

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

UNIVERSITE MOULOUD MAMMARI 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 : **Conduite de Projet Informatique**

Présenté par

**Lebib Souhila
Ait Menguellet Farida**

Thème

Détection de plaque d'immatriculation de véhicule basée sur MapReduce dans une séquence Vidéo

Mémoire soutenu publiquement le 14/07/2016 devant le jury composé de :

Président : M Habet

Encadreur : Mr Dib Ahmed

Co-Encadreur : M Prénom NOM

Examineur : M Chebouba

Examineur : M Kerbiche

Remerciements

Avant tous on remercie dieu le tout puissant de nous avoir donné la santé, la force et le courage mais surtout la volonté d'entamer et de terminer ce mémoire.

On voudrais exprimer notre profonde et sincère gratitude à Mr Ahmed Dib qui nous a proposé ce sujet et encadré, pour le temps qu'il nous a consacré ainsi que la compréhension pour sa patience, et surtout pour ses remarques et ses conseils, sa disponibilité et la bienveillance dont il a toujours fait preuve durant ce travail.

Nos profonds remerciements vont également à nos chers familles et les personnes qui nous ont aidés et soutenue durant tout le long de notre cursus.

On voudrait exprimer notre reconnaissance envers les amis et collègues qui nous ont apporté leur support moral et intellectuel tout au long de notre démarche.

On remercie tous les consultants et internautes rencontrés lors des recherches effectuées et qui ont acceptés de répondre à nos questions avec gentillesse.

Nous remercions chaleureusement les membres du jury pour l'honneur qu'ils nous font en acceptant d'évaluer ce mémoire de fin d'études.

MERCI

Introduction générale

Dans nos jours, le stockage de grands volumes de données est devenu possible et abordable. Parallèlement, la puissance de calcul des microprocesseurs a décuplé, et les caméras numériques sont devenues extrêmement performantes pour un coût de plus en plus bas. Ainsi, depuis que le traitement des données vidéo en temps réel est devenu sérieusement envisageable, pour des problématiques aussi diverses que l'analyse statistique de la fréquentation d'un lieu, la sécurisation de l'accès à des bâtiments, la surveillance routière, ou encore la facturation des véhicules aux péages des autoroutes, les industriels proposent de plus en plus de solutions techniques basées sur l'acquisition numérique de séquences vidéo, alors qu'auparavant, des solutions plus mécaniques auraient été privilégiées.

Le domaine de la vidéo et l'image numérique est un domaine en pleine expansion avec l'explosion d'Internet et aussi le développement à grande échelle de l'image numérique.

Ce mémoire s'inscrit dans le contexte de la détection du véhicule à partir de séquences vidéo qui sera décomposé en une séquence d'images, une des préoccupations majeures dans le domaine de la vision par ordinateur. Les domaines d'application pour les systèmes de vision sont nombreux. On peut citer notamment la vidéo surveillance, la recherche et l'indexation automatique de vidéos ou encore le suivi d'un véhicule. Cette tâche reste problématique étant donné les grandes variations dans la manière de capturer le véhicule, l'apparence de la plaque d'immatriculation dans ce dernier et les variations des conditions d'acquisition de la capture. L'objectif principal de ce travail est de proposer une méthode de détection de plaques d'immatriculation de véhicule la plus efficace qui soit robuste aux différents facteurs de variabilité.

Les représentations fondées sur les points d'intérêt ont montré leur efficacité dans les travaux de l'art, elles ont été généralement couplées avec des méthodes de classification globales étant donné que ces primitives sont temporellement et spatialement désordonnées. Les travaux les plus récents atteignent des performances élevées en modélisant le contexte spatio-temporel des points d'intérêt, par exemple en encodant leur voisinage sur plusieurs échelles. Dans ce mémoire, nous proposons une méthode de détection de plaques d'immatriculation fondée sur OpenCv.

Afin de bien mener notre modeste travail, nous l'avons réparti en quatre chapitres,

Le premier chapitre est une représentation généraliste sur les Big data, Hadoop, MapReduce, le fonctionnement de MapReduce et les domaines d'application qui a pour but d'introduire le sujet, et en particulier le vocabulaire et les notions nécessaires à la compréhension de la suite. Le lecteur qui est déjà familier avec Hadoop et MapReduce va enrichir plus sa connaissance dans le domaine.

Le second chapitre est une introduction de la découverte d'information à base de contenu multimédia qui contient l'ensemble des notions dédiées au traitement d'image et de la vidéo, Content Based Video Indexing and Retrieval (CBVIR) et ses domaines d'application.

Introduction générale

Le troisième chapitre constitue la détection d'immatriculation de véhicule: il présente l'état de l'art de notre travail. Il contient les domaines utilisant la lecture des plaques d'immatriculation de véhicule, les méthodes de détection des plaques d'immatriculations en différents niveaux de traitement d'images, et le développement du processus de l'accélération de la détection de plaque à grande échelle.

Le quatrième chapitre peut être vu comme le cas pratique de notre travail, il constitue la Détection d'immatriculation de véhicule basée sur l'algorithme MapReduce. Il contient la présentation du programme principal, qui donne un autre éclairage sur le sujet, l'architecture de notre application et la démarche de notre méthode, la description détaillée du programme principal que nous avons implémenté et des résultats que nous avons obtenus.

Introduction :

Chaque jour 2,5 trillions d'octets de données se produit. A tel point que 90% des données dans le monde ont été créées au cours des deux dernières années seulement. Ces données proviennent de partout : de capteurs utilisés pour collecter les informations climatiques, de messages sur les sites de médias sociaux, d'images numériques et de vidéos publiées en ligne, d'enregistrements transactionnels d'achats en ligne et de signaux GPS de téléphones mobiles. Pour faire face à l'explosion du volume des données, un nouveau domaine technologique a vu le jour : les Big Data inventés par les géants du web, ces solutions sont dessinées pour offrir un accès en temps réel à des bases de données géantes.

La multiplication des données, dans un volume toujours plus important, et leurs traitements qui engendrent les problématiques posées sur toutes les nouvelles possibilités d'usages qui en découlent sont couvertes par l'expression « Big Data ». Hadoop est au cœur de ces problématiques; c'est (de très loin) le logiciel/*framework* le plus utilisé pour y répondre, il n'utilise aucun principe foncièrement nouveau, il offre en revanche une très forte simplicité et souplesse de déploiement inconnues jusqu'à présent pour l'exécution facile de tâches parallèles variées. Grâce à Hadoop, même des structures limitées en taille/ressources (moyens d'existence) peuvent facilement avoir accès à de fortes capacités de calcul: déploiement à bas coût de *clusters* en interne ou location de temps d'exécution viales services de *cloudcomputing*. Il propose un système de stockage distribué via son système de fichier HDFS qui offre la possibilité de stocker la donnée en la dupliquant. D'autre part, Hadoop fournit un système d'analyse des données appelé MapReduce. Ce dernier officie sur le système de fichiers HDFS pour réaliser des traitements sur des gros volumes de données.

1.1. Big Data ou volumes massifs de données:

Le Big Data est un concept très difficile à définir avec précision, puisque la notion elle-même de Big en termes de volumétrie des données varie d'une entreprise à l'autre. Il n'est pas défini par un ensemble de technologies, bien au contraire, il définit une catégorie de techniques et de technologies. Il s'agit d'un domaine émergent et alors que le but est de chercher à apprendre comment implémenter ce nouveau paradigme et en exploiter la valeur, la définition se transforme. Pourtant, si celle-ci peut s'avérer ambiguë, de nombreux experts sont convaincus que des secteurs entiers de l'industrie et des marchés seront atteints et d'autres créés, au fur et à mesure que ces capacités permettront la réalisation de nouveaux produits et fonctionnalités qui étaient inimaginables auparavant.

Comme l'expression l'indique, le Big Data se caractérise par la taille ou la volumétrie des informations. Mais d'autres attributs, notamment la vitesse et le type de données, sont aussi à considérer. En ce qui concerne le type, le Big Data est souvent rattaché à du contenu non structuré ou semi-structuré, ce qui peut représenter un défi pour les environnements classiques de stockage relationnel et de calcul. Les données non structurées et semi-structurées sont partout : contenu web, poststwitter ou commentaires client en format libre. Par vitesse on entend la rapidité avec laquelle les informations sont créées. Grâce à ces nouvelles technologies, il est maintenant possible d'analyser et d'utiliser l'importante masse de données fournie par les fichiers log des sites web, l'analyse d'opinions des réseaux sociaux, et même les vidéos en streaming et les capteurs environnementaux.

Chapitre I : Hadoop

Afin d'avoir un aperçu de la complexité engendrée par la volumétrie, la vitesse et le type de données, il suffit de considérer les cas suivants :

- ✓ Walmart¹ traite chaque heure plus d'un million de transactions clients, qui sont importées vers des bases de données dont le contenu est estimé à plus de 2,5 petabytes- à savoir 167 fois l'équivalent des informations réunies dans tous les livres de la Bibliothèque du Congrès américain.
- ✓ Facebook gère 40 milliards de photos à partir de sa base d'utilisateurs.
- ✓ Décoder le génome humain a demandé initialement 10 ans de travail ; cette tâche peut maintenant être effectuée en une semaine.
- ✓ La distribution Hadoop par Hortonworks gère à présent plus de 42,000 machines Yahoo! traitant des millions de requêtes par jour.

I.2. Unités de mesure du Big Data : [2]

1 ko = 1 kilooctet = mille octets ;
1Mo = 1 mégaoctet = un million d'octets ;
1Go = 1 gigaoctet = un milliard d'octets ;
1 To = 1 téraoctet = un billion d'octets ;
1 Po = 1 pétaoctet = un billion d'octets ;
1 Eo = 1 exaoctet = un trillion d'octets ;
1 Zo = 1 zettaoctet = un trilliard d'octets.

I.3. Utilisation des Big Data :

Dans une même entreprise, plusieurs départements peuvent être concernés par la mise en place et l'utilisation du Big Data : informatique, commercial, marketing... Les services marketing font le plus appel au Big Data. Ils sont considérés comme précurseurs dans la mise en place de nouvelles stratégies.

A travers ces services, le Big Data répond à plusieurs enjeux pour l'entreprise :

- Améliorer l'expérience client,
- Mieux comprendre les comportements des prospects et des clients,
- Anticiper les besoins et adapter les campagnes marketing.

La mise en place de nouveaux processus liés au Big Data peut aussi permettre l'amélioration de la chaîne logistique, des mécanismes de décision et un avantage concurrentiel significatif.

Quelques exemples d'utilisation du Big Data :

- L'analyse prédictive, c'est-à-dire l'adaptation d'un message marketing à une action probable du consommateur. Exemple : un message publicitaire pour un hôtel à Paris lorsque l'internaute cherche un trajet vers Paris,
- Le marketing automation, c'est-à-dire le marketing automatisé. Exemple : l'envoi d'une publicité selon la géo-localisation ou la date d'anniversaire,
- Le native advertising,

¹Walmart est une entreprise américaine multinationale spécialisée dans la grande distribution, fondée par Sam Walton et aujourd'hui présidée par son fils aîné, S. Robson Walton.

- Le retargeting publicitaire,
- Personnalisation des résultats de recherche, par exemple sur Google.

I.4. Les caractéristiques des BigData par la problématique des quatre V :

I.4.1. "V" pour Volume :

La quantité de données générée est en pleine expansion et suit une loi quasi exponentielle. Le commerce électronique et les réseaux sociaux sont les grands contributeurs de cette profusion de données.

I.4.2. "V" pour Velocity (rapidité) :

La rapidité de renouvellement des données dans un monde connecté n'est plus à démontrer. Toutes les nouveautés et mises à jour sont stockées en respect de la devise du big data : "On garde tout!".

I.4.3. "V" pour Variety (variété) :

Les bases de données spécialisées sont en mesure de gérer la multiplicité des formats des données : numérique, texte, image...

I.4.4. "V" pour Véracité :

Un décideur sur trois ne fait pas confiance aux données sur lesquelles il se base pour prendre ses décisions. Comment pouvez-vous vous appuyer sur l'information si vous n'avez pas confiance en elle? Etablir la confiance dans les Big Data représente un défi d'autant plus important que la variété et le nombre de sources qui augmentent.

Ces 4V soulèvent la question de la gestion et de l'exploitation des données. Comment gérer des données de plus en plus nombreuses, de plus en plus diverses, et le faire de plus en plus rapidement ? Il faut en effet penser à la façon de les collecter, de les stocker puis de les analyser pour les exploiter à bon escient.

I.5. Les principales technologies de BigData :

Afin de saisir l'impact de ce nouveau paradigme, il est important d'avoir des connaissances de base sur les nouveaux concepts, ainsi que sur les termes et les technologies clé, qui définissent les Big Data. Au cœur de cette révolution, une architecture connue sous le nom de MapReduce: elle consiste en un puissant environnement massivement parallèle qui exécute des fonctions avancées en très peu de temps.

Introduit par Google en 2004, MapReduce permet à un programmeur d'exécuter une transformation de données, qui sont ensuite déléguées et traitées par une architecture de «cluster» qui peut inclure des milliers d'ordinateurs qui opèrent simultanément. À la base, il s'articule en deux étapes : 'map', où les problèmes sont découpés et distribués à plusieurs serveurs parallèles, et 'reduce', où les réponses sont consolidées à partir de chaque map et résolvent le problème initial.

De nombreuses technologies Big Data, telles que Hadoop, Pig et Hive, sont disponibles dans l'offre open source. Celle-ci propose, contrairement aux logiciels propriétaires, de nombreux avantages : innovation continue, réduction des coûts, interopérabilité et développement basé sur des standards.

I.5.1. Cloud Computing :

I.5.1.a. Définition :

Le cloudcomputing ou le cloud informatique en nuage est une infrastructure dans laquelle la puissance de calcul et le stockage sont gérés par des serveurs distants auxquels les utilisateurs se connectent via une liaison Internet sécurisée. Les objets connectés tels que le téléphone mobile, un ordinateur de bureau ou portable, les tablettes tactiles et autres... deviennent des points d'accès pour exécuter des applications ou consulter des données qui sont hébergées sur les serveurs. Le cloud se caractérise également par sa souplesse qui permet aux fournisseurs d'adapter automatiquement la capacité de stockage et la puissance de calcul aux besoins des utilisateurs.

Pour le grand public, le cloudcomputing se matérialise notamment par les services de stockage et de partage de données numériques de type Box, Dropbox, Microsoft OneDrive ou Apple iCloud sur lesquels les utilisateurs peuvent stocker des contenus personnels (photos, vidéos, musique, documents...) et y accéder n'importe où dans le monde depuis n'importe quel terminal connecté.

Le Cloud Computing permet d'accéder à des logiciels en ligne, sous forme d'abonnement, dans de nombreux domaines : ERP « *Enterprise Resource Planning* » est une catégorie de business gestion logicielle qui est généralement une suite d'intégration des applications qu'une organisation peut utiliser pour collecter, stocker, gérer et interpréter les données de nombreuses affaires, CRM « *Customer RelationShip Management* » est une approche de la gestion des interactions de l'entreprise avec actuels et futurs clients, et autres applications métiers... Mais aussi des services de stockage et de calcul accessibles par Internet. [7]

I.5.1.b. Les services de cloudcomputing :

On distingue plusieurs types de services cloud :

➤ **IaaS (Puissance de calcul et de stockage à la demande) :**

L'infrastructure as a Service (IaaS) c'est la mise à disposition par Internet de machines virtuelles aux ressources « facilement » modifiables (les ressources facilement modifiables sont les ressources lexicales tel qu'une chaîne de caractère que l'on veut facilement modifier) et hautement disponible

Pour des calculs complexes ou ponctuels comme une simulations budgétaires au titre d'exemple, calculs prédictifs... ou encore du stockage (notamment dans le cadre de sauvegardes), les administrateurs qui sont en charge d'effectuer les opérations d'installation, de maintenance et de contrôle des ressources informatiques (une ressource informatique peut être un ordinateur, un réseau d'ordinateurs, un équipement de télécommunication, périphérique, média numérique, système d'exploitation, donnée, application ou logiciel et toute combinaison de ceux-ci) trouvent dans le Cloud des environnements aux capacités quasi illimitées et dont la mise en œuvre est quasi instantanée.

