s'agit de magasins, des rayons et points de vente. La vidéo de surveillance y est surtout utilisée pour les fins d'enquête. Cependant, ces usagers souhaiteraient obtenir des systèmes de surveillance capables de générer des alertes en temps réel pour permettre une intervention immédiate. Les produits







d'analytiques vidéo accomplissent donc une percée dans ces secteurs. Le retour sur l'investissement constitue un facteur déterminant pour l'achat d'équipements de vidéo surveillance intelligente pour cette clientèle.

La vidéo de surveillance à grande échelle se retrouve dans les villes et les quartiers, les systèmes de transports, les campus académiques, les grands événements (festivals, jeux olympiques, etc.), les périmètres de sécurité étendus. Elle nécessite le déploiement de plusieurs dizaines, voire de centaines de caméras. Celles-ci doivent parfois être accessibles à des centaines d'intervenants en sécurité, provenant de différentes agences gouvernementales, corps policier ou services d'urgence. Dans ces installations, la vidéo de surveillance s'ajoute à une panoplie de systèmes de sécurité et de contrôle : contrôle d'accès, d'incendie, téléphonie, communication radio, systèmes géomatiques, etc. Etant donné le nombre de caméras vidéo impliquées et l'importance des interventions d'urgence, ces applications sont particulièrement propices à l'utilisation d'analytique vidéo pour le traitement automatisé des flux vidéo générant des alarmes lors d'événements suspects. Comme les budgets alloués aux systèmes de surveillance dans ces infrastructures sont souvent importants, l'ajout de logiciels d'analytique peut plus facilement y être envisagé.

Il faut noter une utilisation assez récente de la vidéo de surveillance dans des unités mobiles : voitures de patrouille, ambulances, autobus, etc. la transmission sans fil des signaux vidéo à un centre de surveillance peut poser problème.

#### 4.1. Gouvernement et sécurité publique :

Les différents paliers de gouvernement doivent assurer la sécurité de la population, ainsi que des infrastructures publiques. Au niveau national, elle sera utilisée, par exemple, pour surveiller les éléments suivants :

-  Infrastructures sensibles
-  Frontières
-  Edifices et sites gouvernementaux
-  Laboratoires
-  Bases militaires
-  Prisons

A un niveau local, la vidéo de surveillance est implantée dans plusieurs villes dans le monde pour surveiller la criminalité et la servir d'outil pour les interventions d'urgence. Elle contribue aussi à assurer la sécurité lors de grands rassemblement (spectacles, manifestation, événement sportifs). Londres est la ville la plus souvent citée pour le nombre de caméras déployées dans ces rues. La vidéo de surveillance sert aussi à gérer le stationnement, notamment pour le contrôle de permis de stationnement, l'application des règles, la détection des vols, et le contrôle d'accès.

La vidéo de surveillance est abondamment utilisée par les forces de l'ordre pour mener les enquêtes, surveiller les personnes et les véhicules recherchés, détecter les activités dangereuses ou criminelles. On la retrouve à bord des auto-patrouilles pour la vérification des interventions policières. On retrouve de plus en plus de caméras le long des tronçons routiers pour surveiller la circulation et détecter les incidents, les comportements dangereux ou les infractions.

Le secteur gouvernemental présente plusieurs besoins pour la vidéo de surveillance intelligente : identification d'individus et de véhicules, comptage de personnes et surveillance de foules, reconnaissance de comportement suspects ou violents (bagarres, méfaits), détection d'intrusions, monitoring routier. Des analyses plus poussées, tel que la reconnaissance d'émotion par ordinateur ou du mensonge chez un individu, sont aussi envisagée.

#### **4.2. Education :**

La vidéo de surveillance est de plus en plus présente dans les établissements scolaires. Son utilisation vise à assurer la sécurité des enseignants et étudiants, ainsi que la protection des actifs contre le vandalisme et le vol.

#### **4.3. Commerce de détail :**

Le commerce de détail représente un marché en expansion pour la vidéo de surveillance. Celle-ci y est utilisée autant pour des aspects de sécurité intérieure (magasin, entrepôt) qu'extérieure (stationnement). Même les plus petits commerce se dotent de caméras pour minimalement, conserver des preuves vidéo en cas de vols ou d'incidents. Dans les chaînes de magasins, des systèmes de vidéo de surveillance beaucoup plus sophistiqués sont déployés pour permettre la surveillance centralisée des différentes succursales. Dans l'ensemble du secteur, la vidéo surveillance visera notamment :

- ✚ La surveillance des caisses et des transactions (vols, fraudes par des employés).
- ✚ La protection des biens matériels et des infrastructures.
- ✚ La surveillance de l'inventaire, des marchandises (livraison).
- ✚ La protection du personnel et des clients.
- ✚ Le contrôle d'accès aux aires verrouillées.
- ✚ La vérification des situations d'urgences (incendie, alarmes, etc.).
- ✚ La surveillance des stationnements, des véhicules des entrées et sorties.

#### **4.4. Transports :**

La sécurité et le bon fonctionnement des aéroports, des gares, des ports et des transports publics est critique pour l'économie d'un pays. Un incident de sécurité peut gravement perturber leurs opérations et se traduire en pertes significatives. Or, étant donné les flots importants de passagers qui y transitent et l'étendue de leurs

infrastructures, les systèmes de transports font face à des défis de sécurité hors du commun. Les actes terroristes commis dans différents systèmes de transports dans le monde ont exacerbé ces défis. Les logiciels d'analytique ciblent le secteur des transports en offrant différentes fonctions adaptées : détection d'intrusion dans un périmètre ou une zone contrôlée, détection de personnes pénétrant dans une voie de sortie, détection de bagage abandonnée, reconnaissance de visages, comptage de personnes, reconnaissance de plaques d'immatriculation pour le contrôle d'accès aux stationnements, détection de comportements suspects (rôdeurs, vandalisme, graffiti), détection des personnes sur les voies.

#### **4.5. Milieu bancaire :**

La vidéo de surveillance est très utilisée pour la sécurité des banques. Dans un premier temps, la présence de caméras opère un certain pouvoir de dissuasion sur la perpétration de vols à mains armées et d'agressions. Si un tel délit survient, les séquences archivées servent à l'enquête et l'identification des malfaiteurs.

Les guichets automatiques sont particulièrement à risque pour les actes criminels. Les caméras de surveillance aident à détecter les fraudes, comme par exemple, l'installation d'un dispositif pour lire l'information magnétique des cartes de guichet.

Pour le milieu bancaire, la vidéo de surveillance intelligente peut augmenter l'efficacité de la surveillance. Elle permet d'assurer un suivi sur l'ensemble du réseau des succursales, afin d'y détecter les individus ou comportements suspects. Elle permettra, entre autres, de retrouver toutes les séquences vidéo provenant de toutes les succursales où apparaît un certain individu, grâce à des techniques de reconnaissance faciales.

#### **4.6. Autres :**

La surveillance de vidéo trouve aussi sa place en santé pour assister les interventions. On peut même la retrouver à bord des ambulances, en combinaison avec d'autres instruments de mesure, pour permettre de suivi de patient à distance. L'entreprise Américaine Indigo Vision a développé des solutions d'analytiques embarquées dans des caméras intelligentes spécifiquement pour détecter les mouvements des hélicoptères qui atterrissent sur le site d'un hôpital Américain.

Le marché grand public se développe aussi. De plus en plus de système de vidéo de surveillance pour les résidences sont en vente, même dans les magasins non spécialisés. Les caméras complètent maintenant les différents dispositifs de surveillance et de sécurité pour le secteur résidentiel.

### **5. Indexation et recherche des vidéo surveillances [3] :**

L'indexation et la recherche de vidéo de vidéosurveillance trouve sa place dans des applications différentes avec plusieurs types d'utilisateurs. Nous décrivons ici deux grands types d'applications. La première application est la sécurité. Dans un système de vidéosurveillance, une alarme est déclenchée si le système détecte un événement intéressant. Habituellement, le personnel de sécurité veut trouver des informations antérieures concernant le ou les objets impliqués dans cet événement. Par exemple, dans un

parking, le système détecte une personne qui s'approche d'une voiture. Le personnel de sécurité peut s'intéresser à savoir ce qu'elle a fait avant de s'approcher de la voiture. La deuxième application est l'étude statistique. Il est intéressant de savoir combien de fois par mois un événement aura lieu ou quel événement suit un événement particulier. Par exemple, dans les supermarchés, combien de fois le client visite le rayon A et passe ensuite dans le rayon B.

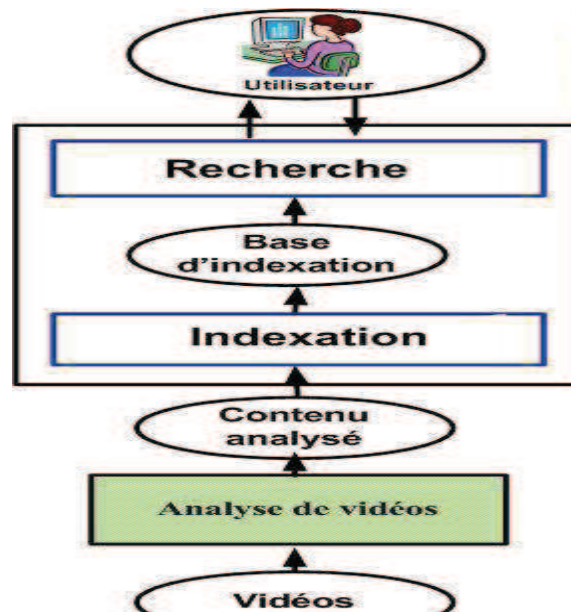
### 5.1. Indexation et recherche :

L'indexation constitue la principale solution au problème de la recherche d'information. Son principe consiste à extraire l'information pertinente des documents d'une base, pour les décrire de manière résumée par un ensemble caractéristique d'éléments de description. Ces éléments composent l'index à partir duquel on pourra effectuer des recherches.

Le premier niveau d'indexation, celui d'une base bibliographique, permet d'effectuer une requête sur une notice du type titre, auteur et mots-clés. Au-delà, l'indexation se fonde sur le contenu complet du document pour obtenir une description plus approfondie qui permette une recherche plus précise portant sur des éléments du contenu. Ainsi, pour les documents textuels, les moteurs de recherche en texte intégral indexent les mots du document, voire les concepts qu'ils représentent. Les documents audiovisuels ont aussi ce double contenu de fond et de forme, auxquels correspondent des types de recherche différents : d'une part le contenu sémantique, les éléments présents, l'action, leur signification, d'autre part le contenu visuel, c'est-à-dire l'apparence de ces éléments, pour laquelle on utilise des types d'index spécifique à l'image. Toutefois, la différence fond et forme est beaucoup plus importante que dans le cas du texte. Le contenu visuel se situe en effet au niveau de la perception, alors que les mots sont prévus pour véhiculer des concepts.

Le processus d'indexation peut être réalisé manuellement par un documentaliste ou automatiquement par un système informatique. Toutefois, le documentaliste travaille au niveau sémantique, qui est hors de portée des algorithmes dans le cas général. Sa capacité d'interprétation demeure irremplaçable, et une voie intermédiaire est alors possible, en considérant les outils informatiques comme une assistance au travail du documentaliste. Ainsi, on utilise des thésaurus informatisés de termes qui permettent de diminuer la variabilité de l'indexation afin d'améliorer les résultats des recherches. De même, des outils d'analyse d'image permettent de faciliter le travail d'indexation en détectant certains types d'objets, comme les visages.

En traitement d'image, l'indexation est une application récente mais qui reprend les recherches en analyse d'image, vision par ordinateur, reconnaissance de forme, et l'aspect sémantique en interprétation d'image. Ce dernier domaine avait été abordé à une époque par l'intelligence artificielle, et demeure une question de recherche toujours aussi ouverte.



**Figure 8[4]** : l'architecture générale de l'indexation et de la recherche de vidéo de vidéosurveillance.

La **Figure 8** montre l'architecture générale de l'indexation et de la recherche de vidéo de vidéosurveillance. Les vidéos acquises sont tout d'abord prétraitées par un module d'analyse vidéo. Les sorties de ce module sont ensuite les entrées de l'indexation et de la recherche.

Les caractéristiques du module d'analyse comprennent :

- ✚ En vidéosurveillance, une scène peut être observée par une ou plusieurs caméras. Dans le cas où les vidéos d'une scène sont enregistrées par plusieurs caméras, l'analyse peut être effectuée avant ou après la fusion des données.
- ✚ Il existe deux types de connaissances a priori : celle du contexte et celle du domaine. La connaissance du contexte décrit tout ce qui est présent dans la scène vide. La connaissance du domaine définit des événements d'intérêt pour le domaine considéré.
- ✚ L'analyse de vidéo est effectuée à différents degrés de granularité. Elle peut s'arrêter à la détection, au suivi, à la classification d'objets ou à la reconnaissance d'événements. Elle contient en générale la détection des objets, le suivi des objets, la classification des objets et parfois la reconnaissance des événements.
- ✚ Les résultats des modules d'analyse ne sont pas toujours parfaits. Cela a été montré dans les évaluations des modules d'analyses vidéo sur les bases communes de vidéos telles que **CAVIAR 7** (Contexte Aware Vision using Image based Active Recognition).

Les caractéristiques de l'indexation et de la recherche de vidéo de vidéosurveillance sont :

- ✚ Dans le cas où une scène est observée par plusieurs caméras, l'indexation et la recherche de vidéo de vidéosurveillance peuvent travailler sur des données qui sont déjà fusionnées ou sur des données qui sont séparément traitées dans les modules d'analyse vidéo.
- ✚ L'indexation et la recherche de vidéo de vidéosurveillance peuvent être effectuées à différents niveaux : images, objet et événements.

✚ L'objectif de la recherche de vidéo de vidéosurveillance est de retrouver les frames contenant des objets d'intérêt, des événements intéressants selon des critères. Ces critères peuvent concerner des attributs des objets, et/ou certains des événements. La recherche de vidéos de vidéosurveillance se fait donc au niveau plus fin que celle des journaux télévisés. Retrouver des plans entiers de vidéo n'est pas l'objectif de la recherche de vidéo de vidéosurveillance. Un des grands dés dans la recherche d'informations est le fossé sémantique entre la similarité calculée sur les informations indexées et la similarité attendue par l'utilisateur. Bien que le fossé sémantique soit causé par plusieurs facteurs, nous présentons les deux facteurs principaux. L'un des facteurs principaux est la distance entre le contenu de vidéos qui est riche en sémantique, et la méthode d'extraction automatique, qui essaie de représenter ce contenu par un ensemble de descripteurs de bas niveaux. L'autre facteur est le manque de conformité entre d'une part les informations indexées sous la forme de vecteurs de descripteurs à bas niveaux et d'autre part les requêtes exprimées par les utilisateurs. Les performances des approches d'indexation et de recherche sont mesurées par la capacité de combler le fossé sémantique. Pour l'indexation et la recherche de vidéo de vidéosurveillance, une approche qui permet de combler le fossé sémantique doit :

- Porter des résultats des modules d'analyse vidéo car de nombreuses approches avec certains de résultats ont été proposées pour l'analyse de vidéos. Pour pouvoir utiliser des approches différentes, l'indexation et la recherche doivent être générales.
- Pourvoir d'une part corriger des erreurs produites par l'imperfection des modules d'analyse vidéo, d'autre part compléter l'indexation effectuer par les sorties des modules d'analyse vidéo.
- Etre interactif : l'interaction avec l'utilisateur nous permet d'une part de comprendre mieux les requêtes des utilisateurs et d'autre part d'avoir les jugements des utilisateurs sur les résultats de recherche.