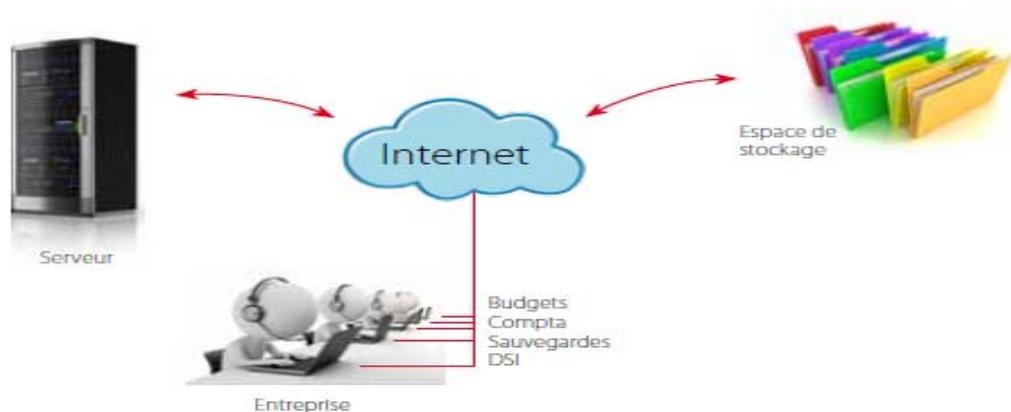


Figure I.1 : L'infrastructure as a Service.

➤ **PaaS (Platform as a Service) :**

Dans ce mode, c'est le fournisseur du service cloud qui administre le système d'exploitation et ses outils. Le client peut installer ses propres applications s'il en a besoin.



Figure I.2 : La Platform as a Service.

➤ **SaaS :**

Software as a Service (le Logiciel à la demande) est accessible à toutes les entreprises et, il est facturé au nombre d'utilisateurs. L'entreprise loue les applications du fournisseur de services. Plus besoin d'acheter un logiciel. Ces applications sont accessibles via différentes interfaces, navigateurs Web, clients légers (désigne parfois des éléments matériels : ordinateurs, et parfois des éléments logiciels comme le navigateur web)...

Les applications sont fournies sous forme de services clés en mains auxquels les utilisateurs se connectent via des logiciels dédiés ou un navigateur Internet. Pour le grand public, il s'agit par exemple de messageries électroniques type Gmail, Yahoo, Outlook.com ou de suites bureautiques type Office 365 ou Google Apps.

Chapitre I : Hadoop

- ❖ De nombreux logiciels sont disponibles dans le Cloud, en mode SaaS :
 - a) **Messagerie** : L'e-mail est certainement l'application la plus utilisée en mode Cloud.
 - b) **CRM** : La gestion de la relation client est l'autre application phare du Cloud.
 - c) **ERP** : Certains logiciels de gestion d'intégrée sont proposés en mode SaaS.
 - d) **Collaboratif** : Les outils de collaboration (partage de documents, réseaux sociaux...se prêtent bien au mode SaaS.
 - e)

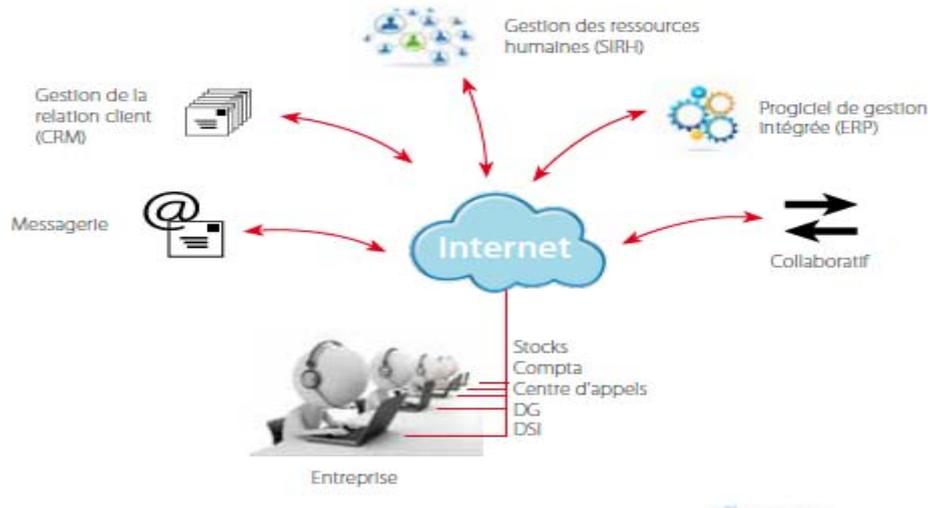


Figure I.3 : Software as a Service.

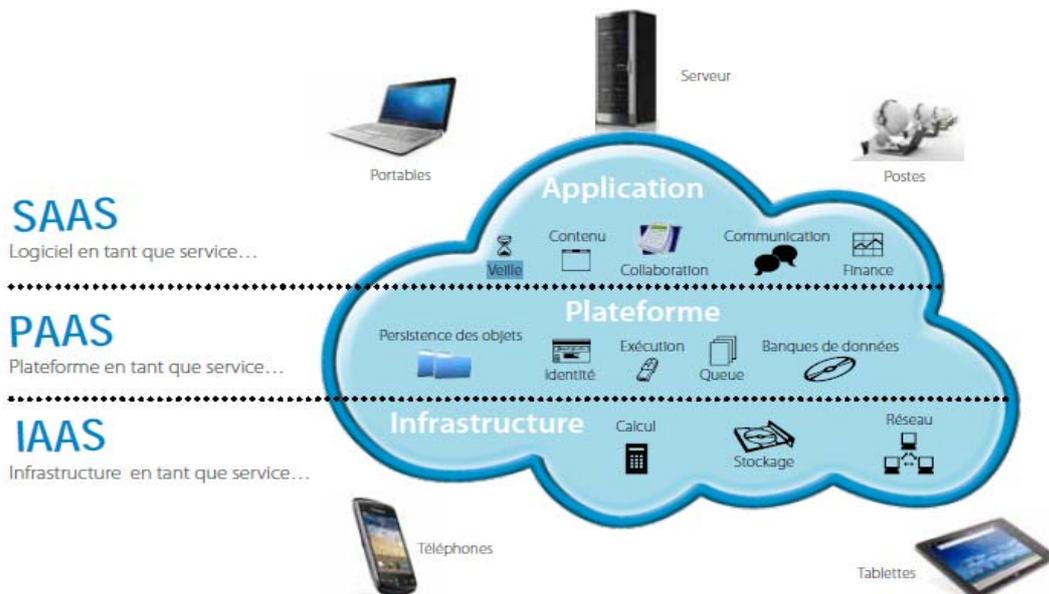


Figure I.4 : Les services de cloudcomputing.

I.5.1.c. Les déploiements :

➤ **Le Cloud Public :**

Cloud « mutualisé » ouvert à tous, type ceux d'Amazon, Microsoft, etc.

➤ **Le Cloud Privé :**

Cloud monté pour une utilisation dédiée à un unique client. Les clouds privés internes, gérés en interne par une entreprise pour ses besoins, les clouds privés externes, dédiés aux besoins propres d'une seule entreprise, mais dont la gestion est externalisée chez un prestataire.

➤ **Le Cloud Hybride :**

Infrastructure faisant appel à la fois à des serveurs sur un Cloud public et sur un Cloud privé.

➤ **Le Cloud Communautaire :**

Un Cloud communautaire est utilisé par plusieurs organisations qui ont des besoins communs.

I.5.1.d. Avantages de cloudcomputing :

➤ **Pas d'investissement initial et Souplesse :**

Plus grande flexibilité des outils informatiques (pas d'installation ni de mises à jour, pas de maintenance matérielle, montée en charge automatisée, nombreux connecteurs disponibles...).

➤ **Réduction des coûts :**

Les utilisateurs ne payent que ce qu'ils consomment.

Forte économie en coût et énergie notamment dans les cas de besoins non constants ou linéaires.

➤ **Sécurité :**

Diminution du risque de panne matérielle.

Les données sont sécurisées et l'informatique est réellement nomade.

➤ **Mobilité :**

L'utilisateur peut à tout moment et à partir de n'importe quel appareil se connecter à ses applications et son workflow.

➤ **Gain de productivité et de temps.**

➤ **Il n'a pas de limites de taille, donc vous n'avez aucun souci de capacité.**

➤ **Vous pouvez accéder à vos applications et services cloud où que vous soyez : tout ce dont vous avez besoin est d'un terminal avec une connexion Internet.**

I.5.2. Des bases de données NoSQL

I.5.2.a Définition :

NoSQL désigne une vaste catégorie de systèmes de stockage de bases de données très différents de l'architecture classique des bases relationnelles.

NoSQL englobe une gamme étendue de technologies et d'architectures, afin de résoudre les problèmes de performances en matière d'évolutivité et de Big Data que les bases de données relationnelles ne sont pas conçues pour affronter. Il est particulièrement utile lorsqu'une entreprise doit accéder, à des fins d'analyse, à de grandes quantités de données non structurées ou de données stockées à distance sur plusieurs serveurs virtuels du Cloud.

Les bases de données relationnelles se prêtent mal aux exigences des applications massivement parallèles exploitant de grandes quantités de données. Les bases de données NoSQL (Not Only SQL) marquent une rupture assez brutale avec la manière de concevoir les schémas de données que l'on pratique depuis quelques décennies. Spécifiquement dédiées aux applications orientées Internet, les bases de données NoSQL pallient ainsi aux difficultés de la gestion de bases de données relationnelles un peu trop conséquentes et réparties sur plusieurs machines. [11] Ils implémentent des systèmes de stockage considérés comme plus performants que le traditionnel SQL pour l'analyse de données en masse (orienté clé/valeur, document, colonne ou graphe).

I.5.2.b Les principaux Bases NoSQL :

- CouchDB, un produit du projet Apache. C'est une base de type "documentaire".
- Apache Cassandra. Anciennement base de données propriétaire de Facebook, Cassandra a été lancée en 2008 sous forme open source, dans la couveuse de projets de la fondation. C'est une base de type "colonnes".
- MongoDB, une base NoSQL orientée document, particulièrement bien diffusée et professionnalisée.
- Hbase (Hbase est un système de bases de données distribuées non relationnelles), est le cœur de Hadoop dans la couveuse de projets de la fondation.
- Hypertable, construite sur le modèle "bigtable" de Google Corp.
- Orientdb est une base de type "graphe"

I.5.3.Des infrastructures de serveurs pour distribuer les traitements :

Des infrastructures de serveurs sont utilisées pour distribuer les traitements sur des dizaines, centaines, voire milliers de nœuds ;c'est ce qu'on appelle le traitement massivement parallèle. Le frameworkHadoop est sans doute le plus connu d'entre eux. Il combine le système de fichiers distribué HDFS, la base NoSQLHBase et l'algorithme MapReduce.

I.5.3.a. Hadoop :

I.5.3.a.1. Définition :

Hadoop est un framework pour l'exécution d'applications sur grand groupe intégré de matériel de base. Le frameworkHadoop fournit de façon transparente des applications à la fois la fiabilité et le mouvement de données. Il met en œuvre un paradigme de calcul nommée Map / Reduce, lorsque la demande est divisée en de nombreux petits fragments de travail, chacun d'eux pouvant être exécutées ou ré-exécutées sur un nœud du cluster. En outre, il fournit un système de fichiers distribué (HDFS) qui stocke les données sur les nœuds de calcul, fournissant très haute bande passante agrégée au sein du cluster. Les deux MapReduce et le système de fichiers distribués Hadoop sont conçus de sorte que les défaillances de nœuds sont automatiquement traitées par le cadre.

Hadoop se présente comme un framework mettant à la disposition des développeurs et des administrateurs un certain nombre de briques essentielles :

- Les éléments d'infrastructure permettant la mise en place de clusters de stockage et de traitement des données. Le stockage des données s'appuie sur HDFS, un système de fichier permettant d'héberger les données à traiter sous forme de fichiers distribués.
- MapReduce : Un framework Java de traitement distribué qui permet de développer des tâches de traitement sur les données du cluster.
- Hadoop fournit par ailleurs un ensemble d'outils qui permet le lancement de tâches ainsi que leur suivi à l'échelle du cluster.

L'idée principale derrière Hadoop : les données sont automatiquement distribuées dans le cluster par HDFS, les traitements doivent s'effectuer au plus près de la donnée (ce que permet MapReduce). Les transferts de données sont ainsi réduits au minimum.

Hadoop est écrit en Java et soutenu par plusieurs startups américaines. Il est en outre devenu une sorte de standard de fait pour l'écriture d'application de traitement de données ralliant l'ensemble des acteurs majeurs du secteur. Il est utilisé par des entreprises ayant de très fortes volumétries de données à traiter. Parmi elles, on trouve notamment des géants du web comme Twitter, LinkedIn, ou encore eBay et Amazon.

I.5.3.a.2. Le principe de fonctionnement de Hadoop :

Hadoop a vu le jour parce que les approches existantes n'étaient pas adaptées au traitement de volumes massifs de données, et pour relever le défi de l'indexation quotidienne du web. Google a développé un paradigme appelé MapReduce en 2004, Yahoo! a mis en place Hadoop comme implémentation de MapReduce en 2005 et a fini par le lancer en tant que projet open source en 2007. À l'instar des autres systèmes opérationnels, Hadoop possède les structures de base nécessaires à effectuer des calculs : un système de fichiers, un langage de programmation, une méthode pour distribuer les programmes ainsi générés à un cluster, un mode pour obtenir les résultats et arriver à une consolidation unique de ces derniers, ce qui est le but.

Avec Hadoop, le Big Data est distribué en segments étalés sur une série de nœuds s'exécutant sur des périphériques de base. Au sein de cette structure, les données sont dupliquées à différents endroits afin de récupérer l'intégralité des informations en cas de panne. Les données ne sont pas organisées par rangs et par colonnes relationnelles, comme dans le cas de la gestion classique de la persistance, ce qui comporte une capacité à stocker du contenu structuré, semi-structuré et non structuré.

I.5.3.a.3. Composent de Hadoop :

❖ MapReduce:

Est une plate-forme de programmation conçue pour écrire des applications permettant le traitement rapide et parallélisé de vastes quantités de données réparties sur plusieurs clusters de nœuds de calcul (se composent sera détail plus tard).

❖ Hive:

Hive est un système d'entrepôt de données facilitant l'agrégation des données, les requêtes ad hoc, et l'analyse de grands ensembles de données stockées dans les systèmes de fichiers compatibles Hadoop. Il permet de fournir une infrastructure de datawarehouse, constituée sur la base du framework Hadoop. Cette implémentation dispose d'un langage de requête particulier de

type SQL appelé HiveQL utilisant des tâches MapReduce pour accéder à des données de manière distribuée.