## 6. Conclusion :

Nous avons illustrés dans ce chapitre le système de vidéo surveillance, ces composants matériel et ces technologies ainsi ces domaines d'application. Comme on a présentés quelques notions sur l'indexation et la recherche de vidéo de surveillance. Les systèmes de vidéosurveillance existent depuis environ 25 ans. Intégralement analogiques à leurs débuts, ils ont évolué progressivement vers la technologie numérique. Les systèmes actuels ne ressemblent plus guère aux anciennes caméras analogiques branchées sur des magnétoscopes traditionnels. Aujourd'hui, ils utilisent les caméras réseau et les serveurs informatiques pour l'enregistrement vidéo dans un système entièrement numérique. Entre les systèmes entièrement analogiques et les systèmes entièrement numériques, il existe encore néanmoins toute une série de solutions partiellement numériques incluant une quantité variable de composants numériques.

# Chapitre 2

## 1. Introduction :

Google a fait face au problème d'analyser les ensembles énormes (l'ordre de petatype), par exemple pagerank, rondins d'accès de web, etc.

L'algorithme pour traiter des données peut être simple raisonnable. Pour cela les programmeurs ont fait une solution à ce problème qui nommé MapReduce. MapReduce est un cadre à l'aide que nous pouvons écrire des applications pour traiter d'énormes quantités de données, en parallèle, sur de grandes grappes de matériel de base d'une manière fiable.

## 2. Présentation de MapReduce :



### 2.1. Définition de MapReduce [1] :

MapReduce est une technique de traitement et un modèle de programme pour le calcul distribué basé sur java. L'algorithme de MapReduce contient deux tâches importantes, à savoir Map et Reduce. Map prend un ensemble de données et le convertit en un autre ensemble de données, où les éléments individuels sont décomposés en tuples (paire clé/valeur). Deuxièmement, Reduce, qui prend la sortie d'une Map comme une entrée et combine ces tuples de données dans un petit ensemble de tuples. Comme la séquence du nom MapReduce implique, la tâche de réduire est toujours effectué après le travail de Map.

Le principal avantage de MapReduce est qu'il est facile de traitement de données à grande échelle sur de multiples nœuds de calcul. Selon le modèle de MapReduce, les primitives de traitement de données sont appelé cartographes et réducteurs. Décomposition d'une application de traitement de données dans cartographes et réducteurs est parfois triviale. Mais, une fois que nous écrivons une application sous forme de MapReduce, mise à l'échelle de l'application à courir sur des centaines, des milliers, voire des dizaines de milliers de machines dans un cluster est un simple changement de configuration. Cette simple évolutivité est ce qui a attiré de nombreux programmeurs d'utiliser le modèle de MapReduce.

### 2.2. L'algorithme [2] :

L'algorithme se divise en deux étapes principales :

-  Map : Emission de paires <clé, valeur> pour chaque donnée d'entrée lue.
-  Reduce : Regroupement des valeurs de clé identique et application d'un traitement sur ces valeurs de clé commune.

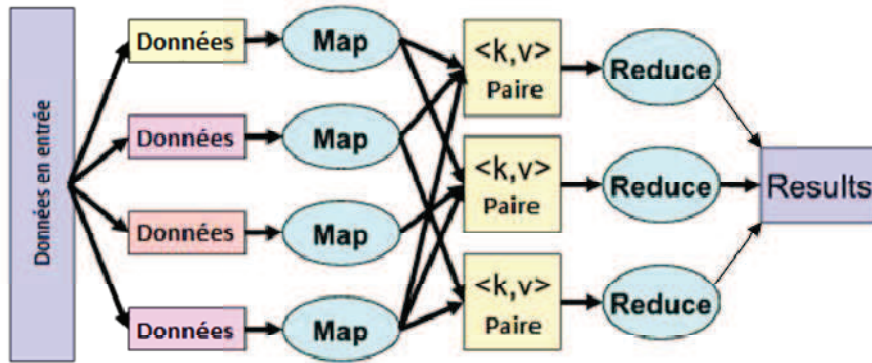


Figure 1 : fonctionnement du MapReduce.

**2.3. Comment écrire un programme MapReduce :**

- Choisir une manière de découper les données afin que Map soit parallélisable.
- Choisir la clé à utiliser pour notre problème.
- Ecrire le programme pour l'opération Map.
- Ecrire le programme pour l'opération Reduce.

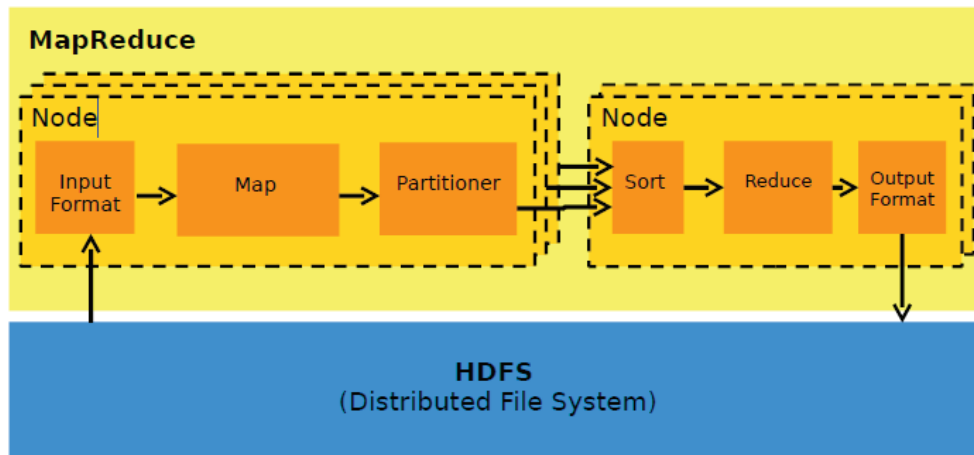
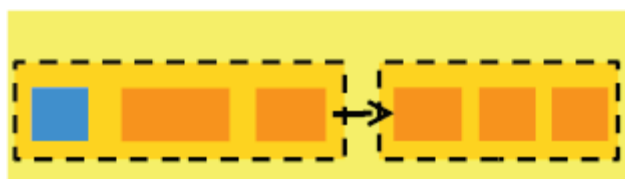


Figure 2 : Architecture basique de MapReduce.

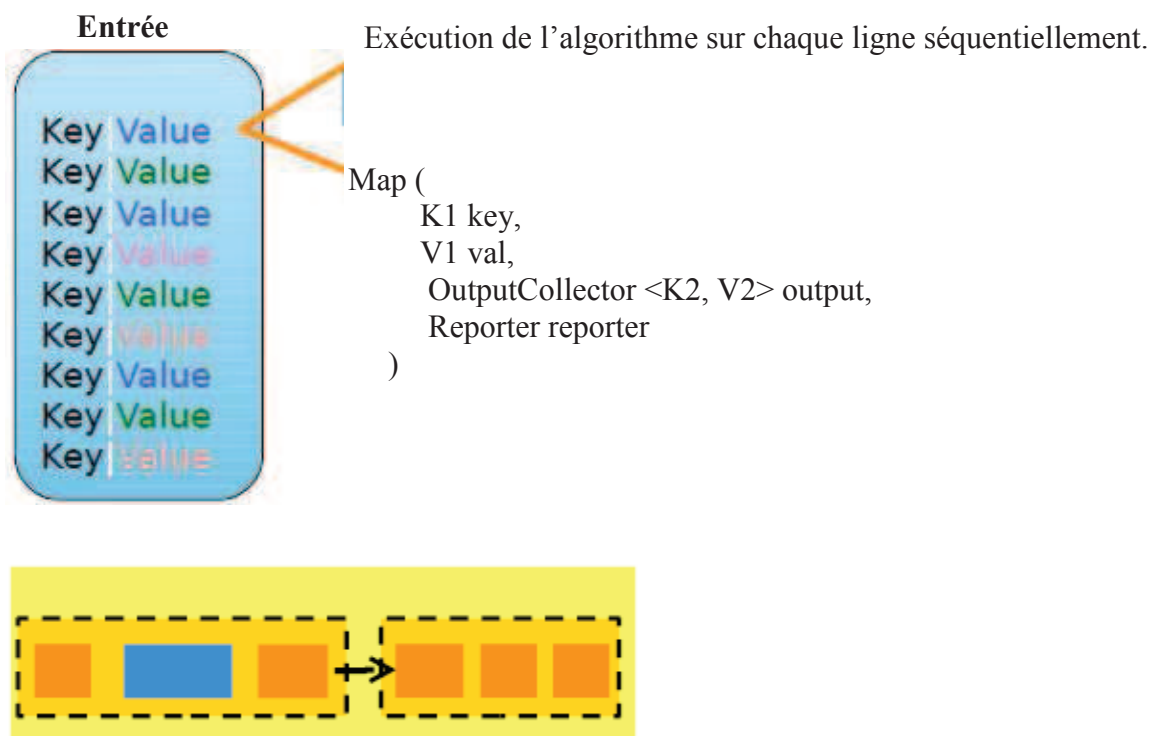
**MapReduce, Entrée (Input) :**



- Détermine la façon dont les données sont fractionnées.
- Crée des tableaux d'input split.
- Chaque tableau est un Map.
- Associé à une liste d'emplacement de nœuds.
- Record Reader.
- Crée des paires clé-valeur
- Conversion des types de données.

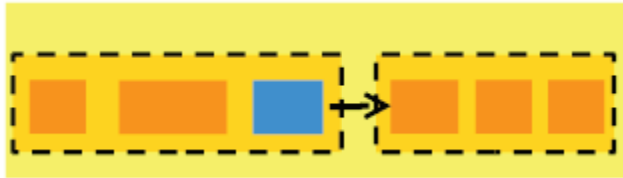
### MapReduce, Map :

Agit individuellement sur chaque data set.

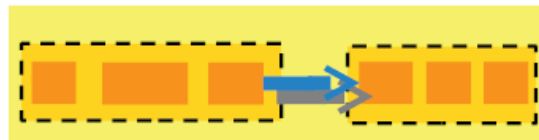
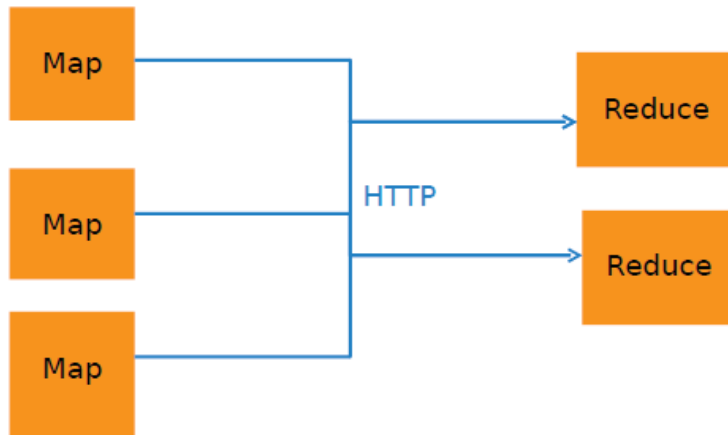


### MapReduce, Partitionner :

- Distribue les paires clés-valeurs.
  - Décide vers du Reducer cible.
  - Grâce à la cléexploite :
    - Une fonction de hachage par défaut
    - Ou une fonction personnalisée
- getPartition (K2 key, V2 value, int numPartitions)

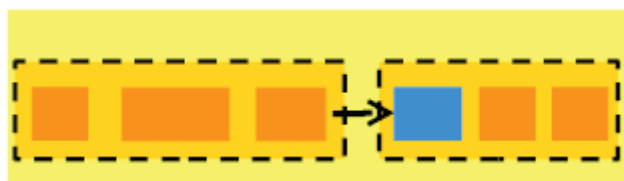


### MapReduce, Shuffle:



### MapReduce, Sort:

- Garantie que les entrées sont triées.
- Etape finale du shuffle.
- Aide à la fusion des entrées du Reducer.

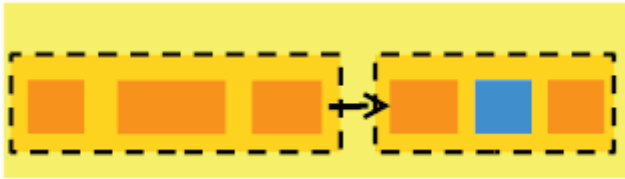


### MapReduce, Reduce :

- Reçoit les sorties depuis beaucoup de Mappers.
- Consolide les valeurs pour les clés intermédiaires.
- Regroupe les valeurs selon les clés.

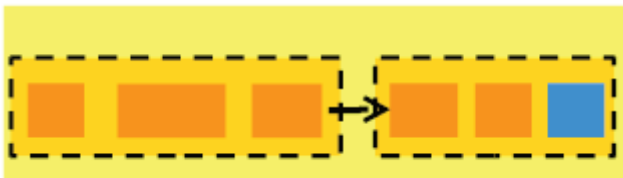
```
Reduce (
  K2 key,
  Iterator <V2> values,
  OutputCollector <K3, V3> output,
```

Reporter reporter  
)



#### ✚ MapReduce, Sorties (Output):

- Valide.
  - ✓ par rapport à la sortie attendue
- Créer un RecordWriter.
  - ✓ qui écrit sur HDFS
  - ✓ organise la sortie en fichiers « morceau-0000x ».



#### 2.4. Les composants d'un processus MapReduce :

- Split : divise les données d'entrée en flux parallèle à fournir aux nœuds de calcul.
- Read : lit les flux de données en découpant en unités à traiter. Par défaut à partir d'un fichier texte : unité=ligne.
- Map : applique sur chaque unité envoyé par le Reader un traitement de transformation dont le résultat est stocké dans des paires <clé, valeur>.
- Combine : composant facultatif qui applique un traitement de réduction anticipé à des fins d'optimisation. Cette étape ne doit pas perturber la logique de traitement.
- Group : regroupement (ou shuffle) des clés communes. Réalisé par un tri distribué sur les différents nœuds du cluster.
- Partition : distribue les groupes de pair sur les différents nœuds de calcul pour préparer l'opération de réduction. Généralement effectué par simple hachage et découpage en morceaux de donnée de tailles égales (dépend du nombre de nœuds Reduce).
- Reduce : applique un traitement de réduction à chaque liste de valeurs regroupées.
- Write : écrit le résultat du traitement dans le fichier résultat. On obtient autant de fichiers résultats que l'on a de nœuds de réduction.

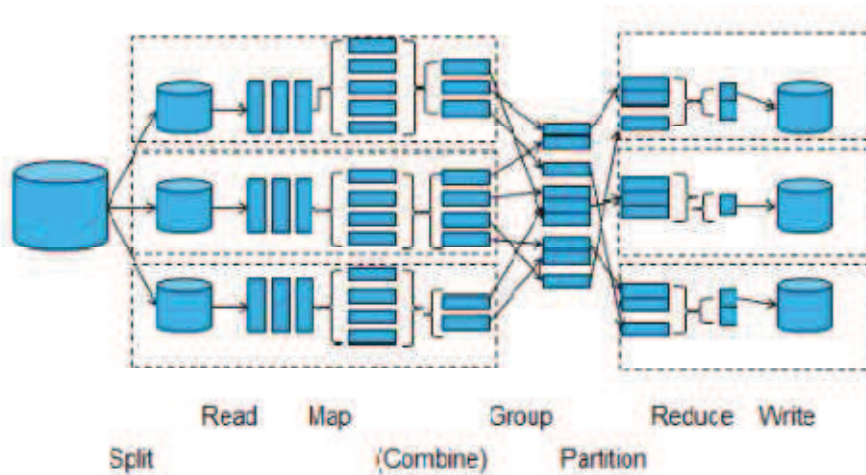


Figure 3 : les composants d'un processus MapReduce.

### 2.5. Domaines d'application [3] :

MapReduce peut être utilisé pour un grand nombre d'applications, dont grep distribué, tri distribué, inversion du graphe des liens web, vecteur de terme par hôte, statistiques d'accès au web, construction d'index inversé, classification automatique de documents, apprentissage automatique, traduction automatique statistique (distributed grep, distributed sort, web link-graph reversal, term-vector per host, web access log stats inverted index construction, document clustering, machine learning, statistical machine translation).

De manière plus significative, quand MapReduce fut terminé, il été utilisé pour régénérer entièrement les index internet de Google, et a remplacé les vieux programmes ad hoc utilisés pour la mise à jour de ces index et pour les différentes analyses de ces index.