❖ Pig:

Pig est une plate-forme d'analyse de vastes ensembles de données. Pig comprend un langage de haut niveau gérant la parallélisation des traitements d'analyse.

❖ HCatalog:

HCatalog est une couche de métalangage permettant d'attaquer les données HDFS via des schémas de type tables de données en lecture/écriture.

❖ Système de données distribué de Hadoop :

Hadoop utilise un système de fichiers virtuel qui lui est propre : le *HDFS (Hadoop Distributed File System)*, et qui se décompose en un *namenode* (le maître) et plusieurs *datanodes* (les nœuds de données).

Les *datanodes* regroupent les blocs de données en les répliquant (ci-dessous, les blocs sont tous répliqués). Le *namenode*, quant à lui, va orchestrer les données, et contient les informations concernant l'emplacement des différentes répliques. Le *secondarynamenode* sert à effectuer des checkpoints réguliers du *namenode*, afin de les réutiliser en cas de problème. Ci-dessous un schéma expliquant les différents nœuds du *HDFS* :

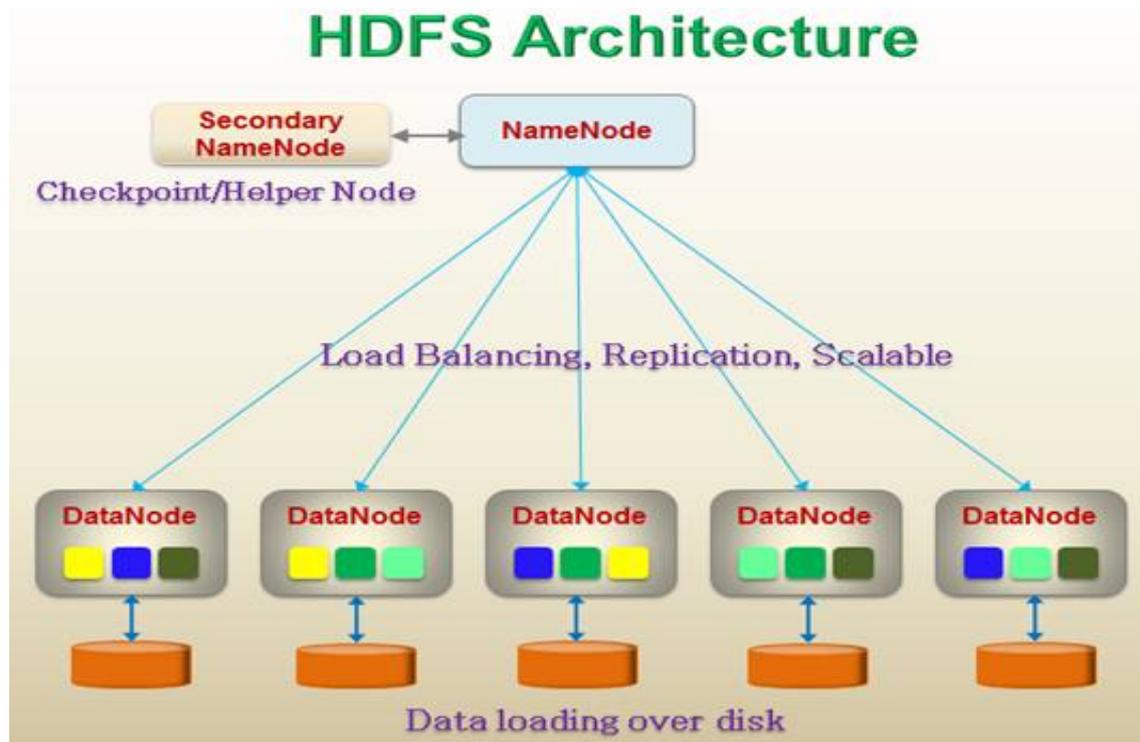


Figure I.5 : Schéma expliquant l'architecture des différents nœuds du HDFS.

❖ *HBase:*

HBase est un système de gestion de base de données non relationnelle, distribuée et orientée colonnes. Il utilise le système de fichiers HDFS pour le stockage de données prenant pour modèle Big Table de Google.

❖ *Oozie:*

Oozie est un outil de workflow dont l'objectif est de simplifier la coordination et la séquence de différents traitements. Le système permet aux utilisateurs de définir des actions et les dépendances entre ces actions.

❖ *ZooKeeper:*

ZooKeeper est un service centralisé pour gérer les informations de configuration, de nommage, et assurer la synchronisation des différents serveurs via un cluster. Tous les services pris en charge par ZooKeeper peuvent être utilisés sous une forme ou une autre par les applications distribuées. Il permet de configurer les machines physiques et services entrant dans le contexte d'applications construites sur Hadoop.

ZooKeeper s'agit également d'un projet de fondation Apache, largement utilisé par les entreprises gérant de gros volumes de données.

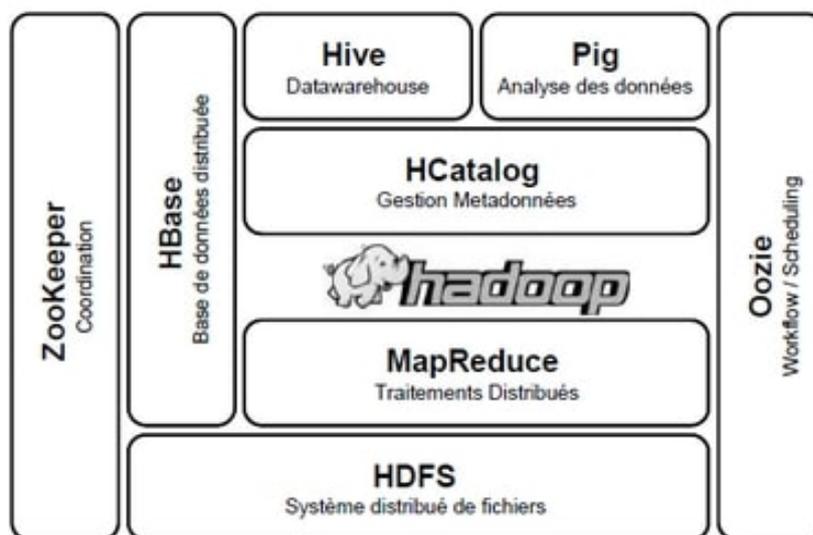


Figure I.6 : Les Principaux composants d'Apache Hadoop.

I.5.3.b. MapReduce :

I.5.3.b.1. Définition :

Map-Reduce est un modèle de programmation massivement parallèle adapté au traitement de très grandes quantités de données et également le nom d'une implémentation du modèle par Google. Il s'agit d'un cadre pour le traitement des problèmes massivement parallèles à travers les très grands ensembles de données utilisées par un grand nombre d'ordinateurs. MapReduce est un framework de traitement distribué, il est une composante essentielle de l'Apache Hadoop comme cadre logiciel et il est un produit de Google Corp.

Les programmes ou les développeurs débutants adoptant ce modèle sont automatiquement parallélisés et exécutés sur des clusters (grappes) d'ordinateurs. Les développeurs débutants trouvent le cadre MapReduce bénéfique car routines de bibliothèques peuvent être utilisés pour créer des programmes parallèles sans aucuns soucis sur la communication infra-cluster, suivi des tâches ou des processus de traitement de l'échec.

MapReduce fonctionne sur un grand cluster de machines et de produits de base est hautement évolutive. [18] Il a plusieurs formes de mise en œuvre fournis par plusieurs langages de programmation, comme Java, C # et C ++ et de nombreux organismes tel que Google, Yahoo, etc...

I.5.3.b.2. Techopedia de MapReduce :

Le cadre de MapReduce comporte deux parties:

1. **Une fonction "carte"**, qui permet aux différents points de la grappe distribuée afin de répartir leur travail
2. **Une fonction "Réduire"**, qui vise à réduire la forme finale des résultats des pôles en une sortie

Le principal avantage du cadre de MapReduce est sa tolérance aux pannes, où sont attendus les rapports périodiques de chaque nœud du cluster lorsque le travail est terminé.

Une tâche est transférée d'un nœud à l'autre. Si les avis de nœud maître qu'un nœud a été silencieux pendant un intervalle plus long que prévu, le nœud principal exécute le processus de réaffectation à la tâche congelés / retardé.

Le cadre de MapReduce est inspiré par les fonctions "carte" et "Réduire" utilisées dans la programmation fonctionnelle. Le traitement informatique se produit sur les données stockées dans un système de fichiers ou dans une base de données, qui prend un ensemble de valeurs clés d'entrée et produit un ensemble de valeurs de clés de sortie.

I.5.3.b.3. Les composants de MapReduce :

MapReduce est composé de plusieurs éléments, y compris:

- **JobTracker** c'est le nœud maître qui gère tous les emplois et les ressources dans un cluster.
- **TaskTrackers** sont des agents déployés à chaque machine du cluster pour exécuter le plan et de limiter les tâches.
- **JobHistoryServer** c'est un composant qui permet de suivre les travaux terminés, et est généralement déployé comme une fonction distincte ou avec **JobTracker**.

Pour distribuer des données d'entrée et de rassembler les résultats, MapReduce fonctionne en parallèle selon la taille des amas (accumulation) massif. Parce que la taille de cluster ne peut pas affecter les résultats finaux d'une tâche de traitement, les emplois peuvent être répartis sur presque tous les serveurs. Par conséquent, MapReduce et le cadre général de Hadoop simplifient le développement de logiciels. Les programmeurs peuvent utiliser des bibliothèques de MapReduce pour créer des tâches sans traiter ni communication ou de coordination entre les nœuds.

MapReduce est également tolérant aux pannes, chaque nœud rend compte périodiquement son état à un nœud maître. Si un nœud ne répond pas comme prévu, le nœud maître réattribue ce morceau de la tâche à d'autres nœuds disponibles dans le cluster. Cela crée la résilience et rend MapReduce pratique pour fonctionner sur les serveurs.

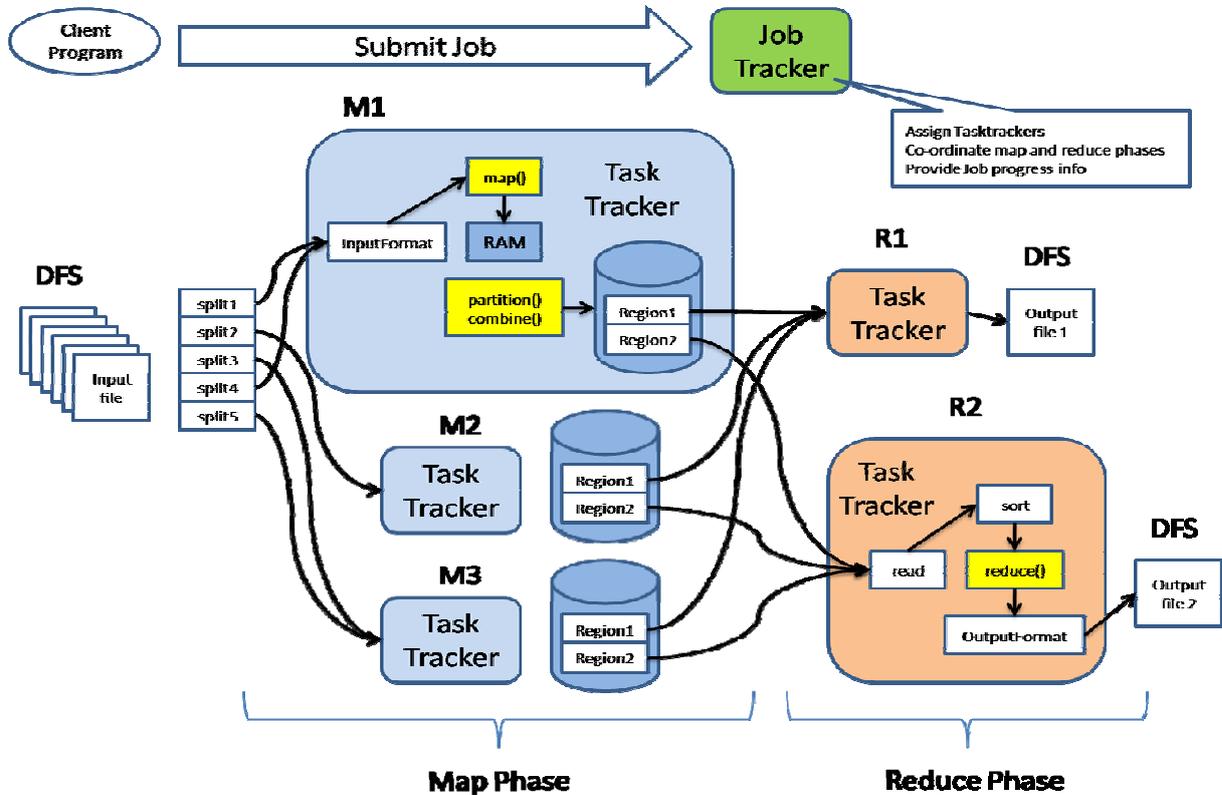


Figure I.7 : Les composants de MapReduce

I.5.3.b.4. Modèle de programmation : [22]

L'utilisateur fournit deux fonctions :

1. Map :

✓ La procédure :

- Prend en entrée un ensemble de « Clé, Valeurs ».
- Retourne une liste intermédiaire de « Clé1, Valeur1 ».
- **Map (key, value) ->list (key1, value1).**

✓ Le résultat :

- Itérer sur un grand nombre d'enregistrements.
- Extraire quelque chose ayant un intérêt de chacun d'eux.

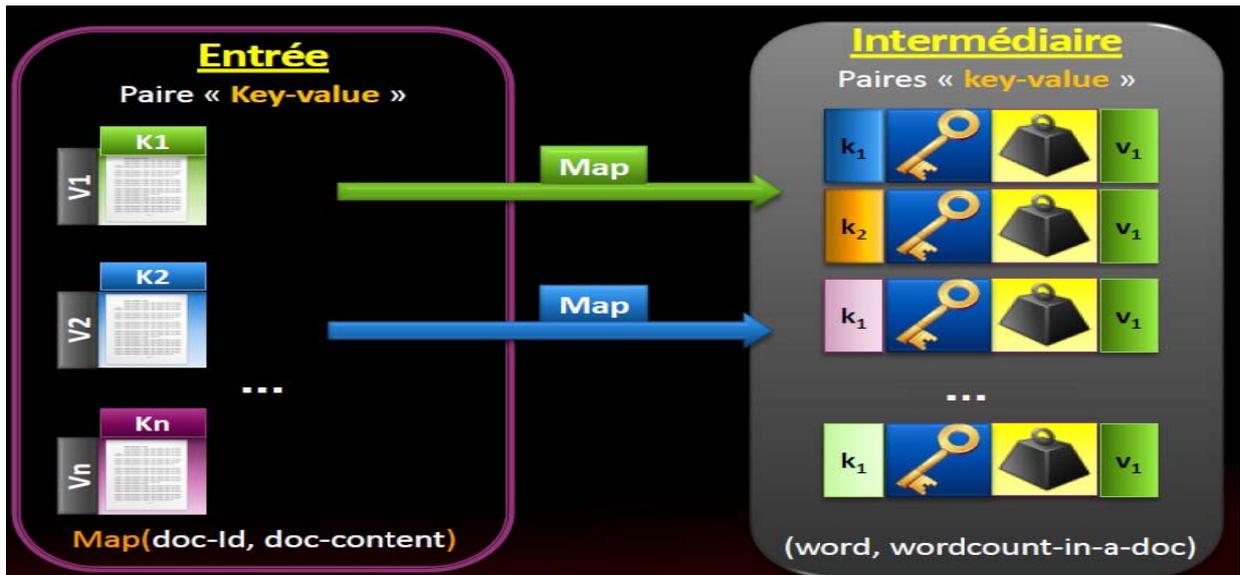


Figure I.8 : La fonction Map.