MapReduce génère un large nombre d'intermédiaires et de fichiers temporaires, qui sont généralement gérés et accédés via le Google File System pour de meilleures performances.

- + **Utilisé dans Google [4] :**
  - ✓ bâtiment d'index pour la recherche de Google.
  - ✓ article groupant pour Google actualités.
  - ✓ traduction automatique statistique.
  
- + **Utilisé dans Yahoo [4] :**
  - ✓ bâtiment d'index pour la recherche Yahoo.
  - ✓ détection de spam pour les courriers Yahoo.
  
- + **Utilisé dans Facebook [4] :**
  - ✓ optimisation d'annonce.

- ✓ détection de spam.

#### ✚ **Autre domaines [4] :**

- ✓ Analyse de risques.
- ✓ Suivi d'opérations bancaires (Ecart d'utilisations/ Utilisations frauduleuse de carte bancaire).
- ✓ Analyse d'informations clients (ex : suivi clients, prospections...).
- ✓ Moteur de recommandation de produit (ex : Amazon, Netflix).
- ✓ Récolte d'informations pour le Retargeting de publicités.
- ✓ Données des réseaux sociaux (ex : Twitter).
- ✓ Suivi de comportement utilisateurs devant une page web (Netflix, Online learning)

## 2.6. Exemple illustratif (wordcount) [5] :

WordCount est une application simple qui compte le nombre d'occurrences de chaque mot dans un ensemble donné d'entrée.

### 2.6.1. Monde de fonctionnement :

L'opération de comptage des mots se déroule en deux étapes à la phase de mappeur et une phase de réducteur. Dans la phase de mappeur d'abord le test est segmenté en mots puis nous formons une paire de valeur de clé avec ces mots où la clé est le mot lui-même et la valeur « 1 ». Par exemple :

- **Fichier d'entrée :**

```
Celui qui croyait au ciel  
Celui qui n'y croyait pas  
[...]  
Fou qui fait le délicat  
Fou qui songe à ses querelles
```

(Louis Aragon, *La rose et le Réséda*, 1943, fragment)

Pour simplifier, on retire tout symbole de ponctuation et caractère spéciaux. On passe l'intégralité du texte en minuscules.

- ✓ Découpage des données d'entrée : par exemple par ligne

celui qui croyait au ciel

celui qui ny croyait pas

fou qui fait le delicat

fou qui songe a ses querelles

✓ Ici, 4 unités de traitement après découpage.



Opération Map :

- ✓ Séparation de l'unité en mots (selon les espaces).
- ✓ Emission d'une paire <mot, 1> pour chaque mot

celui qui croyait au ciel → (celui;1) (qui;1) (croyait;1) (au;1) (ciel;1)

celui qui ny croyait pas → (celui;1) (qui;1) (ny;1) (croyait;1) (pas;1)

fou qui fait le delicat → (fou;1) (qui;1) (fait;1) (le;1) (delicat;1)

fou qui songe a ses querelles → (fou;1) (qui;1) (songe;1) (a;1) (ses;1)  
(querelles;1)



Après le Map : regroupement (ou shuffle) des clés communes.

- ✓ Effectué par un tri distribué.
- ✓ Pris en charge de manière automatique par Hadoop.

(celui;1) (celui;1)

(qui;1) (qui;1) (qui;1) (qui;1)

(croyait;1) (croyait;1)

(au;1) (ny;1)

(ciel;1) (pas;1)

(fou;1) (fou;1)

(fait;1) (le;1)

(delicat;1) (songe;1)

(a;1) (ses;1)

(querelles;1)



Opération Reduce :

- ✓ Sommation des valeurs de toutes les paires de clé commune.
- ✓ Ecriture dans un (ou des) fichiers résultats.

```

qui: 4
celui: 2
croyait: 2
fou: 2
au: 1
ciel: 1
ny: 1
pas: 1
fait: 1
[...]
    
```

Cela donnerait le nombre d'occurrence de chaque mot dans l'entrée. Ainsi réduire les formes d'une phase d'agrégation pour les clés.

Le point à noter ici est que la première de la classe mappeur exécute entièrement sur l'ensemble des données fendage les mots et formant les paires de valeurs clés initiale. Seulement après tout ce processus est terminé le réducteur démarre. Dites si nous avons un totale de 10 lignes dans nos fichiers d'entrée combinés ensemble, d'abord les 10 lignes sont segmenté et des paires de valeurs clés sont formée en parallèle, seulement après cette agrégation /réducteur commencerait son fonctionnement.

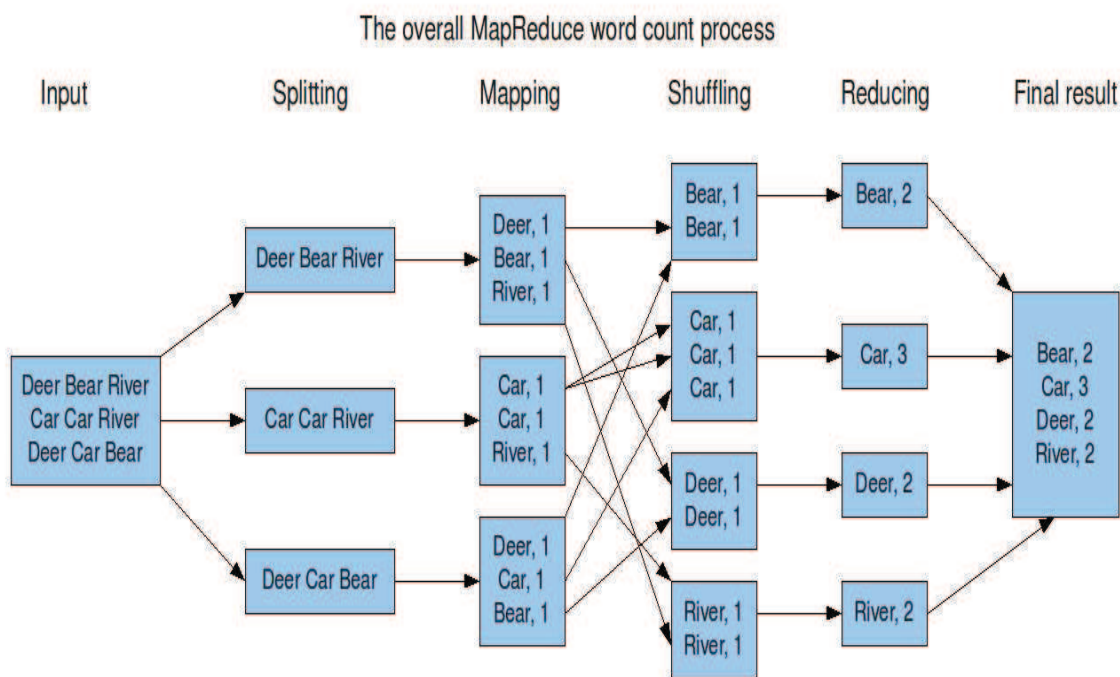


Figure 4 : exemple de wordcount.

## 2.6.2. Intégration ou utilisation de MapReduce avec des framework de cloud computing [6] :

### 2.6.2.1. Hadoop :

#### 2.6.2.1.1. Définition de Hadoop et ces composants :

Hadoop est un framework libre et open source écrit en java destiné à faciliter la création d'applications distribuées (au niveau du stockage des données et de leurs traitements) et échelonnables (scalables) permettant aux applications de travailler avec des milliers de nœuds et des pétaoctets de données. Ainsi chaque nœud est constitué de machines standard regroupées en grappe. Tous les modules de Hadoop sont conçus dans l'idée fondamentale que les pannes matérielles sont fréquentes et qu'en conséquence elles doivent être gérées automatiquement par le framework.

Hadoop a été inspiré par les publications MapReduce, et BigTable de Google. Hadoop a été créé par Doug Cutting et fait partie des projets de la fondation logicielle Apache depuis 2009.

Le noyau d'Hadoop est constitué d'une partie de stockage : Hadoop Distributed File System (HDFS), et une partie de traitement appelé MapReduce. Hadoop fonctionne les fichiers en gros blocs et les distribue à travers les nœuds du cluster. Pour traiter les données, Hadoop transfère le code à chaque nœud et chaque nœud traite les données dont il dispose. Cela permet de traiter l'ensemble des données plus rapidement et plus efficacement que dans une architecture supercalculateur plus classique qui repose sur un système de fichiers parallèle où les calculs et les données sont distribués via les réseaux à grande vitesse.

Le framework Hadoop de base se compose des modules suivants :

#### 1. MapReduce :

- ✓ Exécuter des jobs soumis par des utilisateurs.
- ✓ Contrôle la distribution et la défaut-tolérance de travail.

#### 2. Distributed File System (HDFS)[7] :

Le HDFS est un système de fichiers distribué, extensible et portable développé par Hadoop à partir du GoogleFS. Ecrit en java, il a été conçu pour stocker de très gros volumes de données sur un grand nombre de machines équipées de disques durs banalisés. Il permet d'abstraction de l'architecture physique de stockage, afin de manipuler un système de fichiers distribué comme s'il s'agissait d'un disque dur unique.

Une architecture de machines HDFS (aussi appelée cluster HDFS) repose sur deux types de composants majeurs :

#### NameNode :

Nœud de nom, ce composant gère l'espace de nom, l'arborescence du système de fichiers et les métadonnées des fichiers et des répertoires. Il centralise la localisation des

blocs de données répartis dans le cluster. Il est unique mais dispose d'une instance secondaire qui gère l'historique des modifications dans le système de fichiers (rôle de backup). Ce NameNode secondaire permet la continuité du fonctionnement du cluster Hadoop en cas de panne du NameNode d'origine.

### **DataNode :**

Nœud de données, ce composant stocke et restitue les blocs de données. Lors du processus de lecture d'un fichier, le NameNode est interrogé pour localiser l'ensemble des blocs de données. Pour chacun d'entre-eux, le NameNode renvoie l'adresse du DataNode qui dispose de la plus accessible, c'est-à-dire le DataNode qui dispose de la plus grande bande passante. Les DataNodes communiquent de manière périodique au NameNode la liste des blocs de données qu'ils hébergent. Si certains de ces blocs ne sont pas assez répliqués dans le cluster, l'écriture de ces blocs s'effectue en cascade par copie sur d'autres.

Chaque DataNode sert de bloc de données sur le réseau en utilisant un protocole spécifique au HDFS. Le système de fichiers utilise la couche TCP/IP pour la communication. Les clients utilisent le Remote Procedure Call pour communiquer entre eux. Le HDFS stocke les fichiers de grande taille sur plusieurs machines. Il réalise la fiabilité en répliquant les données sur plusieurs hôtes et par conséquent ne nécessite pas de stockage RAID sur les hôtes. Avec la valeur par défaut de réplification, les données sont stockées sur trois nœuds : deux sur le même support et l'autre sur un support différent. Les dataNodes peuvent communiquer entre-eux afin de rééquilibrer les données et garder un niveau de réplification des données élevé.

Le HDFS n'est pas entièrement conforme aux spécifications POSIX, en effet les exigences relatives à un système de fichiers POSIX diffèrent des objectifs cible pour une application Hadoop. Le compromis de ne pas avoir un système de fichiers totalement compatible POSIX permet d'accroître les performances du débit de données.

Le HDFS récemment amélioré ses capacités de haute disponibilité, ce qui permet désormais au serveur de métadonnées principal d'être basculé manuellement sur une sauvegarde en cas d'échec (le basculement automatique est en cours d'élaboration). Les NameNodes étant le point unique pour le stockage et la gestion des métadonnées, ils peuvent être un goulot d'étranglement pour soutenir un grand nombre de fichiers, notamment lorsque ceux-ci sont de petite taille. En acceptant des espaces de noms multiples desservis par des NameNodes séparés, le HDFS limite ce problème.

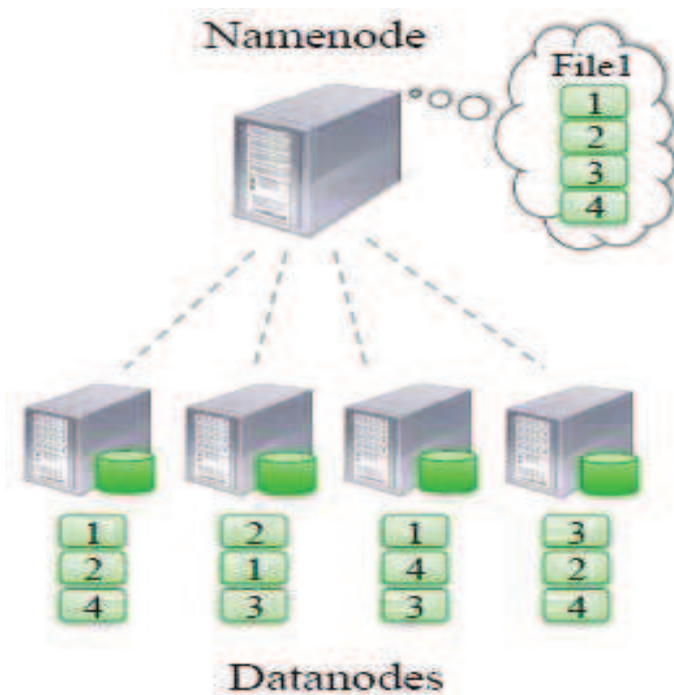


Figure 5: architecture de HDFS.

### 2.6.2.2. Architecture de Hadoop [8] :

Comme pour HDFS, la gestion des tâches de Hadoop se base sur deux serveurs (des daemons) :

✚ **Le job tracker** qui va directement recevoir la tâche à exécuter (un .jar java), ainsi que les données d'entrées (nom des fichiers stockés sur HDFS) et le répertoire où stocker les données de sortie (toujours sur HDFS). Il y a un seul JobTracker sur une seule machine du cluster Hadoop. Le JobTracker est en communication avec le NameNode de HDFS et sait donc où sont les données.

✚ **Le TaskTracker**, qui est en communication constante avec le JobTracker et va recevoir les opérations simples à effectuer (MAP/REDUCE ainsi que les blocs de données correspondants (stockés sur HDFS). Il y a un TaskTracker sur chaque machine du cluster.

- Comme le JobTracker est conscient de la position des données (grâce au NameNode), il peut facilement déterminer les meilleures machines auxquelles attribuer les sous-tâches (celles où les blocs de données correspondants sont stockés).
- Pour effectuer un traitement Hadoop, on va stocker nos données d'entrée sur HDFS, créer un répertoire où Hadoop stockera les résultats sur HDFS, et compiler nos programmes MAP et REDUCE au sein d'un .jar java.
- On soumettra alors le nom des fichiers d'entrée, le nom du répertoire des résultats, et le .jar lui-même au JobTracker. Il s'occupera du reste (et notamment de transmettre les programmes MAP et REDUCE aux serveurs TaskTracker des machines du cluster).

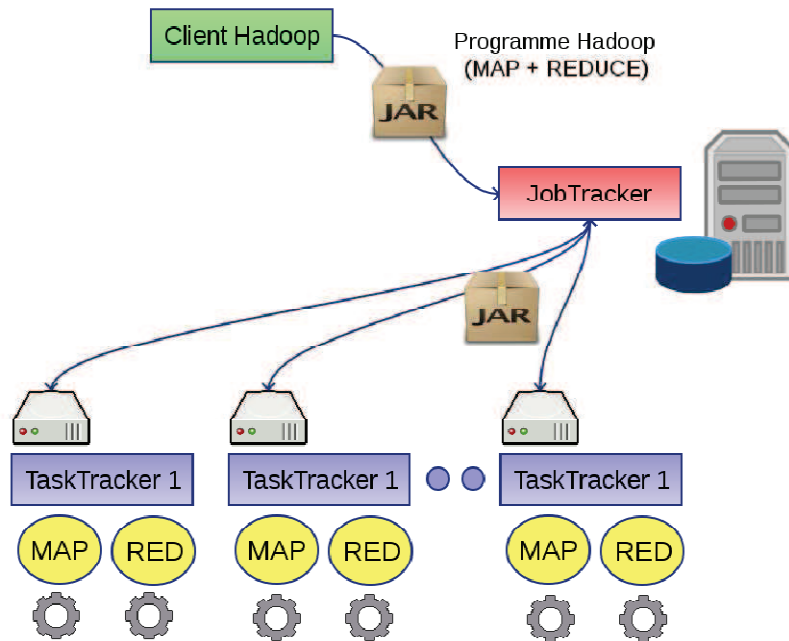


Figure 6 : architecture de Hadoop.