2. Reduce :

✓ La procédure :

- Prend en entrée une liste intermédiaire de « Clé1, Valeur1 ».
- Fournit en sortie un ensemble de « Clé1, Valeur2 ».
- **Reduce (key1, list (value1)) ->value2.**

✓ Le résultat :

- Regrouper et trier les résultats (intermédiaires).
- Agréger ces résultats.
- Générer le résultat final.

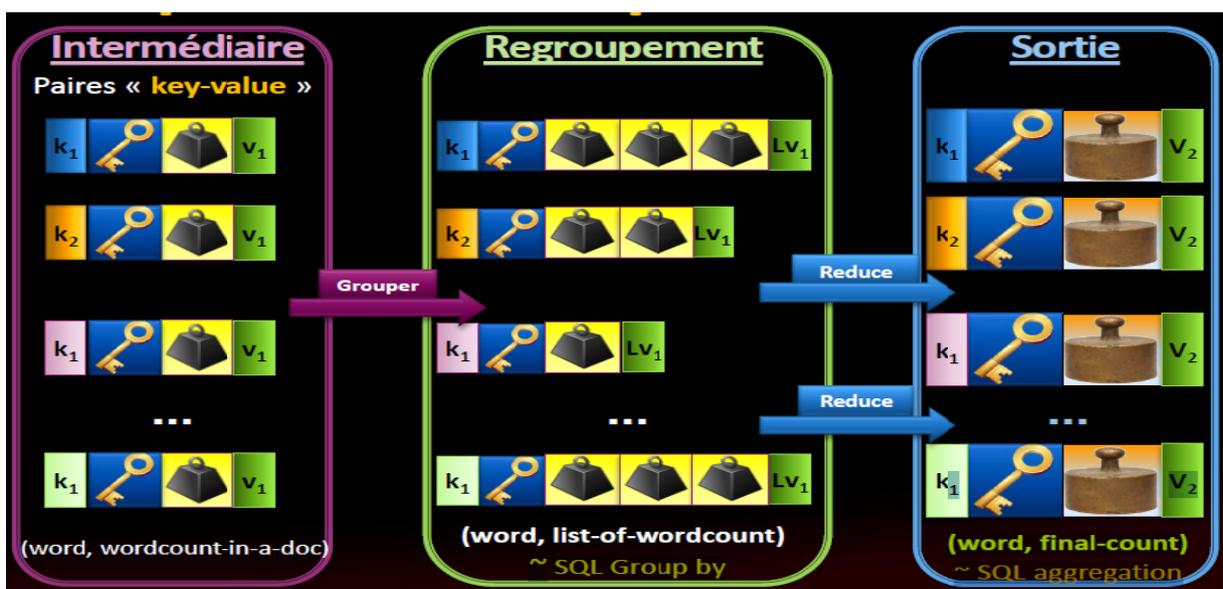


Figure I.9 : La fonction Reduce

I.5.3.b.5. Fonction globale de MapReduce :

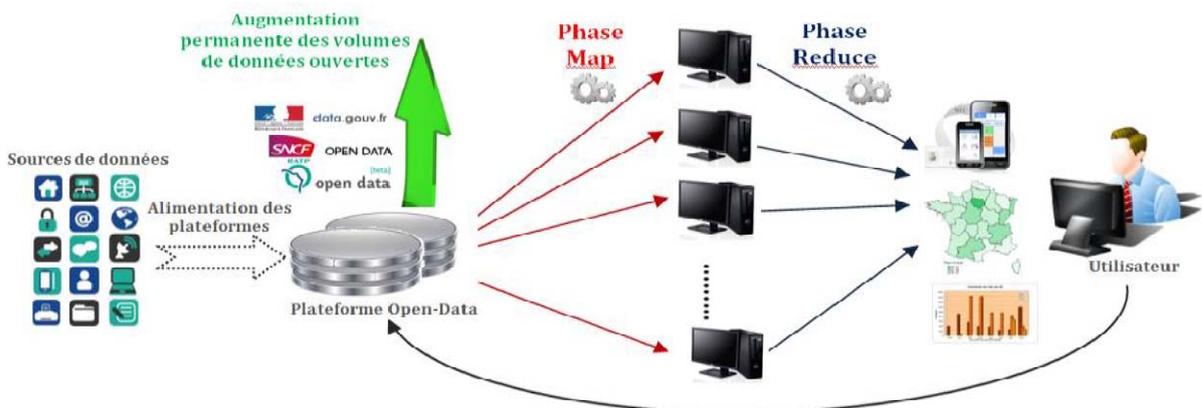
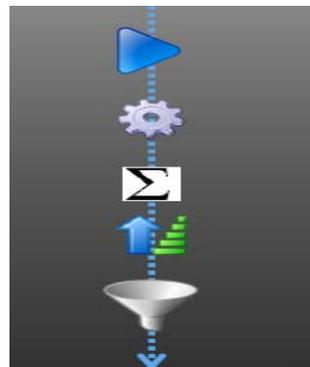


Figure I.10 : Fonction globale de MapReduce.

I.5.3.b.6. Les phases de MapReduce : [23]

1. Initialisation ;
2. Map ;
3. Shuffle (regroupement) ;
4. Sort (tri) ;
5. Reduce.



Figure

I.11 : Les phases de MapReduce.

Exemple : [22]

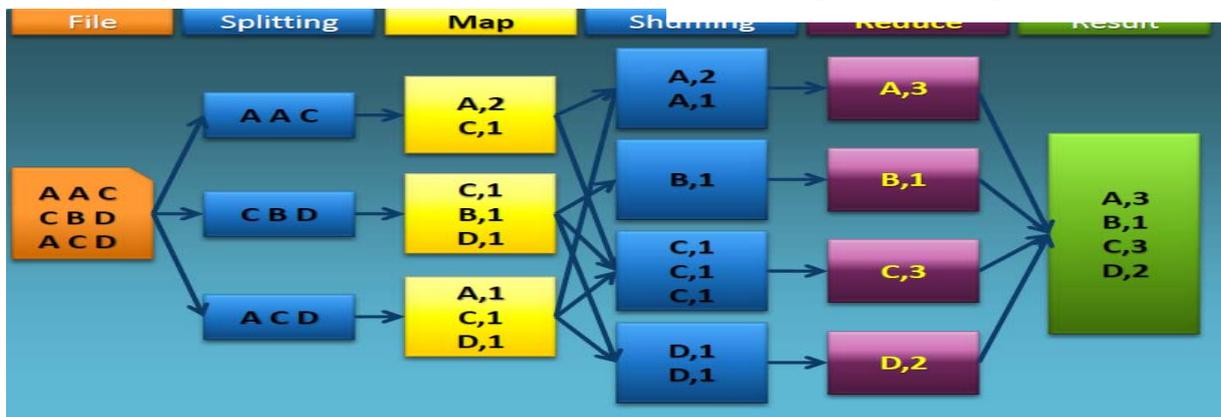


Figure I.12 : Un exemple applicatif des phases de MapReduce.

I.5.3.b.7. Principes de base de l'algorithme : [22]

- ✓ Une tâche est divisée en deux ou plusieurs sous-tâches, chaque sous-tâche est traitée indépendamment, puis leurs résultats sont combinés.
- ✓ Application de trois opérations majeures : Slip, Compute et Join.
- ✓ C'est le principe de « Diviser pour Reigner » avec une structure pseudo-hiérarchique.

I.5.3.b.8. Gestion des données : [22]

- ✓ Les données en Entrée et en Sortie sont stockées sur un system de fichiers distribués ;
- ✓ Les données sont stockées au plus proche de leur source ;
- ✓ Les données intermédiaires sont stockées sur le système de fichier local des unités «Map » et « Reduce » ;
- ✓ Les données en sortie représentent souvent une entrée pour une autre unité MapReduce.

I.5.3.b.9. Mécanisme d'optimisation : [22]

MapReduce implémente plusieurs mécanismes d'optimisations :

- ✓ Optimisation de la bande passante réseau.
- ✓ Utilise un mécanisme de spéculation pour placer des tâches sur des noeuds différents du File System.

Les '*Combiner*' sont utilisés pour réduire le volume de données transmis entre Map et Reduce.

- ✓ Utilise un mécanisme similaire à la fonction *Reduce*.

I.5.3.b.10. Implémentation MapReduce d'hadoop :

➤ L'algorithme de MapReduce dans Hadoop :

Le *MapReduce* est une technique qui décompose le traitement d'une opération (appelée "job" chez Hadoop) en plusieurs étapes, dont il ya deux étapes sont élémentaires, afin d'optimiser un traitement parallèle des données.

1. Le Mapping :

Cette étape accomplit une opération spécifique sur chaque élément de la liste en *input* : à partir d'une liste sous la forme <clé, valeur>, il génère alors une liste en output sous la même forme :

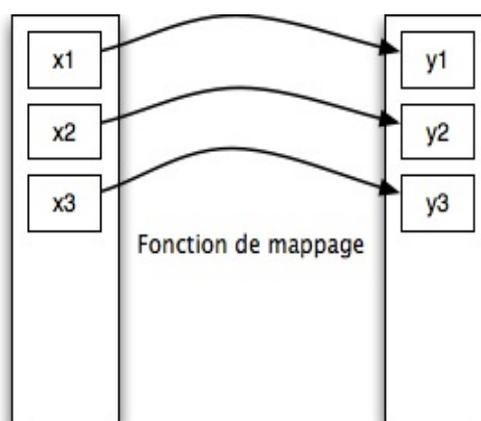


Figure I.13 : La fonction de Mappage

2. Le Reducing :

L'opération qui se situe après le *Mapping* et le *Reducing* est appelée le *Shuffling*, et réarrange les éléments de la liste afin de préparer le *Reducing*. Le traitement voulu est alors effectué, donnant l'*output* final :

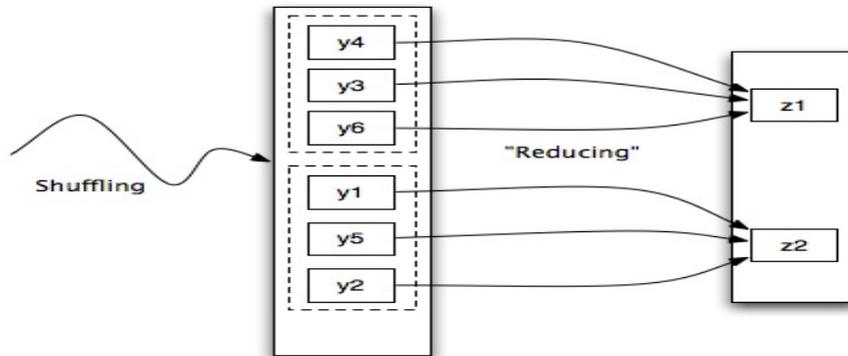


Figure I.14 : La fonction de Reducing

➤ Un exemple de traitement de MapReduce avec Hadoop :

Nous avons donc pu nous familiariser avec les concepts clés de Hadoop, il est temps d'illustrer cela par un exemple. Nous n'avons pas un pétaoctet de données sous la main, mais nous allons profiter de la prolifération de l'open data pour effectuer un petit traitement sur des données simples. Le site open data de nos amis belges donne librement accès à de nombreuses données, comme par exemple la liste des différents prénoms recensés en Belgique en 2007, qui se présente sous la forme d'un fichier Excel contenant plus de 38 000 entrées.

Ces données existent aussi pour les femmes, nous avons tout concaténé afin d'avoir un fichier Les CVs regroupant tous les prénoms belges en 2007, hommes et femmes, sous la forme (Nom, Nombre d'apparitions) :

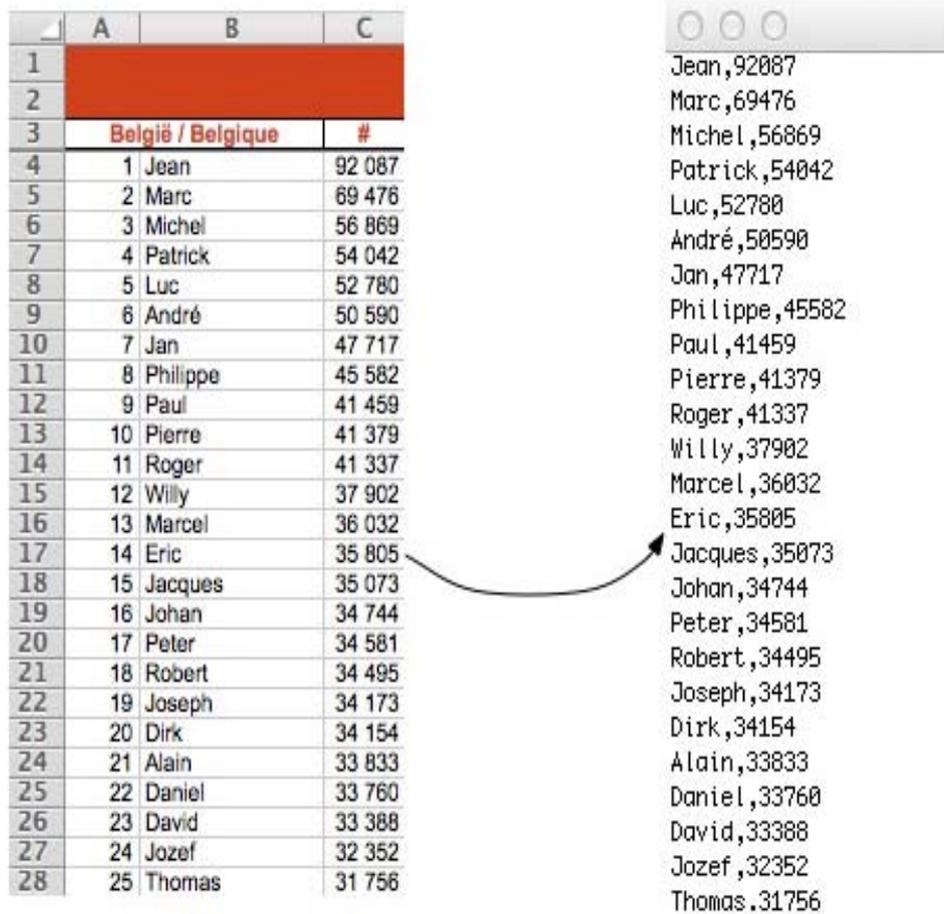


Figure I.15 : Exemple de traitement de MapReduce avec Hadoop.

I.5.3.b.11. Mise en œuvre de MapReduce : [22]

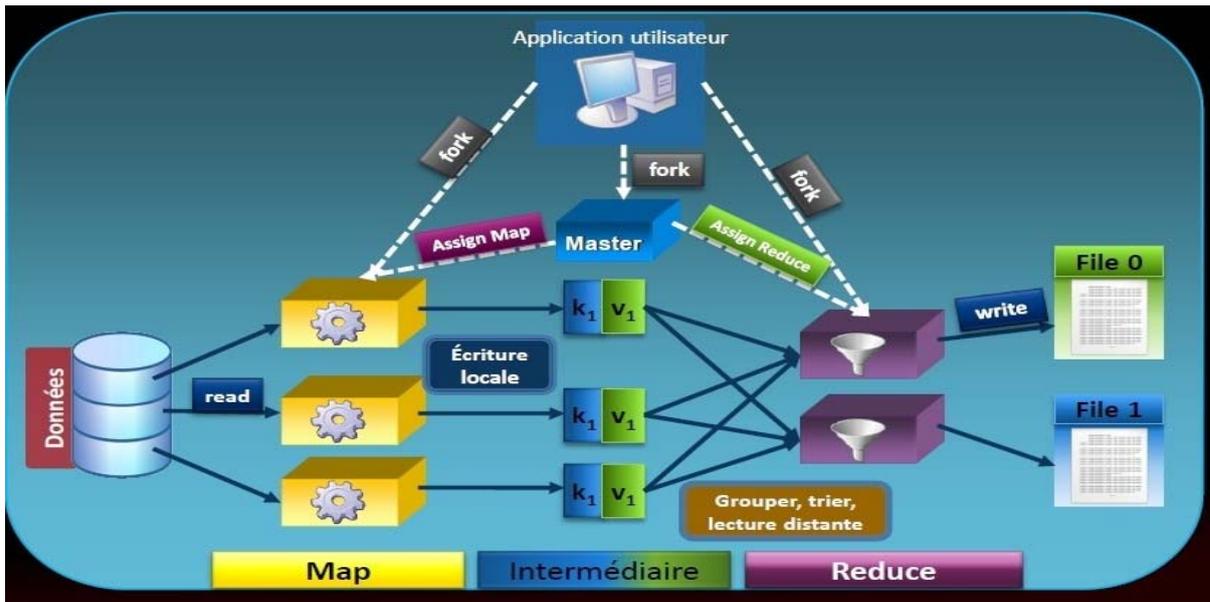


Figure I.16 : La Mise en œuvre de MapReduce.

I.5.3.b.12. Exemples des domaines qui utilisent MapReduce : [22]

✓ Google :

- _ Construction des Index pour Google Search.
- _ Regroupement des articles pour Google News.

✓ Yahoo :

- _ Alimenter *Yahoo! Search* avec “*Web map*”.
- _ Détection de Spam pour *Yahoo! Mail*.

✓ Facebook :

- _ Data mining (fouille de données).
- _ Détection de Spam.

I.5.3.b.13. Exemples d’applications de MapReduce :

Pour des données web comme :

- _ Compter le nombre de certains mots dans un ensemble de documents.
- _ Calculer un fichier inverse pour un ensemble de documents.
- _ Compter les fréquences d’accès URL dans un web log.
- _ Calculer un graphe inverse de liens web.
- _ Compter les vecteurs de termes (Résumant les mots les plus importants) dans un ensemble de documents.
- _ Tri distribué.

I.5.3.b.14. Avantage et Inconvénients :

Avantage :

- _ Fourni une abstraction totale des mécanismes de parallélisations sous-jacents.
- _ Peu de tests sont nécessaires. Les bibliothèques MapReduce ont déjà étaient testées et fonctionnent correctement.
- _ L’utilisateur se concentre sur son propre code.
- _ Largement utilisé dans les environnements de *Cloud Computing*.

Inconvénients :

- _ Une seule entrée pour les données.
- _ Deux primitives de haut-niveau seulement.
- _ Le flux de données en deux étapes le rend très rigide.
- _ Le système de fichiers distribués (HDFS) possède une bande passante limitée en entrée / sortie.
- _ Les opérations de tris limitent les performances du Framework (implémentation Hadoop).

I.5.4. Le stockage des données en mémoire (Memtables) :

I.5.4.a. Définition de Stockage In-Memory ou Base De Donnée In-Memory :

Une base de données dite « en mémoire » (in-Memory), ou IMDB (In Memory DataBase), ou encore MMDB (Main Memory DB), désigne une base de données dont les informations sont stockées en mémoire centrale afin d'accélérer les temps de réponse. Cette nouvelle génération profite de la baisse du prix des mémoires vives et de la puissance de calcul offerte par les nouvelles générations de processeurs multi-cœurs. En limitant au maximum les accès disques, les requêtes s'exécutent en un temps record, c'est évidemment une avancée majeure pour le monde du business intelligence en version analytique. L'analyste n'est plus gêné par la lenteur des accès à l'infrastructure et aux disques de stockage pour bâtir de solides modèles d'analyses.

I.5.4.b. La mise en œuvre :

Les bases de données "In Memory" sont généralement construites comme des bases relationnelles. Elles sont conformes aux exigences ACID (Atomicity, Consistency, Isolation, Durability) qui garantissent l'intégrité des transactions. Les données contenues en mémoire sont volatiles par principe. Un système de sauvegarde périodique par image disque, snapshot, permet de sauvegarder la base. Ce système est complété d'une historisation des transactions afin de remettre la base en état en cas de coupure de courant.

Conclusion :

L'étude appliquée aux données ouvertes a été menée dans une optique exploratoire, nous nous sommes en particulier intéressés à l'application du paradigme MapReduce qui est une abstraction utile, il permet de simplifier le traitement de données à grande échelle chez Google. Les paradigmes de programmation fonctionnelle peuvent être appliqués au traitement de données à grande échelle d'utilisation: focaliser sur le problème, laisser la librairie gérer les détails.

Ainsi, les résultats du traitement des données peuvent être valorisés par leur intégration dans des services innovants, tels que, des sites web ou les applications mobiles. De plus, l'intégration des dimensions spatiales permet à l'utilisateur final une représentation des données sous la forme de carte(s). Enfin, en termes de performances, on notera que le traitement des données publiques avec le modèle *MapReduce* donne pas de meilleurs résultats pour les données considérées comme relativement petits, par exemple inférieurs à 1 Go. Par contre, son efficacité est appréciable au fur et à mesure que la taille des données augmente. Et l'ajout de nœuds supplémentaires au cluster permet d'améliorer significativement les temps de traitement.

Dans le chapitre suivant on va expliquer le concept de la vidéo et l'image pour la découverte et la recherche et aussi l'obtention de l'information et les différentes caractéristiques de l'image qui vont nous aider par la suite de comprendre le processus de la détection et ses différentes étapes.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

Introduction :

La recherche d'information est une des problématiques majeures de l'informatique d'aujourd'hui. Elle prend des tendances de plus en plus multimédia, la quantité d'information audiovisuelle disponible est en augmentation constante et le besoin d'accès fréquent à cette information touche un public de plus en plus large.

De nombreux efforts ont été déployés dernièrement dans ce domaine. Cependant, les outils disponibles semblent loin de répondre aux exigences des utilisateurs et le besoin des moteurs de recherche complets, performants et adaptés à des applications réelles telles dans les secteurs de la télévision numérique, secteurs de surveillances ou la lecture des plaques d'immatriculation, se fait fortement ressentir. Ces moteurs doivent intégrer un système d'extraction et d'indexation faisant face à des quantités croissantes de contenu audiovisuel tandis qu'avec cette croissance constante de masse de données vidéo, outre les problèmes de stockage et d'archivage, les problèmes d'utilisation, de recherche, de navigation et d'extraction se posent.

Les flux ou en d'autre terme les documents vidéo ont un caractère multimédia (image, texte, son) qui fait que la recherche par le contenu dans ceux-ci présente un certain nombre de spécificités. Par exemple, un concept donné (personne, objet...) peut être présent dans différentes manières : il peut être visualisé, il peut être entendu, on peut en parler dans le discours, et la combinaison de ces cas peut également se produire. Naturellement, ces distinctions sont importantes pour l'utilisateur, la consultation des documents vidéo doit en effet être facile. Par conséquent un processus d'indexation et de recherche doit être mis en place de manière à accélérer la recherche et la reconnaissance de l'information souhaitée.

Ce chapitre va expliquer le concept de la vidéo et l'image tout en parlant sur les composants, les formes ...etc, puis on va détailler les caractéristiques de l'image et leurs utilisation.

II.1. Le média vidéo :

II.1.1. Définition de la vidéo : [1]

Une vidéo est une succession d'images se déroulant les unes derrière les autres à un rythme d'au moins 20 images par seconde. En effet, la distinction maximale de l'œil humain est d'environ 20 images par seconde. Ainsi, en affichant plus de 20 images par seconde, il est possible de tromper l'œil et de lui faire croire à une image animée. On caractérise la mobilité d'une vidéo par le nombre d'images par secondes. [4]

D'un point de vue physique ou informatique, un document vidéo est une combinaison de sous-médias ou « pistes » organisés suivant un axe temporel. Chaque piste est présentée sous la forme d'un flux d'éléments synchronisés entre eux, peuvent contenir:

- **Image animée** : tous les flux vidéo contiennent une piste appelée « image ». Les éléments de cette piste sont des images émises à une fréquence fixe (typiquement de 24 à 30 par secondes) et certains documents vidéo peuvent contenir plusieurs pistes image en parallèle ;ce cas de figure est assez rare.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

- **Son** : la plupart des documents vidéo contiennent aussi une ou plusieurs pistes appelée «audio». Les éléments de cette piste sont des échantillons émis à une fréquence fixe (typiquement de 16000 à 48000 par secondes). Une piste audio peut encore être composée de plusieurs flux de tels éléments en parallèle (cas de la stéréo, du codage à 6 canaux). Un flux vidéo peut contenir plusieurs pistes audio en parallèle correspondant à plusieurs langues par exemple.
- **Texte** : certains documents vidéo contiennent aussi une ou plusieurs pistes textuelles.

Le contenu des documents vidéos est « **rendu** » à l'utilisateur par le moyen d'un matériel adapté qui convertit les flux d'éléments représentés sous forme numérique vers une modalité physique appropriée (par l'intermédiaire d'écran, de haut-parleurs, ...).

De ce fait, le spectateur « l'individu » qui regarde et écoute le document ou le flux, la vidéo lui apparaît comme une expérience «globale», qui lui fait percevoir des éléments qui ont du sens pour lui (objets, personnes, événements, ...) et des relations entre eux, en ignorant la présence des différentes pistes et des éléments individuels qui la composent.

II.1.2. Le flux vidéo (ou flux audiovisuel) : [1]

Par définition, un flux est un déplacement d'éléments dans le temps et dans l'espace, les éléments d'un document vidéo sont des données audio-visuelles peuvent être aperçues comme une superposition de flux.

Le contenu d'un document audiovisuel est une structure physique exposant l'ensemble des images, textes et la synchronisation du son dans un schéma plus ou moins complexe, ainsi le terme contenu désigne l'ensemble des informations formant une vidéo (image, audio et texte).

II.1.3. Contenu : [1]

Les documents vidéo peuvent avoir des contenus extrêmement variés ayant une ou plusieurs structures internes (journaux télévisés, films, vidéosurveillance, sécurité routière, etc..). La structure relative au contenu est la structure sémantique du document peut parfois apparaître mal définie, ambiguës ou subjectives.

Cette structure se présente souvent de manière hiérarchique : un document vidéo est composé hiérarchiquement et respectivement selon un arbre, de ce qui suit :

Suites de séquences, « séquence » est une suite de scènes, « scène » est une suite de plans, « plans » est une suite d'images, « images » sont des images physiques, éléments de la piste image, « régions » est une régions dans une image. La figure en dessous montre la composition d'un journal télévisé.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

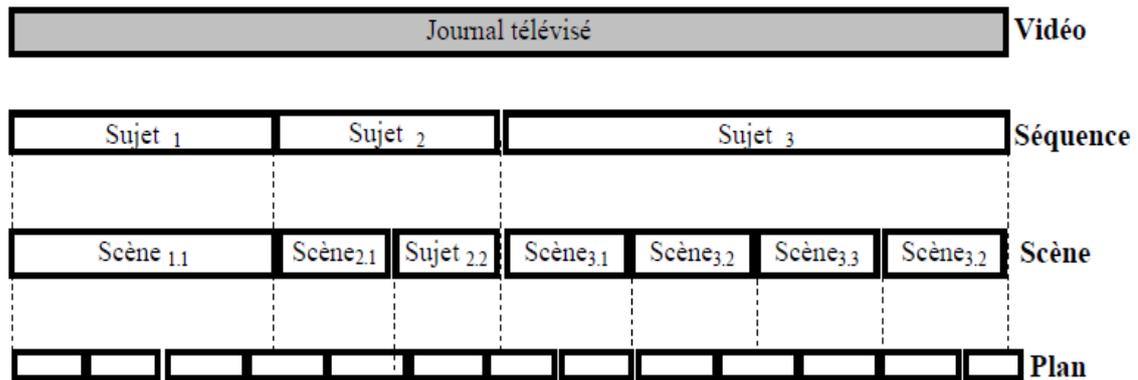


Figure II.1: Structure hiérarchique d'un journal télévisé selon le modèle « vidéo, séquence, scène, plan », les scènes peuvent correspondre à une apparition du présentateur ou à un reportage sur le terrain, les plans correspondent à une prise unique de caméra.

Toujours pour des raisons pratiques, les éléments inférieurs de la hiérarchie ne se recouvrent pas entre eux, ils se suivent temporellement les uns les autres et ensemble ils recouvrent l'élément de niveau juste supérieur dans l'arbre hiérarchique.

NB : Toutes les pistes peuvent être décomposées sauf pour les niveaux les plus bas où elle peut devenir spécifique à un média (le découpage en plans spécifique média image par exemple). Pour le média audio, une ou plusieurs autres décompositions de niveaux les plus bas existent aussi et de même pour le sous média texte.

- **Définition de flux visuel :**

Filmer une scène, c'est enregistrer à l'aide d'une caméra 25 ou 30 images toutes les secondes, et qu'on fait défiler ces images au même rythme devant un téléspectateur, celui-ci ne pourra pas distinguer dans ce qu'il voit une suite d'images. En effet, la persistance des images sur sa rétine fait qu'une image est remplacée par une autre, de telle sorte que les zones qui ne changent pas sont perçues comme un continuum¹ stable, tandis que les mouvements sont uniformisés par la perception.

- **Document numérique :**

Un document numérique vidéo contient souvent du son synchronisé avec les images dans le temps, qui peuvent être enregistrés en même temps, ou bien en différé, ainsi, il est considéré comme une seule ou un ensemble de séquences vidéo, identifiée par son instant de départ et de fin sur l'échelle de temps du document vidéo numérique.

De ce fait, une séquence vidéo est une mise en ordre des éléments d'un document numérique vidéo selon un ensemble de règles. Il s'agit des images numériques disposées chronologiquement avec éventuellement une bande son. En général les documents vidéo sont le résultat d'un montage qui consiste à coller l'un après l'autre des plans.

¹ C'est l'ensemble d'éléments homogènes tels que l'on puisse passer de l'un à l'autre de façon continue.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

II.1.3.1. Plan : [1]

Les plans sont une combinaison d'unités sémantiquement plus cohérentes appelées scènes considérée comme étant l'unité de référence de la vidéo. En d'autres termes, quand on décrit une vidéo à quelqu'un qui ne l'a pas vue, on donne le thème général de chaque scène.

Un plan est défini dans un cadre de montage vidéo à partir d'une série d'images acquises par une seule caméra. La segmentation de la vidéo en plan peut être faite par un processus automatique qui se base sur la détection de transitions entre les plans. Par contre, pour segmenter une vidéo en des parties selon une description sémantique : unité de lieu « scène » ou unité de sujet « séquences », on doit généralement faire appel à un opérateur humain.

II.1.3.2. Scène : [1]

Une scène est constituée d'un ensemble de plan ayant une même unité de lieu. Au niveau visuel, une scène vidéo soulève des problèmes tels que par exemple :

- _ Comment délimiter une scène?
- _ Sur quelle logique doit-on se baser pour déterminer l'enchaînement des scènes?