Le déroulement de l'exécution d'une tâche Hadoop suit les étapes suivantes du point de vue du JobTracker :

➤ **Le JobTracker :**

1. Le client (un outil Hadoop console) va soumettre le travail à effectuer au JobTracker : une archive java .jar implémentant les opérations MAP et REDUCE. Il va également soumettre le nom des fichiers d'entrée, et l'endroit où stocker les résultats.
2. Le JobTracker communique avec le NameNode HDFS pour savoir où se trouvent les blocs correspondant aux noms de fichiers donnés par le client.
3. Le JobTracker, à partir de ces informations, détermine quels sont les nœuds TaskTracker les plus appropriés, c'est-à-dire ceux qui contiennent les données sur lesquelles travailler sur la même machine, ou le plus proche possible (même rack/rack proche).
4. Pour chaque « morceau » des données d'entrée, le JobTracker envoie au TaskTracker sélectionné le travail à effectuer (MAP/REDUCE, code java) et les blocs de données correspondant.
5. Le JobTracker communique avec les nœuds TaskTracker en train d'exécuter les tâches. Ils envoient régulièrement un « heartbeat », un message signalant qu'ils travaillent toujours sur la sous-tâche reçue. Si aucun heartbeat n'est reçu dans une période donnée,

le JobTracker considère la tâche comme ayant échoué et donne le même travail à effectuer à un autre TaskTracker.

6. Si par hasard une tâche échoue (erreur java, données incorrectes, etc.), le TaskTracker va signaler au JobTracker que la tâche n'a pas pu être exécutée. Le JobTracker va alors décider de la conduite à adopter : redonner la sous-tâche à un autre TaskTracker, demander au même TaskTracker de ré-essayer, marquer les données concernées comme invalides, etc. il pourra même blacklister le TaskTracker concerné comme non-fiable dans certains cas.
7. Une fois que toutes les opérations envoyées aux TaskTracker (MAP+REDUCE) ont été effectuées et confirmées comme effectuées par tous les nœuds, le JobTracker marque la tâche comme « effectuée ». des informations détaillées sont disponibles (statistiques, TaskTracker ayant posé problème, etc.).

Par ailleurs, on peut également obtenir à tout moment de la part du JobTracker des informations sur les tâches en train d'être effectuées : étape actuelle (MAP, SHUFFLE, REDUCE), pourcentage de complétion, etc.

#### ➤ **Le TaskTracker :**

- ✓ Le TaskTracker dispose d'un nombre de « slots » d'exécution. A chaque « slots » correspond une tâche exécutable (configurable). Ainsi, une machine ayant par exemple un processeur à 8coeurs pourrait avoir 16 slots d'opération configurés.
  - ✓ Lorsqu'il reçoit une nouvelle tâche à effectuer (MAP, REDUCE, SHUFFLE) depuis le JobTracker, le TaskTracker va démarrer une nouvelle instance de java avec le fichier .jar fourni par le JobTracker, en appelant l'opération correspondante.
  - ✓ Une fois la tâche démarrée, il enverra régulièrement au JobTracker ses messages heartbeats. En dehors d'informer le JobTracker qu'il est toujours fonctionnels, ces messages indiquent également le nombre de slots disponibles sur le TaskTracker concerné.
- Lorsqu'une sous-tâche est terminée, le TaskTracker envoie un message au JobTracker pour l'en informer, que la tâche se soit bien déroulée ou non (il indique évidemment le résultat au JobTracker).

#### **Remarque :**

- ✚ De manière similaire au NameNode de HDFS, il n'y a qu'un seul JobTracker et s'il tombe en panne, le cluster tout entier ne peut plus effectuer de tâches. Là aussi, des résolutions aux problèmes sont prévues dans la roadmap Hadoop, mais pas encore disponible.

- ✚ Généralement, on place le JobTracker et le NameNode HDFS sur la même machine (une machine plus puissante que les autres), sans y placer de TaskTracker/DataNode HDFS pour limiter la charge. Cette machine particulière au sein du cluster (qui contient les deux « gestionnaires »(de tâches et de fichiers) est communément appelée le nœud maître («Master Node»). Les autres nœuds (contenant TaskTracker+DataNode) sont communément appelés nœuds esclave (« slave node »).
- ✚ Même si le JobTracker est situé sur une seule machine, le « client » qui envoie la tâche au JobTracker initialement peut être exécuté sur n'importe quelle machine du cluster-comme les TaskTracker sont présents sur la machine, ils indiquent au client comment joindre le JobTracker.
- ✚ La même remarque est valable pour l'accès au système de fichiers : les DataNodes indiquent au client comment accéder au NameNode.
- ✚ Enfin, tout changement de configuration Hadoop peut s'effectuer facilement simplement en changeant la configuration sur la machine où sont situés les serveurs NameNode et JobTracker : ils répliquent les changements de configuration sur tout le cluster automatiquement.

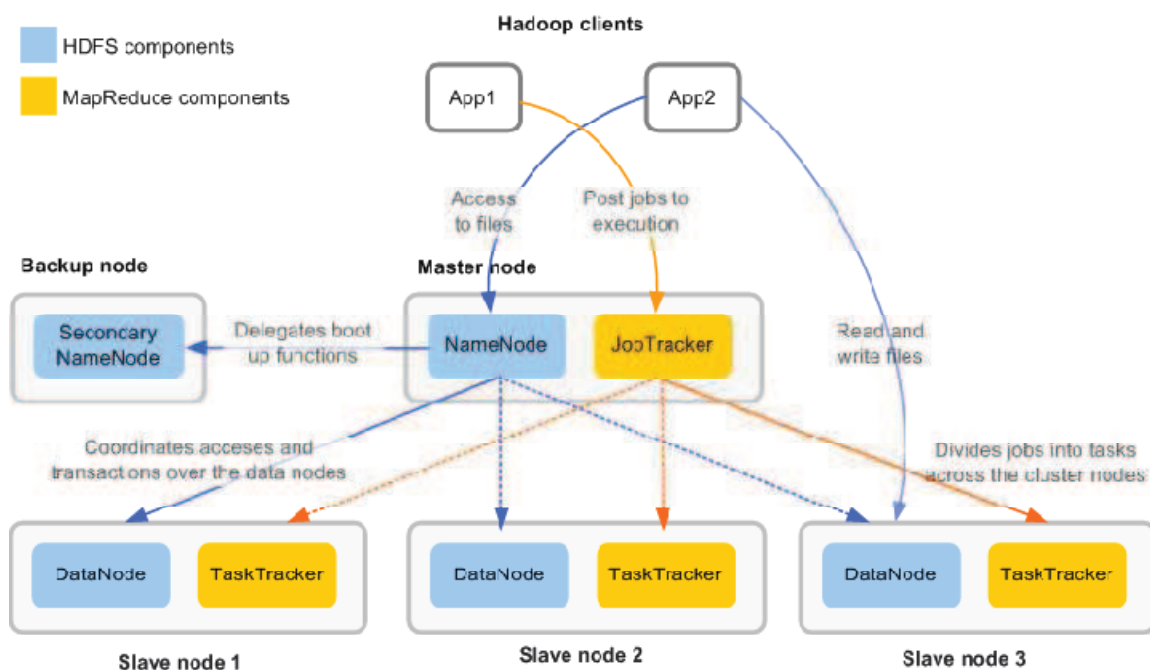


Figure 7 : position de MapReduce dans Hadoop.

### 2.6.2.3. MapReduce dans Hadoop :

Intéressons-nous maintenant à voir comment le modèle de programmation MapReduce est exploité dans le framework Hadoop.

**a. Etape du découpage des données (split) :**

La première étape concerne le découpage (split) des données. Cette étape est à la charge du framework qui se base sur le format des données. La taille de chaque découpage est fixe et généralement identique à la taille d'un bloc HDFS (64 Mo par défaut).

**b. Etape du Map :**

Hadoop crée pour chaque découpage de données une tâche de Map qui exécutera la fonction Map développée en conséquence. Chaque découpage de données n'est traité que par une seule tâche de Map.

Précisons que ce n'est pas la donnée qui est transportée vers le programme, mais l'inverse. Cela permet de pouvoir profiter d'une optimisation appelée optimisation de proximité de la donnée (Data Locality Optimisation en anglais). Hadoop va donc tenter de trouver le nœud le plus proche contenant la donnée pour y transférer la fonction Map.

Une fois la tâche Map terminée, le résultat de la fonction n'est pas stocké sur le système HDFS, mais sur le système de fichier local du nœud. En effet, il s'agit d'un résultat intermédiaire et qui n'a pas besoin d'être stocké de manière sécurisée via la réplication des blocs HDFS. Dans le cas d'une anomalie sur l'exécution de la tâche (nœud qui devient inaccessible par exemple), Hadoop est informé et pourra demander sa réexécution.

**c. Etape Reduce :**

Hadoop ne lance les tâches de Reduce qu'une fois que toutes les tâches de Map sont terminées. Notez que le nombre de tâches de Reduce n'est pas fonction de la taille des données en entrée mais est spécifié en paramètre de configuration d'exécution du job. C'est donc un paramètre qui peut être modifié. Par défaut, le nombre de tâches de Reduce est 1.

L'entrée de la fonction Reduce correspond à la sortie de l'ensemble des fonctions map. Comme la tâche de Reduce ne profite pas d'optimisation de proximité comme cela est fait pour la tâche de Map, les données de sortie de la fonction Map sont transférées via le réseau vers le nœud où la tâche de Reduce est réalisée.

Chaque tâche de Reduce produit un fichier de sortie qui sera stocké, cette fois, dans le système de fichiers HDFS. Le format du nom du fichier du produit est de la forme suivante : part-r-XXXX où XXXX est le numéro de la tâche de Reduce commençant par 0.

**d. Etape Combiner (facultative) :**

Précédemment, nous avons vu que les données de sortie de la fonction Map étaient transférées par le réseau vers le nœud où s'effectuera la tâche de Reduce. Bien entendu, cela peut avoir un impact sur les performances car les données à transmettre peuvent être volumineuses. Pour répondre à ce problème, une autre optimisation consiste à utiliser une fonction dite Combiner en sortie directe de la fonction Map. Le Combiner a le même comportement que la tâche Reduce et il s'appuie sur la même API.

Quand un Combiner est utilisé, la sortie de la fonction Map n'est pas écrite sur le système de fichiers local. Un traitement en mémoire est réalisé afin de regrouper les valeurs par clé (identique à shuffle et sort).

## 8. Conclusion :

Dans ce deuxième chapitre nous avons présentés l'une des méthodes qui manipule les données volumineuse, cette méthode est MapReduce c'est un cadre à l'aide que nous pouvons écrire des applications pour traiter d'énormes quantités de données, en parallèle, sur de grandes grappes de matériel de base d'une manière fiable. Ainsi qu'on a défini le framework Hadoop qui est une collection d'applications logicielles constitué de plusieurs composants tel que: **HDFS** (Hadoop Distributed File System) est un système de fichier utilisé pour stocker des pièces de fichiers à travers plusieurs ordinateurs, et il est composé aussi de **MapReduce** qui comprend deux fonctions : **Map** (qui permet de découper les données en plusieurs pièces) et **Reduce** (qui permet de combiner le résultat d'une analyse pour donner un seul résultat).

# Chapitre 3

## 1. Introduction :

la détection de visage est l'un des sujets les plus étudiés dans la littérature de vision informatique, non seulement à cause de la nature stimulante de visage comme un objet, mais aussi en raison des applications innombrables qui exigent l'application de détection de visage comme un premier pas. Pendant les 15 dernières années, le progrès énorme a été fait en raison de la disponibilité de données dans des conditions de capture non contraintes (prétendu 'dans la nature') par Internet, l'effort fait par la communauté pour développer des points de référence publiquement disponibles, aussi bien que le progrès dans le développement d'algorithmes de vision informatiques robustes. Dans ce chapitre nous allons présenter le système de détection de visage humains commençant par CBVIR ensuite on détaille le système de détection du visage. Au finale, en termine avec quelques méthodes de détection de visage humaine.

## 2. CBVIR (Content Based Video Indexing and Retrieval) [1] :

Depuis le début des années 90 l'indexation et la recherche d'image par le contenu sont devenues un pôle très actif de la recherche et de nombreux systèmes commerciaux et académiques ont été proposés. Puis rapidement des extensions sont apparues pour réaliser des systèmes d'indexation et de recherche de vidéos. A présent l'avancée technologique continue avec l'intégration automatique du contenu sémantique pour accéder à une recherche de plus haut niveau.

### 2.1. Description générale d'un système CBVR:

Les systèmes de recherche d'information basés sur le contenu, communément appelés CBIR (acronyme de « Content Based Information Retrieval »), font intervenir deux phases qui sont l'indexation et la recherche. Nous commençons par une présentation de ces deux phases avant de poursuivre par une rapide énumération des systèmes existants aujourd'hui.

#### 2.1.1. Indexation [2] :

L'indexation consiste à extraire, représenter et organiser efficacement le contenu des documents d'une base de données (figure 1). Pour cela, les documents sont tout d'abord représentés par une signature numérique qui permettra leur identification sous certains critères. Cette opération est réalisée en deux étapes. La première étape implique l'analyse des documents pour en extraire l'essentiel. Il s'agira par exemple de capturer les couleurs ou les textures caractéristiques, de déterminer les visages ou les objets présents. La seconde étape permet éventuellement de compresser l'information extraite tout en conservant l'essentiel. Il est important d'avoir des signatures compactes pour éviter d'avoir des données trop importantes à stocker et à traiter. Finalement, les signatures sont organisées au mieux afin d'optimiser la recherche de l'information. Dans la plupart des cas, une structure hiérarchique est utilisée pour organiser et faciliter l'accès aux signatures pertinentes. Habituellement une structure en arbre  $R^*$  (ou « $R^*$ -tree») est adoptée (Beckmann et al. [4]). Actuellement, la communauté scientifique du domaine ne se restreint plus à la capture du contenu visuel mais elle travaille de plus en plus sur l'inclusion automatique du contenu sémantique dans les signatures (Li and Wang [28]). La détection des objets et des visages est déjà un premier pas

dans ce sens ; et l'avancée se poursuit en détectant automatiquement des concepts de plus haut niveau qui vont permettre une réelle recherche sur le contenu similaire à celle réalisée sur les documents écrits. Les futures bases de données contiendront alors des signatures représentant le contenu brut et des signatures représentant explicitement le contenu sémantique. Et cette indexation mixte sera réalisée automatiquement.

### 2.1.2. Recherche

La recherche d'information est l'ensemble des opérations nécessaires pour répondre à la demande d'un utilisateur (figure 1.2). Tout d'abord, l'utilisateur doit construire une requête. Cette opération évidente pour le texte est bien plus difficile pour les images et encore plus pour la vidéo. La requête peut inclure différentes données : un exemple (image, vidéo, son), un dessin ou une animation. Dans le cas plus rare où des signatures sémantiques sont disponibles, l'utilisateur peut alors inclure des mots clés (Lu et al). En règle générale la difficulté est d'exprimer correctement l'objet de la requête en utilisant au mieux les moyens proposés par le système.