II.1.3.3. Séquence : [1]

Une séquence regroupe divers plans et scènes. Elle constitue une unité de sujet (par exemple un reportage dans un journal télévisé).

Parmi ces trois unités, le plan représente l'entité de base. Au niveau syntaxique, une telle structuration ressemble à celle d'un document textuel (mot, phrase, paragraphe).

II.1.4. Unité audiovisuelle (UAV) : [1]

Unité audiovisuelle est la définition d'une entité abstraite représentant un segment quelconque de document vidéo elle dépend de type de l'information ainsi que la manière selon laquelle ce type d'information est segmenté ainsi elle sera identifiée par la manière dont le document vidéo est segmenté. Par exemple, si on prend le cas d'une suite d'image on peut considérer qu'un plan représente une unité audiovisuelle. Par contre, pour le contenu audio, la segmentation selon le changement de locuteur peut constituer l'unité de repérage.

UAV est décrite par un élément d'intérêt qui est un concept audio-visuel de façon pertinente.

II.1.5. Caractéristiques d'un document vidéo : [1]

II.1.6.1. Volume :

Les données vidéo sont trop volumineuses et nécessitent plus d'espace de stockage par rapport à l'image et au texte. En effet, une seconde de vidéo MPEG contient 25 ou 30 images.

II.1.6.2. Hétérogénéité du contenu :

Les informations contenues dans un document vidéo proviennent de sources variées (audio, texte, image). Cette hétérogénéité cause souvent des problèmes au niveau segmentation et analyse du contenu. Pour le cas des documents images ou textuels le problème d'hétérogénéité ne se pose pas.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

II.1.6.3. La nature spatio-temporelle :

Les données textuelles sont ni spatiales ni temporelles. Dans les données images, la représentation spatiale est bien définie par contre il n'y a pas la notion du temps dans les images fixes. Dans un document vidéo l'aspect spatiotemporel est une de ses caractéristiques.

II.1.6.4. L'expressivité sémantique :

Il est parfois facile d'interpréter le contenu d'un document en visualisant son contenu mais il est souvent encore mieux de disposer également d'une information sonore. Un document vidéo possède une expressivité sémantique beaucoup plus riche qu'un document textuel ou même qu'un document image.

II.1.6.5. Durée :

Pour représenter le contenu d'une image fixe, la durée de présentation est variable. Dans un document vidéo, une image possède une durée de présentation bien définie qui dépend du type de vidéo (~1/30 seconde pour le cas de vidéo NTSC² et 1/25 seconde pour le cas des vidéos PAL³ / SECAM).

II.1.6.6. Variabilité de la qualité :

La qualité d'une vidéo dépend de la résolution dans laquelle elle est codée ainsi que du taux de compression utilisé pour son codage. Un même contenu peut être codé avec des résolutions très variables en fonction des capacités de stockage et/ou de transmission. Cela a un impact important sur la qualité des traitements qui peuvent être effectués pour une indexation automatique. De ce point de vue, le cas de la vidéo est différent de celui du texte pour lequel le contenu et son interprétation ne sont pas ou sont peu affectés par le format dans lequel il est encodé.

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

Les systèmes d'indexation et recherche d'information basés sur le contenu vidéo, communément appelés CBVIR (acronyme de « Content Based Video Indexing and Retrieval »), font intervenir deux phases qui sont l'indexation et la recherche.

Les techniques pour l'indexation et la recherche sont nombreuses et variées, les techniques multimodales sont encore peu nombreuses. La plupart des techniques de recherche multimédia exploitent l'information temporelle dans le document. Le projet Informedia utilise le flux visuel pour la segmentation et le flux audio pour la classification du contenu. De tels systèmes existent également pour des domaines visuels particuliers comme les journaux télévisés et les émissions sportives. Les structures présentées par Yeung et Liu utilisent des modèles de Markov⁴ pour l'analyse de la parole, la détection des événements, etc.

²NTSC (*National Television System Committee*, c'est-à-dire « Comité du système de télévision nationale ») est un standard de codage analogique de la vidéo en couleurs lancée aux États-Unis en 1953.

³*Phase Alternating Line* (PAL : « alternance de phase suivant les lignes ») est un standard historique vidéo couleurs avec 25 images par seconde et 625 lignes par image.

⁴Les modèles de Markov sont des automates probabilistes à états finis.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

II.2.1. Méthodes d'Indexation et de Recherche : [3]

L'étude des structures d'indexation multidimensionnelle a émergé avec l'apparition des structures utilisant des arbres hiérarchiques comme le k-d tree (1979) et les nombreuses déclinaisons du R-tree (1984). Ces structures et leurs variantes ont montrées leur efficacité mais elles sont dépassées pour de grandes dimensions, ce phénomène est connu sous le nom de la « malédiction de la dimension » ; Deux méthodes apparaissent en 1998 pour faire face à cette malédiction :

- **La première technique est le Va-File (1998)** est une méthode basée sur une approximation de l'espace de données par codage. Inspirées par cette méthode, d'autres méthodes sont apparues comme l'IQ-Tree (2000) et le LPC-File (2002). Ces méthodes ont de bonnes performances pour une répartition de données homogènes mais sont dépassées par des données hétérogènes de grandes dimensions.
- **La deuxième technique est la pyramide (1998)** est la première des méthodes basées sur un découpage de l'espace de données pour représenter les points multidimensionnels par des index ou valeur clé à une dimension. Plusieurs méthodes se sont inspirées de la technique de la pyramide, telles que iMinMax (2001), Idistance (2001) et P+Tree (2004). Cette catégorie de méthodes est devenue moins intéressante par conséquent une nouvelle méthode a été proposée qui est nommée Kpyr (2005).

II.3. Le media Image (définition et principe):

II.3.1. Définitions :

II.3.1.1. Pixels :

Une image est constituée d'un ensemble de points appelés pixels(pixel est une abréviation de Picture Element de l'anglais). Le pixel représente ainsi le plus petit élément constitutif d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image. [7]

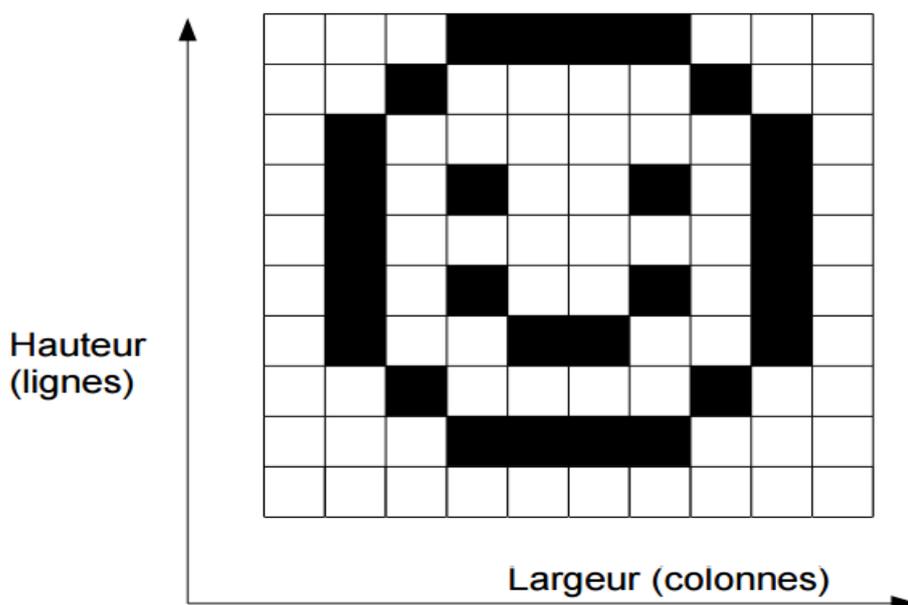


Figure II.2 : Les pixels

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

II.3.1.2. Définition 1 :

Au sens étymologique (son origine), le mot « image », découlant du latin « imago », désigne la représentation visuelle d'un objet par différents moyens ou supports (dessin, photographie, peinture...). L'image d'un objet peut alors être interprétée comme un ensemble de points sur une surface plane. Bien évidemment, la représentation s'effectuant dans un plan, la notion de profondeur n'est pas conservée mais peut être obtenue par un procédé de stéréoscopie en utilisant deux vues de l'objet prises à partir de deux positions différentes. [6]

II.3.1.3. Définition 2:

- Une image numérisée correspond à une image acquise, puis traitée de façon à être stockée sous forme binaire (succession de bits à 0 ou 1). Le stockage de l'image numérique ainsi obtenue peut s'effectuer sous différents formats (jpeg, bmp, tiff, png, gif ...). Ils correspondent à des mécanismes de compression différents, mais une image de même taille que l'image originale peut être reconstituée. Cette image restituée est composée d'éléments communément appelés « pixels ». [6]

Une image peut inclure des sous-images, communément appelées ROI (Régions d'Intérêts, « Region Of Interest » en anglais), ou simplement régions. Ce concept reflète le fait qu'une image contient souvent une collection d'objets qui définissent autant de zones dans l'image.

Dans le domaine des images, on fait habituellement la distinction entre trois catégories: [5]

a. Le traitement des images :

Il permet d'opérer une transformation image \rightarrow image (en anglais on parle d'«image processing»). On désigne par traitement d'images numériques l'ensemble des techniques permettant de modifier une image numérique dans le but de l'améliorer ou d'en extraire des informations. [5]

b. L'analyse d'images :

Il permet d'opérer une transformation image \rightarrow mesures (en anglais on parle d' «image analysis»). [5]

c. L'analyse sémantique :

Il permet de créer une description de haut niveau à partir d'images (en anglais «image understanding»). [5]

Le but de l'analyse d'image réside dans la description (forme, couleur, texture) et / ou la quantification (nombre, densité) des divers objets qui composent l'image étudiée. L'analyse trouve sa place dans de nombreux domaines tels que la robotique, les sciences de la vie ou encore la surveillance routière. Les notions de segmentation et de description objective des composants formant l'image offrirait à l'expert un complément d'informations (quantifiées) lui permettant d'étayer son diagnostic. [6]

II.3.2. Caractéristiques locales et globales d'image :

La plupart des systèmes de reconnaissance d'objets ont tendance à utiliser des caractéristiques globales de l'image, qui décrivent une image dans son ensemble, ou des caractéristiques locales, qui représentent les bits de l'image ou les pièces composante de cette dernière.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

Les caractéristiques globales ont la capacité de généraliser un objet entier avec un seul vecteur. Par conséquent, leur utilisation dans la classification standard de techniques est simple. Les caractéristiques locales, d'autre part, sont calculées à de multiples points dans l'image et elles sont par conséquent plus robustes à l'occlusion (blocage) et au désordre.

Toutefois, ils peuvent exiger des algorithmes de classification spécialisés pour traiter les cas dans lesquels il y a un nombre variable de vecteurs de caractéristiques par image. [27]

Les caractéristiques locales et globales sont comme suit :

II.3.2.1. Les caractéristiques globales d'une image:

Dans ce cas, il s'agit de fournir des observations sur la totalité de l'image. L'avantage des descripteurs globaux est la simplicité des algorithmes mis en œuvre, et le nombre réduit d'observations que l'on obtient. Cependant, l'inconvénient majeur de ces descripteurs est la perte de l'information de localisation des éléments de l'image. [9]

Les avantages des caractéristiques globales sont beaucoup plus rapides et compactes et facile à calculer et nécessitent généralement de petites quantités de mémoire.

a) Texture :

La texture peut être définie comme les motifs visuels qui ont des propriétés d'homogénéité qui ne sont pas résultés de la présence de seulement une couleur ou d'intensité unique. Tamura et al (1978) ont proposé une méthode d'extraction et de la description de caractéristique de texture basée sur les études psychologiques de la perception humaine. Le procédé se compose de six éléments statistiques, y compris la grossièreté, le contraste, la directivité, la ligne-image, la régularité et de la rugosité, pour décrire diverses propriétés de texture. Cooccurrence gris matrice (GLC) est l'une des des méthodes élémentaires et importantes pour la description et extraction de caractéristiques de la texture, et Son idée originale est d'abord proposée par Julesz (1975) qui trouve à travers ses célèbres expériences sur la perception visuelle humaine de texture, que pour une grande classe de textures aucune paire de texture peuvent être discriminées si elles sont d'accord dans leurs statistiques de second ordre. [10]

b) Couleur :

La couleur est l'une des caractéristiques visuelles les plus utilisés largement dans le contexte multimédia (image / vidéo). Pour soutenir la communication sur Internet, les données doivent être adaptées à un environnement hétérogène avec une variété de plates-formes d'utilisateurs et appareils. [10]

II.3.2.2. Les caractéristiques locales d'une image :

Cette approche consiste à calculer des attributs sur des portions restreintes de l'image. L'avantage des descripteurs locaux est de conserver une information localisée dans l'image, évitant ainsi que certains détails ne soient noyés par le reste de l'image. L'inconvénient majeur de cette technique est que la quantité d'observations produite est très grande, ce qui implique un gros volume de données à traiter. [9]

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

II.3.2.3. Letraitement de caractéristiques de l'image : [10]

Dans la plupart des applications, notamment la reconnaissance d'objets, la mise en correspondance des images, les caractéristiques globales sont moins utiles que des caractéristiques locales parce qu'elles ne sont pas robustes à l'occultation⁵. Or, les objets sont souvent cachés partiellement. On cherche donc des caractéristiques locales pour représenter des objets. Le pic et la crête⁶ sont des caractéristiques locales, qui s'adaptent facilement à des applications réelles

La caractérisation d'une image peut être calculée en un nombre restreint de pixel. Il faut pour cela d'abord détecter les zones d'intérêt de l'image puis calculer en chacune de ces zones un vecteur des caractéristiques. Ces zones d'intérêt sont par exemple les arêtes, bords, coins ou blobs (petite région uniforme qui se démarque) de l'image. Il peut aussi s'agir de points pris aléatoirement ou régulièrement dans l'image (échantillonnage dit dense/serré).

Le vecteur de caractéristique contient parfois des données provenant de la détection, telles que l'orientation de l'arête ou la magnitude du gradient dans la zone d'intérêt. Généralement, le vecteur caractéristique en un pixel est calculé sur un voisinage de ce pixel, c'est-à-dire à partir d'une imagerie centrée sur ce pixel. Il peut être calculé à différentes échelles de manière à s'affranchir du facteur de zoom. Parmi les caractéristiques locales couramment calculées, on retrouve des motifs préalablement utilisés globalement, tels que des histogrammes de couleur ou des vecteurs rendant compte de l'orientation des gradients des niveaux de gris.

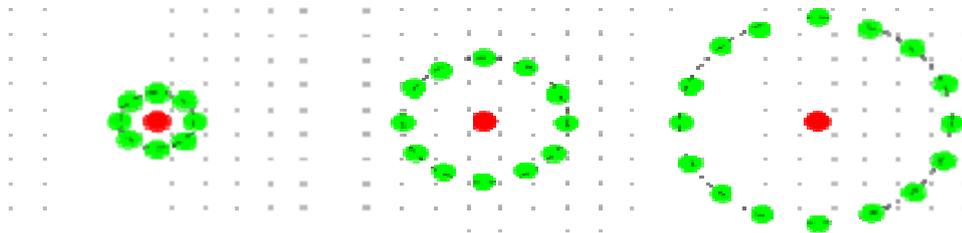


Figure II. 3: Trois exemples de voisinages utilisés pour définir une texture et Calculer un motif binaire local (LBP)

Certaines méthodes telles SIFT ou SURF incluent à la fois la détection de zone d'intérêt et le calcul d'un vecteur caractéristique en chacune de ces zones. Concernant le vecteur caractéristique, les SIFT sont grossièrement un histogramme des orientations du gradient et les SURF consistent en le calcul d'approximation d'ondelettes de Haar. Dans une veine similaire, les motifs binaires locaux sont un calcul de cooccurrence des niveaux de gris locaux d'une image et les HOG des histogrammes de gradients orientés, assez similaires aux SIFT, calculés selon un échantillonnage dense.