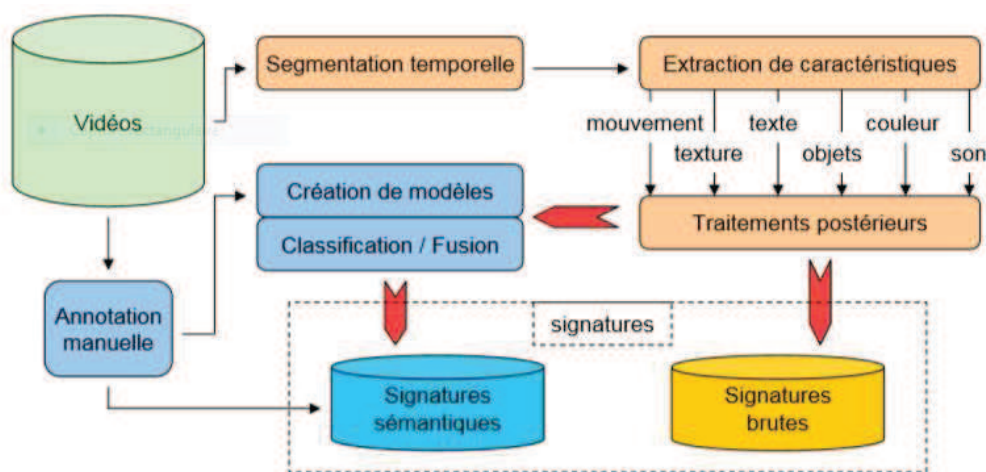


Figure 1 : phase d'indexation par le contenu.

La requête est ensuite transformée en signature en suivant un procédé similaire à l'indexation. Cette signature est alors comparée aux signatures de la base de données afin de retrouver l'information la plus pertinente. Toutefois il est particulièrement difficile de répondre aux exigences des utilisateurs à partir d'une seule requête. Il est alors utile d'intégrer un bouclage de pertinence incluant l'avis de l'utilisateur pour améliorer la requête en fonction du résultat précédemment obtenu. Un tel système permet également à l'utilisateur de clarifier sa demande qui est souvent mal formulée au début de la recherche.

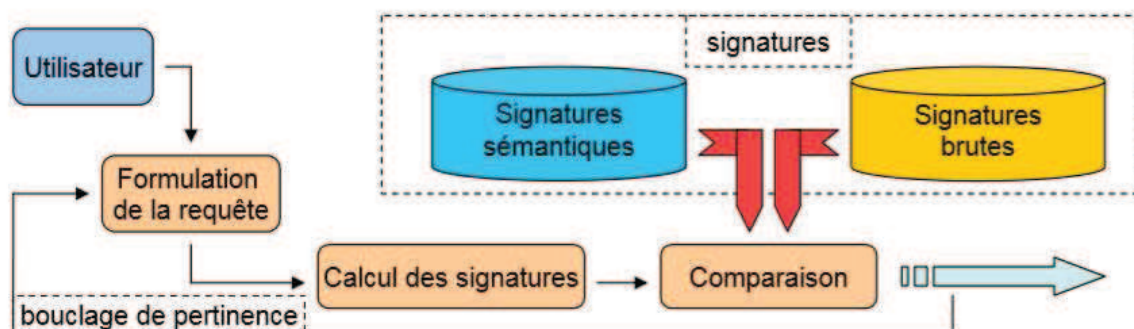


Figure 2 : phase de recherche par le contenu.

## 2.2. Les domaines d'application de l'indexation et la recherche de la vidéo [2]

La gestion des fichiers vidéo s'effectue de nos jours dans de nombreux domaines. Pour faciliter la recherche d'une vidéo donnée, il faut nécessairement l'indexer. Vu la facilité de se procurer des vidéos en grande quantité, même à domicile, cette tâche est devenue une nécessité pour gérer ses propres vidéos personnelles. Ainsi, on peut rencontrer l'indexation de la vidéo dans :

- ✚ Le domaine cinématographique où on peut trouver des films de plusieurs catégories tels que les films humoristiques, les films dramatiques, les films d'actions, etc. afin de retrouver un film donné dans un bref délai, il est donc nécessaire de faire de l'indexation.
- ✚ La sécurité : beaucoup d'organismes utilisent de nos jours des caméras de surveillance dans le cadre de la sécurité. L'indexation permet de retrouver rapidement une vidéo qui aurait été prise lors d'un incident quelconque.
- ✚ Le domaine du sport : par exemple, résumer un match de soccer de façon automatique, extraire les faits saillants d'un match de hockey, etc.
- ✚ L'information : par exemple, produire le résumé d'un journal télévisé, etc.
- ✚ Autre : les systèmes multimédia distribués, les bases de données multimédia, les jeux, le commerce électronique et les systèmes d'information géographiques.

Les vidéos personnelles, la télévision, le cinéma, et celles utilisées par les spécialistes dans les domaines scientifiques et artistiques. Le domaine d'application est alors un atout important dans le cadre de l'indexation car les termes qui sont utilisés pour annoter une vidéo peuvent varier d'un domaine à l'autre. Les vidéos peuvent donc être classées selon l'intention, le contenu, la production et l'usage.

## 3. Système de détection de visage humain :

### 3.1. Système de détection [3]

Un système de détection est un système permettant à l'utilisateur d'observer un événement automatique par le biais d'un appareil électronique ou mécanique. On peut distinguer plusieurs types de systèmes de détection, qui s'inscrivent dans de nombreux domaines. Tout d'abord la sécurité personnelle (système de détection d'intrusion), ou encore informatique (Système de détection d'intrusion de réseau), puis du fait de l'informatisation grandissante, les hommes ont cherché à identifier des personnes de façon automatique ex : **la détection du visage**.

### 3.2. Présentation [4]

La détection de visage est un domaine de la vision par ordinateur consistant à détecter un visage humain dans une image numérique. C'est un cas spécifique de détection d'objet, où l'on cherche à détecter la présence et la localisation précise d'un ou plusieurs visages dans une image. C'est l'un des domaines de la vision par ordinateur parmi les plus étudiés, avec de très nombreuses publications, brevets, et de conférences spécialisées. La forte activité de recherche en détection de visage a également permis de faire émerger des méthodes génériques de détection d'objet.

La détection du visage est un sujet difficile, notamment dû à la grande variabilité d'apparence des visages dans des conditions sans contraintes :

- Variabilité intrinsèque des visages humains (couleur, taille, forme).
- Présence ou absence de caractéristiques particulières (cheveux, moustache, barbe, lunettes...).
- Expression faciales modifiant la géométrie du visage
- Occultation par d'autres objets ou d'autres visages.
- Orientation et pose (de face, de profil).
- Condition d'illumination et qualité de l'image.

### 3.3. Caractéristique du visage humain :

Le **visage** est la zone externe de la partie antérieure du crâne de l'être humain, appelée aussi face ou figure. Il se structure autour de zones osseuses abritant plusieurs organes des sens ; il comprend notamment la peau, le menton, la bouche, les lèvres, le philtrum, les dents, le nez, les joues, les yeux, les sourcils, le front, les cheveux et les oreilles.

La phase de détection du visage humain repose sur la détection du nez, des yeux et de la bouche (parfois plus). Le problème de la détection du visage se réduit alors à trouver un arrangement géométrique correct pour les différentes parties détectées. Les yeux et la bouche sont les caractéristiques les plus utilisées [Hjelmas 99, Hsu 02, Xie 03], mais le nez est aussi parfois employé [Jeng 98]. Les descripteurs du visage peuvent aussi servir pour valider des visages candidats détectés par d'autres méthodes : par exemple dans

[Hsu 02], la détection est faite en utilisant une information sur la couleur, puis chaque candidat est validé en utilisant ses caractéristiques du visage.

### **3.3.1. Domaine d'application :**

La détection du visage a de très nombreuses applications directes en vidéo-surveillance, biométrie, robotique, commande d'interface homme-machine, photographie, indexation d'images et de vidéos, recherche d'image par le contenu, etc... elle permet également de faciliter l'automatisation complète d'autres processus comme la reconnaissance de visage ou la reconnaissance d'expressions faciales.

Parmi les applications directes, la plus connue est sa présence dans de nombreux appareils photo numériques, ou elle sert à effectuer la mise au point automatique sur les visages. C'est également une technique importante pour les interfaces homme-machine évoluées, afin de permettre une interaction plus naturelle entre un humain et un ordinateur.

La détection de visage est aussi utilisée en indexation d'images et recherche d'information, ou elle peut être utilisée pour rechercher des images contenant des personnes, associer directement un visage à un nom dans une page web, identifier les principales personnes dans une vidéo par clustering.

La détection du visage peut aussi servir à déterminer l'attention d'un utilisateur, par exemple face à un écran dans l'espace public, qui peut également, une fois le visage détecté, déterminer le sexe et l'âge de la personne afin de proposer des publicités ciblées. Cela peut être également servir à savoir si une personne est bien présente devant une télévision allumée, et dans le cas contraire mettre l'appareil en veille ou réduire la luminosité afin d'économiser de l'énergie. De façon plus indirecte, la détection de visage est la première étape vers des applications plus évoluées, qui nécessitent la localisation du visage, comme la reconnaissance de visage, la reconnaissance d'expressions faciales, l'évaluation de l'âge ou de sexe d'une personne, le suivi de visage ou l'estimation de la direction du regard et de l'attention visuelle.

## **3.4. Détection des visages humains et apprentissage du système**

L'apprentissage consiste à mémoriser les modèles calculés dans la phase d'analyse pour les individus connus. Un modèle est une représentation compacte des images qui permet de faciliter la phase de reconnaissance mais aussi de diminuer la quantité de données à stocker en quelque sorte l'apprentissage est la mémoire du système.

### **3.4.1. Méthodes d'apprentissage (approches statistiques) :**

De nos jours, les méthodes d'apprentissage statique sont très répandues pour la détection d'objets [Schneiderman 04, Schoikopf 02, Sung 98, Viola 05, Yang 01b]. L'apprentissage par l'exemple (ou apprentissage supervisé) fait référence à des systèmes qui sont mis au point à l'aide d'un ensemble d'exemples, qui sont un ensemble de couples (observation, classe) [Vapnik 98]. Le problème le plus répandu est le problème de

classification à deux classes : si on suppose qu'on a deux classes d'objets, chaque nouvel objet doit être assigné à l'une des deux. Quand ces techniques d'apprentissage sont appliquées à la détection de visages, les deux classes deviennent ((visage) et (non visage)). Plusieurs techniques d'apprentissage ont été proposées, mais toutes les approches utilisent le même principe. Tout d'abord, le classifieur est entraîné avec plusieurs centaines voire plusieurs milliers d'exemples de visages, appelés ((exemples positifs)), qui sont mis à la même échelle (habituellement  $20 \times 20$  pixels), et d'exemple négatifs, qui sont des morceaux d'images aléatoires de la même taille. Un exemple est donné dans la figure 3. De plus pour augmenter artificiellement le nombre d'échantillons, des exemples positifs virtuels sont parfois créés en inversant, effectuant une légère rotation, déplaçant ou changeant l'échelle des exemples de départ. Une fois la base d'apprentissage créée, la méthode d'apprentissage la plus utilisée est AdaBoost [Freund 97], qui construit de manière automatique un classifieur fort à partir de plusieurs classifieur fiables [Schneiderman 04, Viola 01, Xie 03]. Après que le classifieurs à été entraîné, il peut être appliqué à une région d'intérêt de l'image de la même taille que celle utilisée pour les données d'apprentissage. L'image d'entrée est alors parcourue avec une fenêtre rectangulaire pour déterminer si un visage est présent ou non. Pour détecter des visages à différentes échelles, l'image est sous échantillonnée par un facteur donné (habituellement 1.2), et parcourue ainsi plusieurs fois à différentes échelles. Un exemple est présenté sur la figure 2. Les différentes applications qui utilisent cette approche diffèrent par la méthode de classification employée pour vérifier chaque partie d'image : à chaque position de fenêtre glissante, le système doit classer la région comme ((visage) ou (non visage)) (figure 3). Cette classification peut être effectuée par réseaux de neurones [Rowley 98, Sung 98], par SVM [Fang 03, Fransens 03] ou encore par cascade de classifieurs simple utilisant des filtres de type Haar [Lienhart 02, Viola 01].

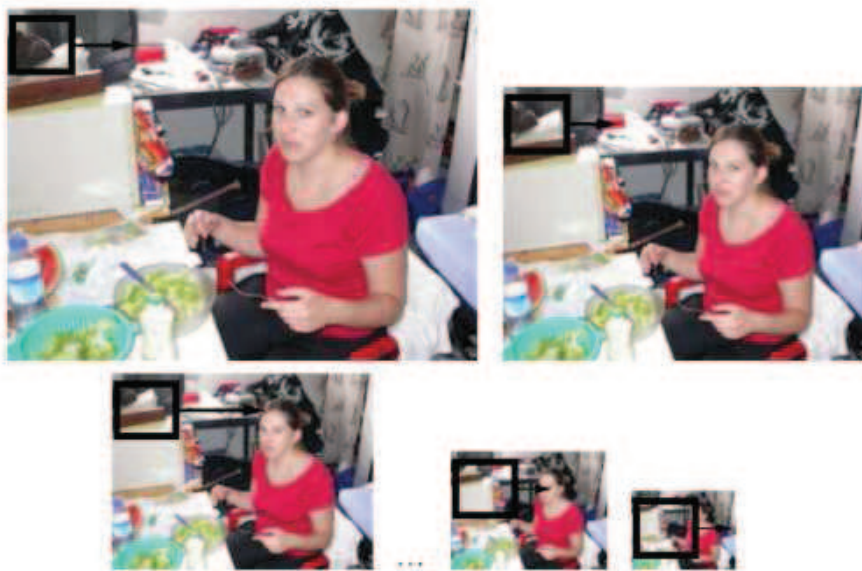


Figure 3. Détection de visage par classification.

Une fenêtre de taille des modèles est déplacée dans toute l'image et en chaque position la présence ou non d'un visage. Puis l'image est réduite, et la recherche est réitérée avec la même taille de fenêtre, jusqu'à que l'image soit trop petite.

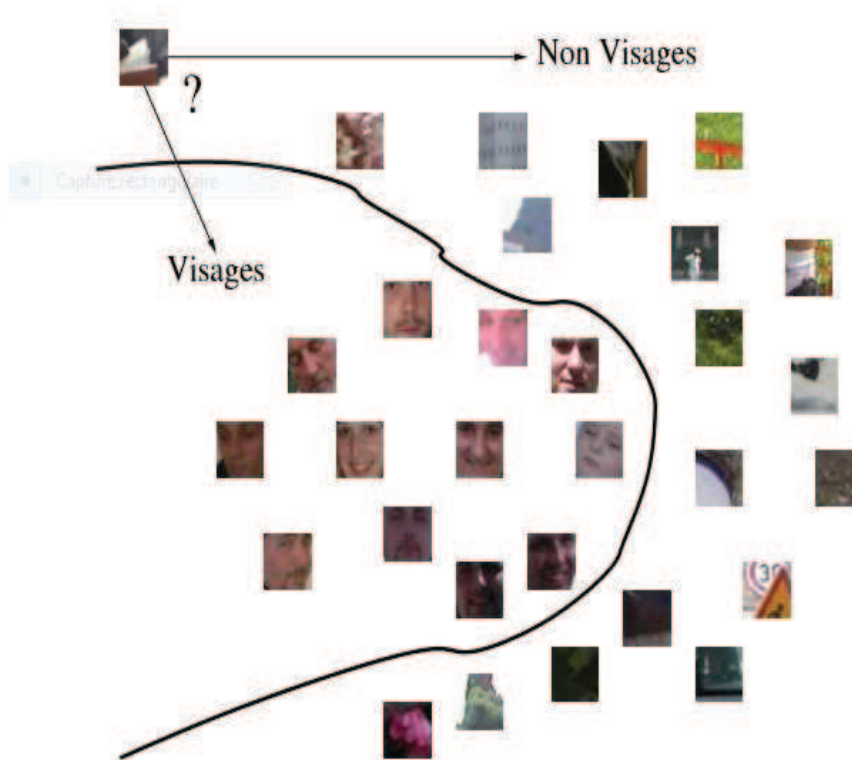


Figure 4. Classification d'une région de l'image.

Lors du parcours de l'image par la fenêtre glissante (cf.figure 2), chaque morceau d'image doit être classifié en ((visage) ou (non visage)). Cette classification se fait à partir des données d'apprentissage. Les différentes applications qui utilisent cette approche diffèrent par la méthode de classification employée (SVM [Fang 03, Fransens 03], réseaux de neurones [Rowley 98, Sung 98], etc.).

#### 4. Enquête sur les méthodes de détection du visage humain [6]

Depuis les dix dernières années, les méthodes de détection de visages sont de plus en plus performantes. La détection de visages n'est certes pas un problème résolu, mais les méthodes actuelles fournissent des résultats rapides (proches du temps réel) de très bonne qualité si les images ne sont pas trop complexes ou trop bruitées. Ainsi, plusieurs approches de détection de personnes [Jaffré 04, Jin 03, Lee 04, Sprague 02, Zhai 04]. Simplifient le problème de détection en cherchant les visages présents dans l'image : si un visage est détecté, elles déduisent qu'une personne est présente. Quand les coordonnées du corps des personnes sont recherchées, elles sont alors déterminées en fonction de la taille et de la position du visage [Jaffré 04, Zhai04]. La détection de visages est un problème très difficile, à cause des changements possibles d'échelle, des changements de couleur d'un visage à un autre, de l'orientation variable (vue de face ou de profil), etc. de plus, l'apparence visuelle d'un visage peut changer en fonction des expressions faciales ou des conditions d'éclairage.

Il existe dans la littérature de très nombreux algorithmes de détection de visages, qui peuvent être plus ou moins efficaces. Les articles [Hjelmas 01, Yang 02] ainsi que le livre [Yang 01a] sont les états de l'art les plus récents sur la détection de visages, mais ils

n'incluent pas les travaux tels que la détection par mean shift [Bradski 98, Comaniciu 00], et surtout les méthodes de détection qui utilisent des filtres de type Haar [Lienhart 02, Viola 01] qui sont très répandues aujourd'hui. Les algorithmes de détection de visages utilisent différents méthodes, que l'on peut regrouper en différentes classes :

- Les analyses de bas niveaux, qui emploient des descripteurs tels que la couleur ou les contours.
- L'analyse de caractéristiques, qui décompose un visage en caractéristiques visuelles (nez, bouche, yeux) et les méthodes à base d'apprentissage.

Certaines applications associent plusieurs méthodes et peuvent être à cheval sur plusieurs classes. Contrairement aux méthodes d'indexation traditionnelles, telles que les descripteurs de texture, ou de couleur, et à l'instar des détecteurs de personnes, il n'existe pas de norme concernant la détection de visages. Cependant le domaine devient de plus en plus structuré. Des efforts sont faits pour proposer des outils d'apprentissage et d'évaluation communs. De plus, il existe aujourd'hui en ensemble de détecteurs considérés comme standards par la communauté [Osuna 97, Rowley 98, Sung 98], et plus récemment [Viola 01], auxquels les nouvelles méthodes se comparent. Ces détecteurs utilisent tous des méthodes à base d'apprentissage. Même si ces méthodes sont évaluées avec même données et présentent des taux de détection de l'ordre de 90%, les données de tests sont des photos. Il s'agit très rarement d'images extraites de séquences vidéo, ce qui est le cas des contenus que nous étudions. Dans les séquences vidéo extraites d'émission télévisée, même si le caméraman essaie de cadrer les personnes qu'il filme, il ne peut pas empêcher les effets dus aux mouvements des personnes, en particulier lorsque les émissions sont tournées en direct. De plus, à cause de la compression des vidéos, la qualité des images est beaucoup moins bonne que pour les photos posées qui composent ces corpus. Ainsi, même si un détecteur présente de bons résultat théorique, ses performances seraient dégradées si nous l'utilisation sur nos séquences vidéo.

Les méthodes de détection de visage peuvent être classées en quatre catégories [YanKM02], à savoir :

- Les méthodes basées sur la reconnaissance.
- Les méthodes basées sur les caractéristiques invariantes.
- Les méthodes d'appariement de modèles.
- Les méthodes basées sur l'apprentissage de modèles.

Dans ce qui suit, nous présentons la « philosophie » associée à chacune des approches.

#### **4.1. Les méthodes basées sur la connaissance :**

Dans cette catégorie, les méthodes de détection du visage sont développée à partir de règle dérivée de la connaissance que l'on a des visages humains [YanH94], en effet, il est

assez aisé de proposer des règles simples pour décrire les caractéristiques d'un visage et leurs relations. Par exemple, un visage de face apparaît dans une image avec un nez, une bouche et deux yeux symétriques. Les relations entre les caractéristiques peuvent être représentées par leurs positions et leurs distances relatives. Des caractéristiques faciales dans l'image d'entrée sont d'abord extraites, puis des visages candidats sont identifiés en se basant sur des règles prédéfinies [KorP97]. Un procédé de vérification est habituellement appliqué pour réduire les fausses détections. L'un des problèmes de cette approche est la difficulté de traduire la connaissance humaine en règles bien définies. En effet si celles-ci sont trop strictes. Elles peuvent ne pas détecter les visages qui ne satisfont pas toutes les règles. A l'inverse, si elles sont trop « permissives », elles peuvent prendre à tort divers objets de la scène qui les respectent pour les visages. De plus, il est difficile d'étendre cette approche pour détecter des visages dans différentes postures puisqu'il est fastidieux d'énumérer tous les cas possibles. Quoiqu'il en soit, ces méthodes peuvent bien fonctionner sur des scènes simples et où la vue des visages est simple plus moins frontale.

#### **4.2. Les méthodes basées sur des caractéristiques invariantes :**

En réponse aux « points faibles » de l'approche précédente, on essaye ici de trouver des caractéristiques invariantes qui existent quels que soient la posture, le point de vue, et les conditions d'éclairages. On emploie ensuite ces derniers pour localiser des visages [YowC97]. Partant du constat que les humains peuvent sans effort détecter des visages et d'autres objets en différents posture et conditions d'éclairages, on fait l'hypothèse qu'il doit exister des propriétés ou des caractéristiques qui sont invariante malgré les facteurs d'influence précédemment cités. De nombreuses méthodes ont été proposées pour d'abord détecter les caractéristiques faciales qui ensuite indiquent la présence d'un visage [YowC97]. Ces derniers, telles que les sourcils, les yeux, le nez, la bouche, et les cheveux sont généralement extraites en utilisant des détecteurs de contour. En se basant sur ces éléments, un modèle statistique est construit pour décrire leurs relations et vérifier l'existence d'un visage. Cependant, du fait de la nature même de l'information de départ (une image), il demeure, particulièrement difficile de s'affranchir des variations d'éclairage et des phénomènes d'occultation à l'aide des primitives précédente. En effet, les contours de celle-ci peuvent être estompés, alors que les ombres peuvent créer de nombreux artefacts qui compliquent considérablement les tâches de ces algorithmes. Aussi, utilise-t-on des éléments moins sensibles à ces phénomènes. A ce titre, les propriétés colorimétriques de la peau (qu'on appellera ici « couleur de la peau ») ainsi que sa texture ont largement été utilisées [ForF99] [LeeK07] [Sun10]. En effet celle-ci peuvent présenter une invariance suffisante malgré la posture du visage, son orientation ou son expression.

#### **4.3. Les méthodes d'appariement de modèles :**

Dans cette catégories de méthodes, plusieurs modèles standard de visages sont appris et stockés pour décrire ce dernier dans son ensemble ou certaines caractéristique faciales séparément [CraTB92] [LanTC95]. Pour une image d'entrée donnée, les valeurs de corrélation avec les modèles appris sont calculées indépendamment pour le contour du visage, les yeux, le nez et la bouche. La décision relative à la présence d'un visage dans l'image présente est alors déterminée en se basant sur ces valeurs de corrélation. Cette approche a

l'avantage d'être simple à implémenter mais est très peu robuste aux variations d'échelle, de posture et de forme, et surtout plus coûteuse à exécuter.

#### 4.4. Les méthodes basées sur l'apprentissage de modèles :

Dans cette catégorie, et contrairement aux méthodes d'appareillement de modèle ou ceux-ci sont prédéfinis par des experts, les modèles exploités ici sont appris d'exemples d'images [SchK98] [SunP98]. En générale les méthodes basées sur l'apparence se basent sur des techniques d'analyse statistique et d'apprentissage pour trouver les caractéristiques appropriées des images de visage et de « non-visage ». Des caractéristiques apprises sont exprimées sous forme de modèle de distribution ou de fonction discriminantes qui sont employées ensuite pour la détection. Les méthodes de cette catégorie obtiennent généralement les meilleures performances (taux de détection) mais ont l'inconvénient d'être coûteuses en calcul et assez laborieuses à mettre en œuvre.

#### 4.5. Algorithme de Viola et Jones :

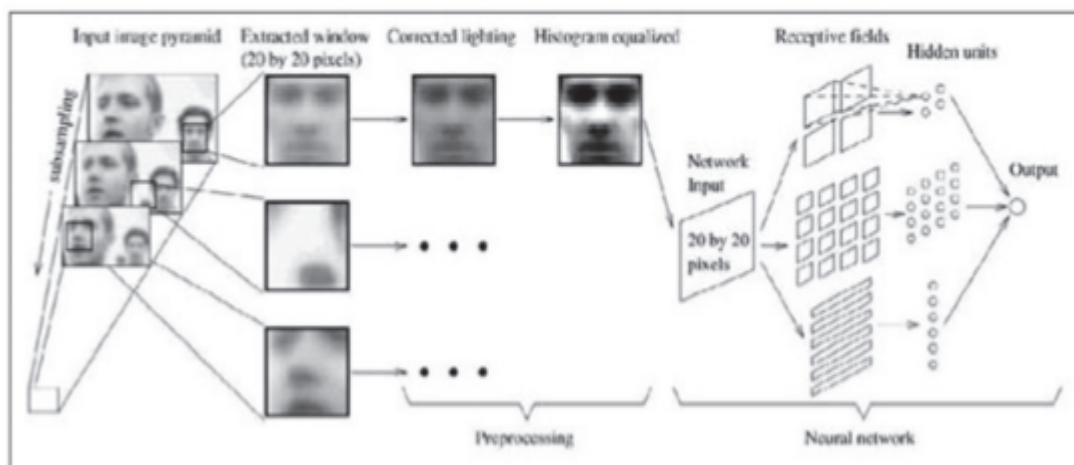
La méthode de Viola et Jones est une méthode de détection d'objet dans une image numérique, proposée par les chercheurs Paul Viola et Michael Jones en 2001. Elle fait partie des toutes premières méthodes capables de détecter efficacement et en temps réel des objets dans une image. Inventée à l'origine pour détecter des visages, elle peut également être utilisée pour détecter d'autres types d'objets comme des voitures ou des avions. La méthode de Viola et Jones l'une des méthodes les plus connues et les plus utilisées, en particulier pour la détection des visages et la détection de personnes. En tant que procédé d'apprentissage supervisé, la méthode de Viola et Jones nécessite de quelques centaines à plusieurs milliers d'exemples de l'objet que l'on souhaite détecter, pour entraîner un classifieur. Une fois son apprentissage réalisé, ce classifieur est utilisé pour détecter la présence éventuelle de l'objet dans une image en parcourant celle-ci de manière exhaustive, à toutes les positions et dans toutes les positions et toutes les tailles possibles.

Considérée comme étant l'une des plus importantes méthodes de détection d'objet, la méthode de Viola et Jones est notamment connue pour avoir introduit plusieurs notions reprises ensuite par de nombreux chercheurs en vision par ordinateur, à l'exemple de la notion d'image intégrale ou de la méthode de classification construite comme une cascade de classifieurs boostés. Cette méthode bénéficie d'une implémentation sous licence BSD dans OpenCV, c'est une bibliothèque très utilisée en vision par ordinateur.

#### 4.6. Réseau de neurones :

Le réseau de neurones est utilisé pour classifier les pixels de l'image, en tant que visage ou non-visage. Dans toute utilisation de réseaux de neurones, il faut définir une topologie du réseau. La topologie de réseau est déterminée par des tests successifs et il n'y a aucune méthode standard à suivre pour définir la meilleure. Un visage se distingue surtout par des yeux, un nez et une bouche. La topologie de base sera donc d'une unité finale fournissant une réponse binaire ou probabiliste. On mettra derrière cette unité les couches cachées du réseau, on appelle notamment cela une topologie de base car le nombre d'unités, leur taille et leur position restent non empiriques et ne peuvent jamais être exactement fixés. Les réseaux

de neurones les plus répandus et les plus simples à la fois restent les perceptrons multicouches (PMC) qui consistent à une succession de 9 couches, interconnectées totalement ou partiellement. L'algorithme d'apprentissage total reviendra à transmettre à tous les PMC, traitant chacun une unité, le résultat attendu. Si l'exemple à apprendre est un visage, la première étape est de transmettre aux PMC les yeux, la bouche et le nez. De là, chaque PMC applique son algorithme d'apprentissage. L'inconvénient de cette approche réside dans le temps de calcul qui ne permet pas souvent de faire des traitements en temps réel. Le schéma suivant explique bien cette approche.



**Figure 5 :** principe de réseau de neurone pour la détection du visage.

## 5. Conclusion :

Dans ce chapitre nous avons présenté c'est quoi le CBVIR est ses domaines d'application. Comme seconde tâche on a vu le système de détection de visage humain ainsi que ses caractéristiques. Au final on a parlé sur quelques méthodes pour la détection de visage, et laquelle de ses méthode et plus performante pour l'accélération de détection de visage en temps réel. D'après les chercheurs la méthode la plus utiliser dans ses derniers temps, et la méthode la plus puissante et celle de Viola et Jones. Cette méthode et implémenté dans OpenCV quand va voir dans le prochaine chapitre.

# Chapitre 4

## 1. Introduction :

De nos jours, les clusters sont souvent reliés par des réseaux longue distance, formant des grilles, qui permettent d'offrir une quantité de ressources très importante aux utilisateurs. MPI est la bibliothèque de communication standard utilisée pour écrire des applications parallèles pour ces grilles. Les utilisateurs souhaitent exécuter sur des grilles les applications qu'ils avaient écrites pour des clusters afin de disposer du plus grand nombre de ressources. Par exemple, des applications comme ray2mesh, une application sismique de projection de rayons dans un maillage 3D de la Terra, ou des applications médicales comme Simri, un simulateur d'imagerie à résonance magnétique en 3D ont été portées sur la grille avec succès. Cependant, MPI a été initialement écrit pour les clusters et ne prend pas en considération les spécificités des grilles. L'exécution des applications MPI pose essentiellement trois problèmes quand on veut les porter sur des grilles de calcul. Premièrement, les implémentations de MPI doivent gérer de manière efficace les liens longue distance entre les différents sites. La forte latence entre les sites est très coûteuse, particulièrement pour les petits messages. Les communications inter-sites prennent plus de temps que les communications intra-sites. Les liens inter-sites offrent des débits plus élevés que les communications intra-sites.

Deuxièmement, les implémentations MPI doivent prendre en compte l'hétérogénéité des différents réseaux rapides présents dans les grilles. Il s'agit d'une part de permettre des communications intra-site (par exemple communiquer entre un réseau Myrinet et un réseau Infiniband) et d'autre part de gérer les communications inter-sites c'est-à-dire entre deux clusters Myrinet séparés par un WAN (Wide Area Network) sur lequel TCP est utilisé.

La détection de visage est un domaine de la vision par ordinateur consistant à détecter un visage humain dans une image numérique. C'est un cas spécifique de détection d'objet, où l'on cherche à détecter la présence et la localisation précise d'un ou plusieurs visages dans une image. C'est l'un des domaines de la vision par ordinateur parmi les plus étudiés, avec de très nombreuses publications, brevet, et de conférences spécialisées. La forte activité de recherche en détection de visage a également permis de faire émerger des méthodes génériques de détection d'objet.