⁵**Occultation** : est une action de cacher quelque chose.

⁶**Un pic** ressemble à une "boule": il contient un point "sommet" et une zone de points "en dessous". On peut imaginer un pic comme une montagne isolée et plutôt ronde, alors qu'**une crête** est un chemin que l'on peut suivre "à dos de montagne".

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

Il existe un certain nombre de méthodes de détection de caractéristiques et de la description outre que ces dernières, on cite par exemple:

- ✓ Échelle Feature Invariant Transform (EIPD),
- ✓ Accélérer Caractéristiques Robuste (SURF),
- ✓ Oriented FAST et Rotated BREVE (ORB) [44], ... etc.

Le choix des caractéristiques extraites est souvent guidé par la volonté d'invariance ou de robustesse par rapport à des transformations de l'image.

II.3.3. La détection et la description et l'appariement (matching) des caractéristiques de l'image :

Au cours des dernières décennies, les détecteurs et les descripteurs de caractéristiques de l'image sont devenus des outils populaires dans la communauté de vision par ordinateur et ils sont largement appliqués dans un grand nombre d'applications : Représentation des images, la reconnaissance et d'appariement d'objets, la classification et l'extraction d'image.

Tous reposent sur la présence de caractéristiques stables et représentatives de l'image. Ainsi, la détection et l'extraction des caractéristiques de l'image sont des étapes essentielles pour ces applications. L'idée de base est d'abord détecter les régions d'intérêt (keyPoints) qui sont covariant à une classe de transformations. Ensuite, pour chacune des régions détectées, une représentation de vecteur de caractéristique invariante (c-a-d descripteur) pour des données d'image autour du point-clé détectée est construite.

Les Descripteurs de caractéristique extraite de l'image peuvent être basés sur des statistiques de second ordre, des modèles paramétriques, les coefficients obtenus à partir d'une image transformée, ou même une combinaison de ces mesures. Deux types de caractéristiques de l'image peuvent être extraites sous forme de représentation de contenu de l'image; c'est-à-dire les caractéristiques globales et caractéristiques locales.

II.3.3.1. Les types des caractéristiques de l'image :

a) Points d'intérêt :

La détection de points d'intérêts (ou coins) est au même titre que la détection de contours, une étape préliminaire à de nombreux processus de vision par ordinateur. Les points d'intérêts dans une image, correspondent à des doubles discontinuités de la fonction d'intensités. Celles-ci peuvent être provoquées comme pour les contours, par des discontinuités de la fonction de réactance ou des discontinuités de profondeur. Ce sont par exemple : les coins, les jonctions en T ou les points de fortes variations de texture. [13]



Figure II.4 : Différents types de points d'intérêts :

Coins, jonction en T et point de fortes variations de texture.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

b) Régions d'intérêt (blobs):

Une région est un ensemble de pixels connexes ayant des propriétés communes qui les différencient des pixels des régions voisines."

La définition des régions d'intérêt est donc basée sur une similarité matérialisée dans la propriété « P » définie en chaque pixel de l'image. Cette propriété peut être l'intensité du niveau de gris du pixel ou tout vecteur d'attributs (vecteurs de moments par exemple). [6]

❖ **Avantage des régions d'intérêts : [13]**

1. Sources d'information plus stable que les contours car plus de contraintes sur la fonction d'intensité.
2. Robuste aux occultations (soit occulté complètement, soit visible).
3. Pas d'opérations de chaînage (→ contours !).
4. Présents dans une grande majorité d'images (≠ contours !).

❖ **Les approches de détection des régions d'intérêts :**

Il existe différents opérateurs pour détecter les points d'intérêt. Les objectifs de ce stage sont :

- a) Établir un état de l'art des détecteurs de points d'intérêt, de leur mise en correspondance et de la manière d'évaluer ces détecteurs ;
- b) Étudier l'influence des différents opérateurs. [12]

De nombreuses méthodes ont été proposées pour détecter des points d'intérêts. Elles peuvent être classées grossièrement suivant trois catégories :

1. **Approches contours** : l'idée est de détecter les contours dans une image dans un premier temps. Les points d'intérêts sont ensuite extraits le long des contours en considérant les points de courbures maximales ainsi que les intersections de contours.
2. **Approches intensité** : l'idée est cette fois-ci de regarder directement la fonction d'intensité dans les images pour en extraire directement les points de discontinuités.
3. **Approches à base de modèles** : les points d'intérêts sont identifiés dans l'image par mise en correspondance de la fonction d'intensité avec un modèle théorique de cette fonction des points d'intérêts considérés.

Les approches de la deuxième catégorie sont celles utilisées généralement. Les raisons sont : indépendance vis à vis de la détection de contours (stabilité), indépendance vis à vis du type de points d'intérêts (méthodes plus générales). [13]

c) Contours:

Les bords sont des points où il y a une limite (ou un bord /bordure) entre deux zones d'image. En général, une bordure peut être de forme presque arbitraire et peut comprendre des jonctions. Dans la pratique, les bords sont généralement définis comme des ensembles de points dans l'image qui ont une forte amplitude du gradient. De plus, certains des algorithmes communs vont alors enchaîner des points à gradient élevé pour former une description plus complète d'une bordure. Ces algorithmes placent généralement des contraintes sur les propriétés d'une bordure, comme la forme, et la valeur de gradient.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

II.3.3.2. Détection and description des caractéristiques /detectionfeatures and description :

La détection de caractéristiques sélectionne les régions d'une image qui ont un contenu unique, tels que les coins ou blobs. Utiliser la détection de caractéristique pour trouver des points d'intérêt d'une image lors d'un traitement ultérieur. Ces points ne correspondent pas nécessairement à des structures physiques, telles que les coins d'une table. La clé de détection de caractéristiques est de trouver des caractéristiques qui restent localement invariant afin de pouvoir les détecter même en présence de rotation ou de changement d'échelle.

La description des caractéristiques comme la détection des caractéristiques, est une des étapes importantes dans la vision par ordinateur, il vise à attribuer un vecteur de description à chaque keypoint détecté. SIFT est l'une de plusieurs méthodes qui fournissent un détecteur de caractéristiques, il calcule l'ampleur gradient et orientation gradient pour créer un ensemble d'histogrammes sur une fenêtre centrée sur chaque keypoint identifié.

Inspiré par SIFT, des méthodes avancées ont été développées telles que SURF, histogramme des gradients orientés (HOG) et Gradient emplacement et l'orientation Histogramme (GLOH)

II.3.3.3. Feature Extraction :

Après la détection, l'application d'un algorithme d'extraction de caractéristiques à une image. Cela consiste à calculer sur chaque zone détectée ce que l'on appelle un vecteur caractéristique, qui d'une certaine façon « résumera » le contenu de la zone en question, sous un point de vue particulier. Parmi les méthodes populaires on peut citer les dérivées partielles d'ordre N (N-jets) ou les histogrammes locaux (histogrammes de couleur, ou d'orientation des contours par exemple). D'autres algorithmes tels que SIFT comprennent à la fois une phase de détection et une phase d'extraction.

Le vecteur caractéristique contient parfois des données issues directement de la détection, telles que l'orientation de l'arête ou la magnitude du gradient au point d'intérêt.

Les vecteurs caractéristiques constituent une façon de décrire numériquement le contenu d'une image. De ce fait, ils sont souvent utilisés par des algorithmes plus globaux tels que la comparaison d'images ou la recherche d'images par le contenu. [11]

Il y a quatre aspects de l'extraction de caractéristiques:

- ❖ Construction de caractéristique;
- ❖ Génération d'un sous-ensemble de caractéristique (ou stratégie de recherche);
- ❖ L'évaluation de la définition du critère (par exemple l'indice de pertinence ou de la puissance prédictive);
- ❖ Estimation du critère d'évaluation (ou méthode d'évaluation).

II.3.3.4. L'appariement de caractéristiques /Featurematching :

La correspondance ou l'appariement de Caractéristiques ou généralement l'image correspondante, une partie de nombreuses applications de vision par ordinateur telles que l'enregistrement de l'image, le calibrage de la caméra et la reconnaissance d'objets, qui est aussi la tâche d'établir des correspondances entre deux images de la même scène / objet. Une approche commune à l'image correspondante consiste à détecter un ensemble de points d'intérêt associés chacun à descripteurs d'image à partir des données d'image.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

Une fois les caractéristiques et leurs descripteurs ont été extraits à partir de deux ou plusieurs images, l'étape suivante consiste à établir quelques correspondances de caractéristiques entre ces images.

II.3.3.5. La différence entre la détection et extraction et l'appariement de caractéristiques :

- La détection de caractéristique, appelée aussi la détection des points d'intérêt ou la détection de keypoint est de trouver des points dans l'image qui sont en quelque sorte «spécial». Typiquement, cela signifie que ces points correspondent à des éléments de la scène qui peuvent être localisés de manière fiable dans des vues différentes de cette scène.
- L'extraction de caractéristique veut dire le calcul d'un descripteur à partir des pixels autour de chaque point d'intérêt. Le descripteur le plus simple est juste les valeurs brutes de pixels dans une petite pièce ou bit autour du point d'intérêt. descripteurs plus sophistiqués comprennent SURF, HOG et FREAK.
- Matchingdescriptor ou la correspondance des descripteurs est un moyen de trouver la correspondance de point entre deux images.

Chapitre 2 : Introduction à la découverte d'information à base de contenu multimédia

Conclusion :

Les orientations contenues dans le présent chapitre viennent nous éclairer respectivement sur le concept du media vidéo et de l'image qui constitue la vidéo, pour une meilleure compréhension de l'origine de l'information.

Dans le même ordre d'idées, on a expliqués les caractéristiques locales et globales de l'image, ainsi que le traitement appliqué sur la détection de point d'intérêt et la description.

De ce fait, à la lumière de ce qui précède, l'extraction de caractéristiques, facilite l'étude de la reconnaissance et la détection des plaques d'immatriculations.

Le contenu du chapitre suivant sera sur le processus de détection des plaques d'immatriculations.

Introduction :

Les véhicules ont été largement utilisés dans tous les domaines de la production et dans la vie quotidienne, et le numéro d'immatriculation représente un moyen efficace pour identifier les véhicules. Il s'agit d'une information unique pour chaque voiture. Avec le nombre de véhicules qui augmente rapidement, les violations du trafic apparaissent plus fréquemment dans la circulation publique, comme les fraudes aux péages dans les autoroutes ou les parkings, excès de vitesse, les vols des voitures, etc.

Fréquemment, il est nécessaire d'identifier les plaques d'immatriculation des véhicules pour la sécurité, avec le développement rapide des routes et la large utilisation des véhicules, les spécialistes routiers ont commencé à s'intéresser aux avantages et à l'efficacité des Systèmes Intelligents de Transport. Récemment la nécessité de la Reconnaissance des Plaques d'Immatriculation de Véhicules a augmenté de façon très significative. En général, tout système de reconnaissance de plaque d'immatriculation se compose de trois parties: détection de la plaque du véhicule, localisation de la plaque d'immatriculation et l'identification des numéros de la plaque (reconnaissance).

Le but de ce troisième chapitre est d'expliquer et de citer les différentes méthodes de détection de plaques d'immatriculations et de connaître les domaines et les divers systèmes de lecture de plaques d'immatriculation.

III.1. La lecture des plaques d'immatriculation :

III.1.1. Lecture automatique des plaques d'immatriculation : [3]

La lecture automatisée des plaques d'immatriculations est une méthode d'identification qui utilise des techniques de traitements d'images et de vision par ordinateur pour lire les plaques d'immatriculation de véhicules. Elle est une méthode de surveillance de masse qui utilise la technique de la reconnaissance optique de caractères sur des images pour lire les plaques d'immatriculation de véhicules.

Elle peut être utilisée aussi bien pour afficher les images capturées par les caméras que le texte de la plaque d'immatriculation, avec la possibilité de stocker une photographie du conducteur. Ces systèmes recourent largement au rayonnement infrarouge pour permettre à la caméra de photographier à toute heure du jour.

Dans leur version la plus simple, les systèmes de LAPI utilisent en cascade deux techniques de traitements d'images : la détection d'objet pour repérer les plaques d'immatriculation potentielles suivie de la reconnaissance optique de caractères afin d'identifier les caractères alphanumériques de la plaque.

Pour obtenir des résultats satisfaisants, ce sont des séries de techniques de traitement d'image qui sont appliquées afin de détecter, normaliser et agrandir l'image de la plaque d'immatriculation, et enfin, de pouvoir réaliser la reconnaissance optique de caractères (OCR) pour extraire les caractères alphanumériques de la plaque.

III.1.2. Phase de détection : [3]

La phase de détection est généralement réalisée en appliquant des algorithmes de reconnaissance de forme, qui se basent sur des descripteurs idoines (propre à) de la plaque ou sur la comparaison de modèles. A ce stade, la plaque n'est pas recherchée dans le détail, mais dans son aspect général. Cette phase peut alors se faire sur une image sous-échantillonnée afin d'accélérer le processus. La normalisation consiste à appliquer des méthodes de restauration sur les plaques potentiellement détectées.

Cette phase se fait après un recalage de la plaque détectée sur l'image d'origine (non-échantillonnée), on peut citer l'application d'une série de transformations affines, afin de remettre en forme la plaque (et selon une taille prédéfinie) qui est naturellement déformée par les effets de perspectives inhérentes à la prise de vue, ensuite des traitements de restauration des contrastes et divers restaurations des contours (dilatations, érosions, ...) de la plaque sont appliqués, celle-ci pouvant avoir subi des occlusions dues à des salissures ou des masquages.

III.1.3.Phase de reconnaissance : [3]

La phase précédente permet de préparer au mieux la phase de reconnaissance optique des caractères. Cette dernière utilise généralement des mesures de distance ou de comparaison entre des modèles permettant de réaliser la classification des caractères alphanumériques des plaques et de reconstituer l'intégralité du numéro de la plaque. La fiabilité et la précision du système dépendent de la complexité de chacune de ces phases.

III.1.4.Algorithmes de lecture des plaques d'immatriculation : [4]

La lecture des plaques d'immatriculation nécessite une exécution de cinq algorithmes primaires pour que le logiciel puisse l'identifier:

1. Localisation de la plaque : responsable de trouver et d'isoler la plaque sur l'image,
2. Orientation et dimensionnement de la plaque : correction de l'orientation de la plaque et ajustement des dimensions à la taille nécessaire,
3. Normalisation : ajustement de l'intensité et du contraste de l'image,
4. Segmentation des caractères : localisation des caractères sur la plaque,
5. Reconnaissance optique de caractères (OCR).

La complexité de chacune des étapes du programme détermine la précision du système, pendant la troisième phase (normalisation), certains systèmes utilisent des techniques de détection de contour pour augmenter le contraste entre les lettres et la couleur de fond de la plaque. Un filtre numérique peut aussi être utilisé pour réduire le « bruit » visuel de l'image.