Pour cela, on a opté pour la méthode Haar classificateurs en cascade, à base de fonctionnalités est une méthode de détection d'objet efficace proposé par Paul Viola et Michel Jones. Dans leur document, et est une approche basée sur l'apprentissage de la machine où fonction cascade est formé à partir d'un grand nombre d'images positives et négatives, et ensuite est utilisé pour détecter des objets dans d'autres images et donc elle est utilisé pour la détection du visage ainsi que notre programme est basé sur MapReduce sur Hadoop. Hadoop est une infrastructure de traitement par lots distribuée à grande échelle, idéalement adaptée pour le traitement de données à grande échelle. Hadoop inclut un système de fichiers distribué qui rompt des données en entrée et envoie les fractions des données originales à plusieurs machines pour que les données puissent être traitées en parallèle.

## 2. Structure de l'application (Hadoop) :

### 2.1. hdfs :

HDFS (Hadoop Distributed File System) reprend de nombreux concepts proposés par des systèmes de fichiers classiques comme ext2 pour Linux ou FAT pour Windows. Nous retrouvons donc la notion de blocs (la plus petite unité que l'unité de stockage peut gérer), les

métadonnées qui permettent de retrouver les blocs à partir d'un nom de fichier, les droits ou encore l'arborescence des répertoires.

Toutefois, HDFS se démarque d'un système de fichiers classique pour les principales raisons suivantes :

- ✚ HDFS n'est pas solidaire du noyau du système d'exploitation. Il assure une portabilité et peut être déployé sur différents systèmes d'exploitation. Un des inconvénients est de devoir solliciter une application externe pour monter une unité de disque HDFS.
- ✚ HDFS est un système distribué. Sur un système classique, la taille du disque est généralement considérée comme la limite globale d'utilisation. Dans un système distribué comme HDFS, chaque nœud d'un cluster correspond à un sous-ensemble du volume global de données du cluster. Pour augmenter ce volume global, il suffira d'ajouter de nouveaux nœuds. On retrouvera également dans HDFS, un service central appelé Namenode qui aura la tâche de gérer les métadonnées.
- ✚ HDFS utilise des tailles de blocs largement supérieures à ceux des systèmes classiques. Par défaut, la taille est fixée à 64 Mo. Il est toutefois possible de monter à 128 Mo, 256 Mo, 512 Mo voire 1 Go. Alors que sur des systèmes classiques, la taille est généralement de 4 Ko, l'intérêt de fournir des tailles plus grandes permet de réduire le temps d'accès à un bloc. Notez que si la taille du fichier est inférieure à la taille d'un bloc, le fichier n'occupera pas la taille totale de ce bloc.
- ✚ HDFS fournit un système de réplication des blocs dont le nombre de répliqués est configurable. Pendant la phase d'écriture, chaque bloc correspondant au fichier est répliqué sur plusieurs nœuds. Pour la phase de lecture, si un bloc est indisponible sur un nœud, des copies de ce bloc seront disponibles sur d'autres nœuds.

## 2.2. Namenode :

Un Namenode est un service central (généralement appelé aussi maître) qui s'occupe de gérer l'état du système de fichiers. Il maintient l'arborescence du système de fichiers et les métadonnées de l'ensemble des fichiers et répertoires d'un système Hadoop. Le Namenode a une connaissance des Datanodes (étudiés juste après) dans lesquels les blocs sont stockés. Ainsi, quand un client sollicite Hadoop pour récupérer un fichier, c'est via le Namenode que l'information est extraite. Ce Namenode va indiquer au client quels sont les Datanodes qui contiennent les blocs. Il ne reste plus au client qu'à récupérer les blocs souhaités.

Toutes ces métadonnées, hormis la position des blocs dans les Datanodes, sont stockées physiquement sur le disque système dans deux types de fichiers spécifiques `edits_xxx` et `fsimage_xxx`.

La connaissance de la position des blocs dans les Datanodes est reconstruite à chaque démarrage du Namenode dans un mode appelé `safe mode`. Pendant le `safe mode`, l'écriture sur HDFS est impossible, le Namenode charge les fichiers `edits_xxx` et `fsimage_xxx` et attend le retour des Datanodes sur la position des blocs. Une fois toutes les opérations réalisées, le `safe mode` est relâché et l'accès en écriture est de nouveau autorisé. Soyez patient sur la durée du `safe mode`. Celui-ci peut être très long si vous avez beaucoup de fichiers à traiter.

Comme vous l'aurez remarqué, le Namenode charge tout en mémoire. Cela devient donc problématique si vous avez énormément de petits fichiers à gérer. D'après la

documentation officielle de [Cloudera](#), chaque fichier, répertoire et bloc dans HDFS est représenté comme un objet mémoire et occupe 150 octets. Si, par exemple, vous avez 10 millions de fichiers à gérer, le Namenode devra disposer d'un minimum de 1,5 Go de mémoire. C'est donc un point important à prendre en compte lors du dimensionnement de votre cluster. Le Namenode est relativement gourmand en mémoire.

### 2.3. Namenode secondaire :

Le Namenode dans l'architecture Hadoop est un point unique de défaillance (Single Point of Failure en anglais). Si ce service est arrêté, il n'y a pas moyen de pouvoir extraire les blocs d'un fichier donné. Pour répondre à cette problématique, un Namenode secondaire appelé Secondary Namenode a été mis en place dans l'architecture Hadoop. Son fonctionnement est relativement simple puisque le Namenode secondaire vérifie périodiquement l'état du Namenode principal et copie les métadonnées via les fichiers `edits_xxx` et `fsimage_xxx`. Si le Namenode principal est indisponible, le Namenode secondaire prend sa place.

### 2.4. Datanode :

Précédemment, nous avons vu qu'un Datanode contient les blocs de données. Les Datanodes sont sous les ordres du Namenode et sont surnommés les Workers. Ils sont donc sollicités par les Namenodes lors des opérations de lecture et d'écriture. En lecture, les Datanodes vont transmettre au client les blocs correspondant au fichier à transmettre. En écriture, les Datanodes vont retourner l'emplacement des blocs fraîchement créés. Les Datanodes sont également sollicités lors de l'initialisation du Namenode et aussi de manière périodique, afin de retourner la liste des blocs stockés.

### 2.5. Mapreduce :

MapReduce adresse deux choses. La première concerne le modèle de programmation MapReduce, étudié dans cette section. La seconde concerne le framework d'implémentation MapReduce, étudié dans le prochain article. Pour ce dernier, nous nous focaliserons sur les différentes API proposées par Apache Hadoop pour développer un programme MapReduce à partir du langage Java.

Le modèle de programmation fournit un cadre à un développeur afin d'écrire une fonction de Map et de Reduce. Tout l'intérêt de ce modèle de programmation est de simplifier la vie du développeur. Ainsi, ce développeur n'a pas à se soucier du travail de parallélisation et de distribution du travail, de récupération des données sur HDFS, de développements spécifiques à la couche réseaux pour la communication entre les nœuds, ou d'adapter son développement en fonction de l'évolution de la montée en charge (scalabilité horizontale, par exemple). Ainsi, le modèle de programmation permet au développeur de ne s'intéresser qu'à la partie algorithmique. Il transmet alors son programme MapReduce développé dans un langage de programmation au framework Hadoop pour l'exécution.

Autre chose avant de continuer, le terme de « job » MapReduce est couramment utilisé dans la littérature. Celui-ci concerne une unité de travail que le client souhaite réaliser. Cette unité est constituée de données d'entrée (contenues dans HDFS), d'un programme MapReduce (implémentation des fonctions map et reduce) et de paramètres d'exécution. Hadoop exécute ce job en le subdivisant en deux tâches : les tâches de Map et les tâches de Reduce.

### 3. Algorithme de détection de visage utilisée

#### 3.1. Le détecteur de Viola et Jones :

Paul Viola et Michael Jones ont publiés la première version de leur papier le 13 juillet 2001. Leurs travaux sont inspirés de ceux de Papageorgiou, Oren et Poggio en 1998, qui utilisent eux-même les ondelettes de Haar. Ils utilisent également les travaux de Paul Viola et Kinh Tieu, concernant l'utilisation de techniques de boost (inspiré d'AdaBoost). La méthode Viola et Jones est considérée comme l'une des plus efficaces en détection de visage, et est devenue un standard dans le domaine.

Le détecteur de Viola et Jones opère sur l'information de luminance uniquement. Il génère une boîte englobant rectangulaire du visage très faible lorsque le visage est de face ou légèrement de profil. Cependant il est défaillant en présence de fortes occultations.

#### 3.2. Fonctionnement des détecteurs du Viola et Jones :

Cette méthode permet d'analyser une image, sans avoir besoin d'étudier chaque pixel la constituant. En effet, la notion d'image intégrale permet de définir plusieurs zones rectangulaires au sein d'une image. L'intérêt de cette technique réside dans le fait qu'elle offre la possibilité d'accéder à la valeur des autres zones à gauche et au-dessus de la zone sur laquelle nous sommes.

Ces zones permettent de créer des caractéristiques pseudo Haar, qui sont en fait des masques permettant de plusieurs patterns :

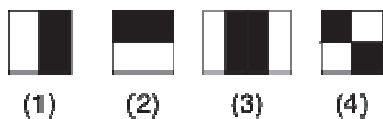


Figure 1 : exemple de caractéristiques pseudo-Haar.

Ces caractéristiques permettent de détecter des motifs. Par exemple, la reconnaissance des visages est rendue possible par :

- La variation de l'intensité de la lumière entre les yeux et le nez.
- Et la variation de l'intensité de la lumière entre les yeux et les pommettes.

La méthode repose donc sur l'utilisation de ce caractère pseudo Haar et des images intégrales, améliorant ainsi la vitesse de traitement. Deux autres mécanismes interviennent : le classifieur et le boosting.

#### 3.3. Apprentissage du Viola et Jones :

L'apprentissage consiste à analyser un très grand nombre d'images positives et négatives.

Plusieurs milliers d'exemples sont nécessaires, et leur analyse prend plusieurs semaines. L'apprentissage comprend :

- Le calcul des caractéristiques pseudo Haar sont positifs ou négatifs.
- L'entraînement de la cascade : pour chaque étage, on crée un classifieur fort, par ajout successif de classifieur faibles entraînés sur la même caractéristique. On ajoute ces

classificateurs faibles jusqu'à obtenir un classifieur fort possédant la performance souhaitée.

### 3.4. Algorithme AdaBoost de Viola et Jones pour la détection de visages :

L'algorithme AdaBoost est un algorithme plus sophistiqué pour le boosting de la combinaison des hypothèses. Il est appelé Adaptive dans le sens que les exemples qui sont mal classifiés doivent obtenir les poids les plus élevés dans la prochaine itération, par exemple, les exemples près de la frontière de décision sont plus difficiles à classer et à cet effet obtenir des poids élevés dans l'ensemble d'entrée après les premières itérations. Viola et Jones ont proposé un algorithme AdaBoost avec une détection en cascade et lui appliqué avec succès à la détection de visages. Dans AdaBoost de Viola, le classifieur faible a été établi par une caractéristique rectangulaire, l'apprentissage par AdaBoost est adopté pour combiner ces classificateurs faibles. Par conséquent, dans un certain sens, dans AdaBoost de Viola, le classifieur faible est un peu équivalent à une caractéristique faible. Pour le problème spécifique de la détection de visages.

#### 3.4.1. Détection en Cascade :

La cascade est la combinaison successive des classificateurs forts, dont la complexité est croissante tout au long de la structure. Au départ, les classificateurs simples, éliminent la plupart des fausses alarmes, avant que les classificateurs plus complexes ne soient utilisés pour éliminer des fenêtres plus difficiles. L'attention du système se concentre, au fur et à mesure des étages de la cascade, sur des zones de plus en plus réduites et pertinentes. Viola et Jones consiste en l'introduction d'un apprentissage basé sur la méthode de dopage qui permet d'obtenir une fonction de classification avec une architecture en cascade Attentionnelle. L'architecture de cascade est très efficace, l'image d'entrée est passée par une séquence de classificateurs. Pour qu'une image soit classifiée en tant que visage, elle doit passer par tous les classificateurs. La classification s'arrête immédiatement si un quelconque de ces étapes échoue.

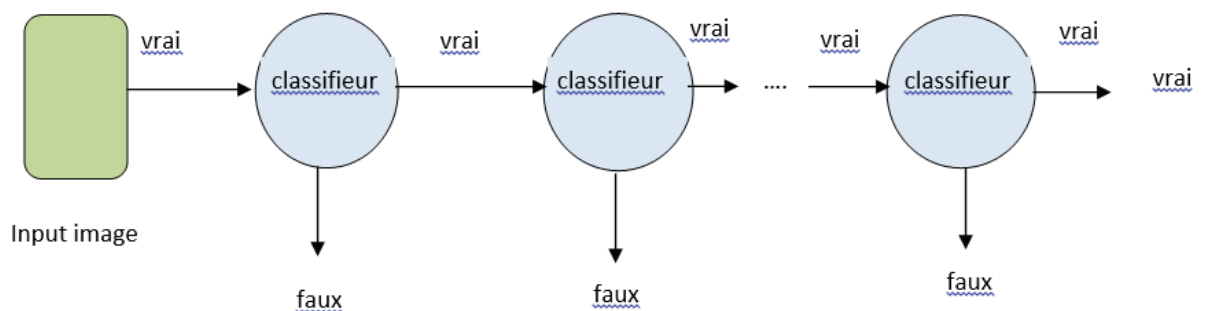


Figure 2 : architecture de cascade.

La décision « fausse » stoppe le calcul et impose le détecteur de retourner « faux ». Une décision « vraie » passe l'image d'entrée au prochain classifieur dans la cascade. Si tous les classificateurs votent vrai alors l'entrée est classifiée comme un vrai exemple. Si n'importe quel classifieur vote « faux » alors le calcul se stoppe et l'entrée sera classifiée comme fausse.

## 4. Algorithme de détection de visage avec MapReduce :

### 4.1. Algorithme de Map :

Map (clé : nom de la frame en texte, valeur : image en binaire)

1. Récupérer l'image
2. Détecté les visages présent dans l'image
3. Produire une sortie pour le reducer (clé : nom de l'image comme text, value: coordonnée du visage comme un rectangle aussi comme text)

#### 4.2. Algorithme de Reduce :

reduce (clé : nom de l'image comme text, value: coordonnée du visage comme un rectangle aussi comme text)

1. regrouper les resultats
2. ecrire le resultat dans le HDFS (clé : nom de l'image comme text, value: coordonnée du visage comme un rectangle aussi comme text)

### 5. Réalisation et environnement

#### 5.1. Dans eclipse :



Figure 3 : l'image avant l'exécution du programme java dans eclipse.

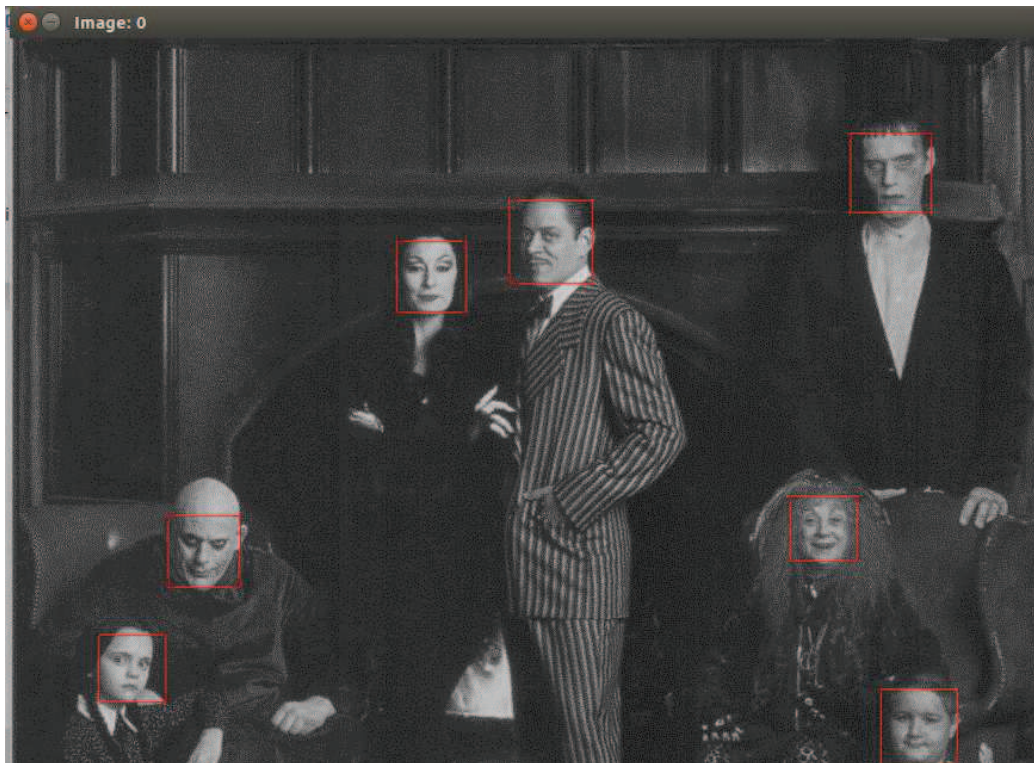


Figure 4 : l'image après l'exécution de programme java dans eclipse.

## 5.2. Dans Hadoop :

```
Deleted /tracker/output
m1@m1-VirtualBox:/usr/local/hipi$ hadoop jar faceDetection/faceDetection.jar /tr
acker/sozzo.seq /tracker/output
16/07/12 17:33:23 WARN util.NativeCodeLoader: Unable to load native-hadoop libra
```

Figure 5 : lancement d'exécution le fichier .jar dans Hadoop.

```
m1@m1-VirtualBox:/usr/local/hipi$ hadoop fs -cat /tracker/output/part-r-00000
16/07/12 18:07:09 WARN util.NativeCodeLoader: Unable to load native-hadoop libra
ry for your platform... using builtin-java classes where applicable
sozzo/0005.jpg 196.0,523.0,64.0,64.0;726.0,558.0,45.0,45.0;
sozzo/0007.jpg 33.0,473.0,121.0,121.0;
sozzo/0009.jpg 793.0,558.0,73.0,73.0;
sozzo/0012.jpg 714.0,568.0,69.0,69.0;
sozzo/0014.jpg 713.0,548.0,57.0,57.0;
```

Figure 6 : le résultat de l'exécution de fichier .jar (MapReduce).

## 6. Plateforme utilisé :

### 6.1. VirtualBox

VirtualBox ou machine virtuel est un logiciel de virtualisation de système d'exploitation. En utilisant les ressources matérielles de l'ordinateur (système hôte), VirtualBox permet la création d'un ou plusieurs ordinateurs virtuels dans lesquelles s'installent d'autre système d'exploitation (systèmes invités). Les systèmes invités fonctionnent en même temps que le système hôte, mais seul ce dernier a accès directement au véritable matériel de l'ordinateur.

Les systèmes invités exploitent du matériel générique, simulé par un « faux ordinateur », (machine virtuelle) créé par VirtualBox. VirtualBox permet de faire fonctionner plus d'un système d'exploitation en même temps en toute sécurité. En effet, les systèmes invités n'interagissent pas directement avec le système hôte, et n'interagissent pas entre eux. Le champ d'action des systèmes invités est confirmé, limité à leur propre machine virtuelle.

## 6.2. Apache Hadoop 2.7.1

Apache Hadoop 2.7.1 est une version mineure dans la ligne 2.xy de libération, se fondant sur la version précédente 2.7.0. Ceci est la prochaine version stable après Apache Hadoop 2.6.x.

Voici un bref aperçu des principales caractéristiques et améliorations depuis 2.6.x.

- ✓ Notes importantes
  - ✚ Cette version goute le support pour JDK6 exécution et fonctionne avec JDK7+ seulement
- ✓ Commun
  - ✚ Support de Windows Azure Storage-Blob comme un système de fichiers dans Hadoop.
- ✓ HDFS
  - ✚ Prise en charge de fichier truncate.
  - ✚ Prise en charge des quotas par type de stockage
  - ✚ Support des fichiers avec des blocs de longueur variable
- ✓ FIL
  - ✚ Faire enfichables d'autorisation de YARN
  - ✚ Automatique partagée, la mise en cache globale des ressources YARN localisée (bêta)
- ✓ Mapreduce
  - ✚ Possibilité de limiter l'exécution Carte / Réduire les tâches d'un emploi
  - ✚ Accélérer FileOutputStream pour de très gros travaux avec de nombreux fichiers de sortie.

## 6.3. Ubuntu 14.04

**Ubuntu 14.04 LTS** (nom de code : the trusty tahr, le tahr sûr) est la vingtième version d'Ubuntu. Son développement a débuté en octobre 2013 et s'est échelonné sur une période de six mois. La version stable, finale d'Ubuntu 14.04.1 LTS (long terme support = support disponible pendant une longue durée) est sortie le 17 avril 2014 et sera supportée jusqu'en avril 2019.

### 6.3.1.1. Objectif et développement

- ✚ Du côté des postes de travail, Ubuntu 14.04 LTS sera davantage une version de maintenance. Elle profitera encore une fois d'une équipe entièrement dédiée à l'amélioration de la qualité.
- ✚ Du côté des plateformes mobiles, l'ensemble est encore tout neuf. La marche vers la convergence poursuivra.

- ✚ Du côté des serveurs, Ubuntu 14.04 LTS sera une base solide pour le déploiement de solutions en informatique pour les entreprises avec les trois prochaines versions d'OpenStack.

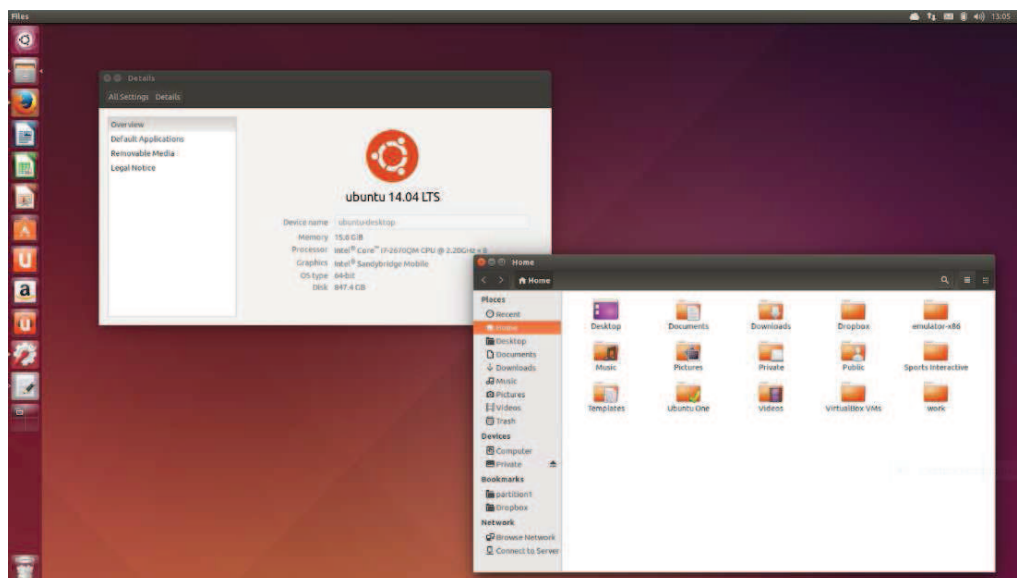


Figure 6: premier interface a l'ouverture d'Ubuntu.

#### 6.4.OpenImaj

OpenImaj est un primé ensemble de bibliothèques et d'outils pour l'analyse de contenu multimédia et la génération de contenu. OpenImaj est très large et contient tout de la vision state-of-the-art (par exemple EIPD descripteurs, la détection de la région saillante, détection de visage, etc.) et le regroupement des données avancées, par le biais d'un logiciel qui effectue une analyse sur le contenu, la présentation et la structure des pages web.

OpenImaj fournit une suite d'outils de ligne de commande qui permettent d'accéder facilement aux fonctionnalités de base dans la bibliothèque OpenImaj. Alors que nouvelles fonctionnalités sont ajoutées à la bibliothèque, ils vont probablement être exposés comme des outils afin de permettre l'intégration et l'essai des outils et des techniques mises en œuvre facile. Les outils offrent la possibilité de construire un pipeline de récupération d'image entière, qui est :

- ✓ Image téléchargé
- ✓ Extraction de caractéristique
- ✓ La génération de dictionnaire
- ✓ Quantification et indexation

Il existe également des versions Hadoop de ces outils qui permettent des versions évolutives de chaque tâche. Nous fournissons également des outils simples d'aide pour fonction/visualisation des images.

## 7. conclusion

Dans ce chapitre, nous avons réalisé une petite application pour détecter le visage, comme on a réalisé un programme java, à base d'openImaj et de la méthode Haar, qui est utilisé pour détecter les objets. Le programme est tester et exécuter sur eclipse, et on a fait l'algorithme à base de MapReduce qu'on a exécuté sur Hadoop puis charger un fichier de l'extension .seq pour qu'il puisse avoir l'ensemble des images d'une vidéo pour lequel test et donne un résultat sous forme de vecteur qui spécifier les caractéristique de visage détecter.

# Conclusion générale

Dans notre travail nous avons étudié la détection du visage basé sur MapReduce dans le domaine de systèmes de vidéosurveillance. Il existe plusieurs méthodes de détection tels que les méthodes basé sur la connaissance est l'un des problèmes de cette approche est la difficulté de traduire la connaissance humain en règles bien définis, et il on a d'autres méthodes tels que les méthodes basé sur des caractéristique invariantes, et les méthodes d'appariement de modèles et que dans cette méthodes a l'avantage d'être simple à implémenter, mais est très peu robuste aux variations d'échelle, de posture et de forme et surtout plus couteuse à exécuter. Et une autre méthode basé sur l'apprentissage de modèle, la méthode la plus utilisé celle de Jones et Viola.

Pour notre application on a implémenté un programme basé sur mapreduce qui s'exécute dans le framework Hadoop, tels que Hadoop est un programme évolutif qui rencontre actuellement trois difficultés majeures que les fournisseurs de solutions de stockage en Cloud privé pourraient aplanir, voire éradiqué :

HDFS fournit un système de fichiers extrêmement résistant et bien documenté. Mielleusement, son NameNode unique est un point de défaillance identifié qui réduit la disponibilité de la solution. Le NameNode coordonne l'accès aux données dans le système de fichiers. Pour les clusters Hadoop qui recourent à des charges interactives (HBase), à l'extraction, la transformation et le chargement en temps réel ou à des processus métier traités en lots, une panne du NameNode HDFS peut poser un sérieux problème. Lorsqu'elle produit, c'est l'immobilisation : les utilisateurs sont mécontents et la productivité s'en ressent. La communauté Hadoop et Apache travaillent aujourd'hui d'arrache-pied pour développer un NameNode haute disponibilité. Le produit devrait être prêt avec Hadoop 2.0. Dans le même temps, plusieurs fournisseurs de solutions de stockage en cloud privé, tels que NetApp avec FAS et V-Series, EMC avec Isilon, et Cleversafe et sa technologie Dispersed Storage, intègrent à leurs produits de stockage un correctif pour NameNode.

Le deuxième inconvénient d'Hadoop auquel le stockage en cloud privé peut apporter une réponse est légèrement pire que le premier. HDFS effectue au moins deux copies des données, ou trois au total, pour garantir la résistance souhaitée. Ce qui implique de consommer trois fois plus d'espace de stockage. Même en utilisant l'espace de stockage de serveurs économiques, cela fait beaucoup : pour chaque péta-octet de données réelles, 3 Po d'espace sont consommés. Tout ce stockage utilise de l'espace sur les racks, de l'espace au sol, et surtout de l'électricité pour l'alimentation et le refroidissement. Cleversafe a résolu la question en fournissant une interface HDFS qui élimine les copies multiples grâce à l'utilisation d'un code d'effacement du stockage réparti. Ce stockage réparti permet de diminuer de 60 % l'espace utilisé, tout en garantissant une fiabilité supérieure à celle qu'offre la technologie HDFS standard.

Se pose ensuite la question du déplacement des données vers un cluster Hadoop. Pour être traitées, les données doivent migrer vers le cluster Hadoop. Ce n'est pas une sinécure et l'opération peut s'avérer plus ou moins chronophage, selon la quantité de données à traiter et

à analyser. Et il s'agit d'une tâche continue. La solution EMC Isilon propose un raccourci. Elle peut représenter des données NFS ou CIFS (SMB1 ou SMB2) placées dans un cluster de stockage Isilon sous la forme de données HDFS, éliminant ainsi le recours à une migration. À l'inverse, elle peut également représenter les données HDFS sous forme de données NFS ou CIFS pour permettre leur exploitation hors du cluster Hadoop.

# Webliografie

# Webliographie

## Chapitre 1 :

1. [1] : pdf : les différents matériels de vidéosurveillance, [www.ssnict.net/wp-content/themes/ssn/pdf/video-srf/pdf](http://www.ssnict.net/wp-content/themes/ssn/pdf/video-srf/pdf)
2. [2] : la video de surveillance intelligent : promesse et defi rapporte de veille technologique et commerciale :  
[https://www.crim.ca/publications/2009/documents/plein-texte/VIS-GouVal\\_09-03.pdf](https://www.crim.ca/publications/2009/documents/plein-texte/VIS-GouVal_09-03.pdf)
3. [3] PDF : assistance a l'indexation vidéo par analyse du mouvement : theses.insa-lyon.fr/publication/2002ISAL0015/these.pdf
4. [4] : pdf : indexation et recherche de video pour la vidéosurveillance,  
[https://www.sop.inria.fr/teams/pulsar/publications/2009/le %2009/LLePhDthesis.pdf](https://www.sop.inria.fr/teams/pulsar/publications/2009/le%2009/LLePhDthesis.pdf)

## Chapitre 2

1. [1] : [www.tutorialspoint.com](http://www.tutorialspoint.com), hadoop/hadoop-mapreduce.html
2. [2] : pdf : hugfrs/2013 mapreduce.130527194811.phpapp02.pdf
3. [3] : <https://fr.wikipedia.org/wiki/mapreduce>
4. [4] : pdf : hindman-bootcamp-2011, <http://parlab.eecs.berkeley.edu/site/all/parlab/files/hindman-bootcamp-2011-pdf>
5. [5] : pdf : hadoop/BigData : [www.mbds-fr.org/wp-content/uploads/2014/09/mooc-fun-big-date-semaine-9-hadoop.pdf](http://www.mbds-fr.org/wp-content/uploads/2014/09/mooc-fun-big-date-semaine-9-hadoop.pdf)
6. [6] : <http://fr.wikipedia.org/wiki/hadoop>.
7. [7] : <http://fr.wikipedia.org/wiki/hadoop>.
8. [8] : pdf : hadoopBigData : [www.mbds-fr.org/wp-content/uploads/2014/09/mooc-fun-big-data-semaine-9-hadoop.pdf](http://www.mbds-fr.org/wp-content/uploads/2014/09/mooc-fun-big-data-semaine-9-hadoop.pdf)

## Chapitre 3

1. [1]:[Indexation et recherche de plans vidéo par le contenu sémantique, pdf, HAL Id: pastel-00001298 <https://pastel.archives-ouvertes.fr/pastel-00001298> Submitted on 1 Jul 2005]
2. [2]: [travail de fin d'études : implémentation d'une méthode de détection et suivi de visage en temps réel]
3. [3] :[F:\systeme de détection de visage\Systeme de détection Wikipédia.mht].
4. [4] : [Mémoire Online - Identification des personnes par reconnaissance de visage pour la sécurité d'une institution bancaire - Serge KOMANDA BASEMA].
5. [5] : [google Wikipédia indexation de la vidéo par le costume Gael\_Jaffre.pdf].
6. [6] : [Memoire Online - Identification des personnes par reconnaissance de visage pour la sécurité d'une institution bancaire - Serge KOMANDA BASEMA]
7. [7] : [<https://ori-nuxeo-univ-lille1.fr>, Lyes Hamoudi :application de techniques d'apprentissage pour la détection et la reconnaissance d'individus]