III.2.Systèmes de lecture des plaques d'immatriculation :

III.2.1.LAPI_ENGINE : [2]

Le produit LAPI ENGINE représente le cœur technologique permettant la Lecture Automatique de Plaques d'Immatriculation qui a été intégré dans l'ensemble de solutions. Principalement dédiée à la traçabilité de véhicules, est un OCR de lecture de plaques d'immatriculation et un produit autonome dédié principalement à la traçabilité de véhicules et pouvant s'adapter à un large éventail d'applications. Ce produit s'adresse aux professionnels souhaitant disposer d'un OCR spécialisé LAPI afin de l'intégrer dans divers projets.

Chapitre III : Détection de plaque d'immatriculation de véhicule

LAPI ENGINE est un produit autonome pouvant s'adapter à un large éventail d'applications. Il s'intègre facilement dans des solutions "métiers" comme des logiciels de contrôle d'accès, de pesée industrielle ou de vidéosurveillance, Intégrant de nombreuses innovations algorithmiques, et permet la reconnaissance en temps réel des plaques de tous types de véhicules, de jour comme de nuit se présentant devant les caméras avec un taux de reconnaissance optimal. Les fichiers XML et JPG peuvent être envoyés à l'application "Métier" par l'intermédiaire d'alertes automatiques composées d'une adresse IP, d'un numéro de port et d'un entête.

❖ Fonctionnement :

Le logiciel LAPI-ENGINE permet d'identifier et de tracer les véhicules passant sur une zone et se présentant devant une caméra, à partir d'un flux vidéo en direct, enregistrés ou bien des séquences vidéo d'évènements détectées par les caméras. Après l'extraction du numéro de la plaque analysée, chaque évènement (passage de véhicules) est tracé et enregistré dans un fichier XML accompagné d'une photo JPG horodatée de l'avant ou arrière du véhicule. Les informations récupérées dans ce fichier après leurs envois aux utilisateurs par ce dernier peuvent ensuite être couplées avec d'autres données (Numéro de la plaque, Date et heure de passage, Référence caméra, Indicateur Entrée / Sortie, Indice de confiance de la reconnaissance effectuée).



Figure III.1 : Architecture de LAPI-ENGINE.

III.2.2. LAPI_NETWORK : [2]

LAPI-NETWORK est un logiciel permettant, dans le cas d'un déploiement de plusieurs LAPI-ENGINE sur différents sites, de récupérer les évènements détectés localement. Les différents fichiers XML et JPG sont envoyés automatiquement dans un répertoire dédié et peuvent ainsi être gérés par LAPI-NETWORK.

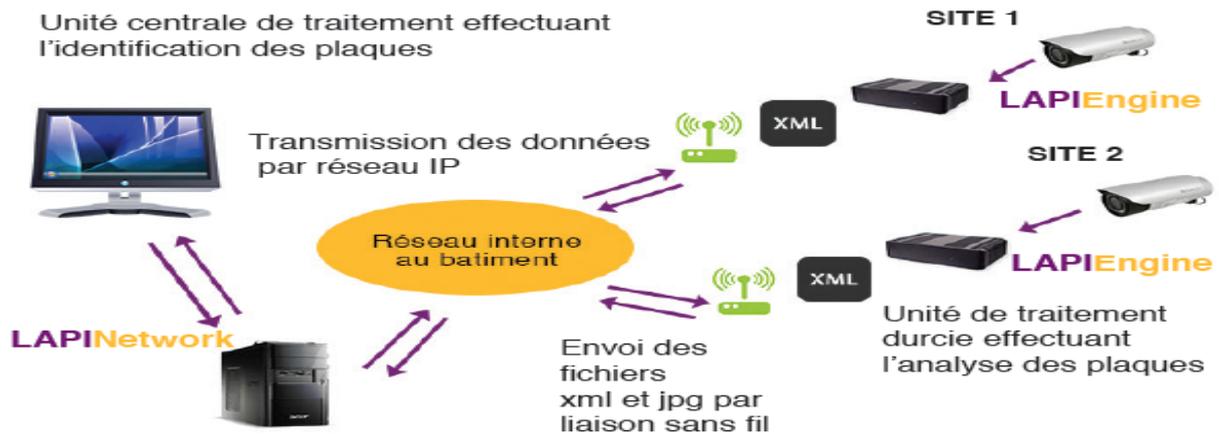


Figure III.2 : Architecture de LAPI-NETWORK.

❖ Caractéristiques et fonctionnalités :

- Affichage en temps réel des évènements détectés,
- Recherche d'un numéro d'immatriculation sur une période donnée, pour un site local ou l'ensemble des sites,
- Export des évènements par fichier CSV ou PDF accompagné de la photo horodatée du véhicule,
- Gestion de listes noires et paramétrage d'alarmes (mail, son, fenêtre Pop up).

III.2.3. Systèmes de détection et de reconnaissance de plaque d'immatriculation :

III.2.3.1. LAPI-ACCES: [2]

LAPI-ACCES est un logiciel de contrôle d'accès pour la reconnaissance de plaques d'immatriculation par système automatique s'adressant au marché de la sécurité d'accès. Le logiciel de reconnaissance de plaques d'immatriculation LAPI-ACCES intègre de nombreuses innovations algorithmiques permettant la reconnaissance en temps réel de jour comme de nuit des plaques d'immatriculation de tous types de véhicules (autos, motos, poids lourds) passant dans le champs de vision des caméras et permet la gestion des autorisations et l'enregistrement des plaques d'immatriculation des véhicules entrant et/ou sortant d'une zone.

LAPI ACCES est un produit autonome et indépendant du matériel disposant d'un taux de reconnaissance optimal.

❖ Fonctionnement :

- Le système LAPI ACCES permet d'identifier et d'enregistrer les véhicules qui se présentent devant les caméras placées aux entrées et/ou sorties d'un site. Une base de données permet d'enregistrer au préalable les listes des plaques d'immatriculation à contrôler ; les informations relatives au conducteur et à leurs véhicule sont enregistrées dans une fiche où l'on retrouve : nom et prénom du propriétaire, le n° d'immatriculation du véhicule, la date et l'heure de début et fin d'autorisation d'accès sur la zone, les informations sur le véhicule. Les autorisations d'accès sont paramétrables depuis chaque "fiche véhicule" par date ou selon les jours de la semaine, l'accès à une zone est autorisé ou refusé en fonction de la reconnaissance de plaques

Chapitre III : Détection de plaque d'immatriculation de véhicule

d'immatriculation qui est faite. L'exploitant peut ainsi visualiser en temps réel les entrées et les sorties autorisées et les refus d'accès afin de mieux gérer l'intrusion d'usagers.

- La plaque reconnue est ensuite comparée avec la base de données. Selon le statut d'autorisation affecté au véhicule, un système automatique de barrières se déclenche ou non. L'ensemble des données récupérées lors de la détection d'une plaque peut être exporté vers un autre système d'informations, l'utilisateur peut également effectuer des recherches selon les critères suivants : N° d'immatriculation, date et heure de début et fin d'autorisation.

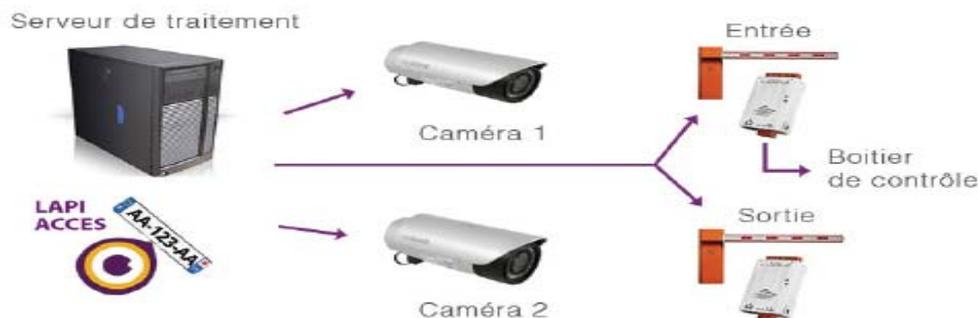


Figure III.3 : Architecture de LAPI-ACCES.

III.2.3.2. LAPI_SMARTPHONE : [2]

LAPI-SMARTPHONE est une application mobile permet la détection et la reconnaissance automatique de plaques d'immatriculation et de gérer des problématiques de gestion et de contrôle de stationnement, à partir de l'analyse d'une photo ou d'un flux vidéo en temps réel. Qui est dédiée principalement à la gestion et au contrôle de stationnement, cette application permet le contrôle d'une plaque d'immatriculation par rapport à une base de données locale ou distante et s'intègre facilement à d'autres systèmes de gestion.

❖ Fonctionnalités :

- ✓ Récupération du numéro de la plaque, ainsi que la date et l'heure de la prise de photo,
- ✓ Identification de la nationalité,
- ✓ Gestion des fiches client et interrogation à distance d'une base de données pour la gestion des autorisations de stationnement, Si "AUTORISE" alors les informations concernant le propriétaire du véhicule s'affichent.

III.3.Méthode pour la détection des plaques d'immatriculation : [5]

Les algorithmes de la reconnaissance de plaques d'immatriculation LPR (License plate recognition) sont généralement constitués de trois étapes de traitement:

1. L'extraction et l'emplacement des régions de la plaque d'immatriculation;
2. La segmentation des caractères de la plaque;
3. La reconnaissance de chaque caractère.

Les techniques pour détecter LPs (Licence plate / les plaques d'immatriculations) dans une séquence d'images ou de vidéo unique sont basées sur la terminologie technique en "temps réel", l'opération de reconnaissance de la plaque d'immatriculation (LPR) représente une opération

assez rapide pour ne pas manquer un seul objet d'intérêt qui se déplace à travers la scène. Néanmoins, la croissance exponentielle de la puissance du traitement, les derniers développements fonctionnent à moins de 50 ms pour la détection et la reconnaissance de plaque.

En ce qui concerne l'extraction de la région de la plaque d'immatriculation concernée, une catégorisation des méthodes a été exposée dans la littérature, en même temps que la description principale de méthode de traitement, il convient de noter que certaines méthodes peuvent être classées dans plus d'une catégorie et que les frontières entre les sous-catégories ne sont pas toujours sans ambiguïté.

III.3.1. Traitement binaire de l'image :

Les techniques basées sur des combinaisons de statistiques de pointe et de la morphologie mathématique ont présentées de très bons résultats, l'amplitude du gradient et de la variance locale d'une image sont calculés sur la base du principe selon lequel le changement de luminosité dans la région de LP est plus remarquable et plus fréquents qu'ailleurs. Plusieurs méthodes sont présentées pour ce traitement, par d'exemple :

➤ L'algorithme d'extraction hybride LP :

L'algorithme d'extraction hybride LP basé sur les statistiques de pointe et de la morphologie de la surveillance des systèmes de billetterie autoroute est proposé, cette approche peut être divisée en quatre sections suivantes: détection de bord vertical, bord de l'analyse statistique, l'emplacement de LP a base hiérarchique à base LP et d'extraction LP a base de la morphologie. La précision moyenne de localisation d'un véhicule (LP) est un taux impressionnant de 99,6% (9786 à partir de 9825 images). Les images numériques ont été acquises à partir d'une distance et d'un angle fixe et par conséquent, des régions candidates dans une position spécifique ont une priorité comme déjà décrit. Cette connaissance a priori serait certainement stimuler les résultats à un niveau élevé de précision.

➤ Analyse de composants (Connected component analysis) CCA :

Est une technique vitale dans le traitement de l'image binaire qui scanne une image déjà binarisée¹ et les étiquettes de ses pixels en composants basés sur la connectivité de pixel (Soit 4 connecté ou habituellement, 8 connectés). Une fois que tous les groupes de pixels ont été déterminés, chaque pixel est étiqueté avec une valeur en fonction du composant auquel a été attribué. Extraction et étiquetage des différents composants disjoints et connectés dans une image est fondamentale pour beaucoup d'applications automatisées d'analyse d'image ; autant de mesures utiles et fonctionnalités des objets binaires peuvent être extraits, des mesures spatiales telles que la surface (zone), l'orientation et le rapport d'aspect (AR) ne sont que quelques-unes des caractéristiques souvent intégrées dans les algorithmes de traitement d'image pour la détection de LP. Ensuite, en utilisant des techniques de filtrage simples, les objets binaires avec des mesures qui dépassent les limites souhaitées peuvent être éliminés dans les prochaines étapes de l'algorithmiques.

¹En traitement d'image, la binarisation est une opération qui produit deux classes de pixels noirs et blancs.

III.3.2. Traitement au niveau de gris :

Le grand contraste entre les caractères et l'arrière-plan est exploitée pour détecter les plaques d'immatriculation avec des caractères noirs sur un fond blanc. Tandis que certains autres algorithmes ont supposés que la densité des bords dans la région de la plaque d'immatriculation est plus grande que d'autres régions si le contraste du caractère et de la plaque d'immatriculation est suffisamment grand.

Par exemple, ils ont scanné les images du véhicule avec la distance N-rangée pour compter les bords existants. Les régions à forte densité de bord auront probablement la plaque d'immatriculation à l'intérieur.

➤ Suivi d'un objet probabiliste dans une vidéo : (Probabilistic Object Tracking in Vidéos)

Une nouvelle méthode d'estimation de la position et le suivi des PLs dans vue de la caméra monoculaire en 3-D a été récemment proposée par Yalçınand Gökmen in [106]. Une estimation initiale a été donnée, la position de la plaque est localisée dans les frames successives de la vidéo. Pour estimer l'objet et filtrer les mesures, les développeurs ont développé un nouvel algorithme composé de la propagation de la densité de probabilité de l'algorithme de condensation et une étape d'optimisation à petite échelle en fonction de l'algorithme à évolution différentielle DE, d'après les expériences réalisées dans [106], l'algorithme DE- condensation est plus performant que l'algorithme de condensation standard, ce qui diminue le temps de calcul d'environ 35%, pour le même niveau de précision de localisation..

III.3.3. Traitement des images couleurs :

Dans de nombreux pays ou régions, le format des plaques d'immatriculation est strictement appliqué. La couleur du texte et du fond est fixé, de sorte que de nombreux algorithmes utilisent des informations de couleur pour détecter les plaques d'immatriculation. Toutefois, si les conditions d'éclairage changent, la couleur des plaques d'immatriculation varie. Ainsi, les algorithmes de détection de plaque d'immatriculation qui reposent uniquement sur les informations de couleur ne peuvent pas atteindre des taux de détection élevés. Et pour les pays comme les États-Unis, car il existe une grande variété de plaques d'immatriculation, de telles méthodes ne peuvent pas être appliquées.

➤ Modèle de Transformation de Couleur:

L'idée de base de l'extraction d'une région de la plaque est que la combinaison de couleurs d'une plaque (arrière-plan) et le caractère (au premier plan) est unique, et cette combinaison se produit presque exclusivement dans une région de la plaque. Comme les PLs chinoises ont des formats spécifiques. Shi et al. ont proposé que tous les pixels dans l'image d'entrée doivent être classés en utilisant la couleur du modèle HLS² dans les 13 catégories suivantes: 1) bleu foncé; 2) bleu; 3) bleu clair; 4) jaune foncé; 5) jaune; 6) jaune clair; 7) dark noir; 8) noir; 9) gris noir; 10) gris blanc; 11) blanc; 12) la lumière blanche; et 13) les autres. Après la classification d'une région dans les couleurs ci-dessus, l'AR de la plaque attendue est ensuite vérifiée.

➤ Traitement d'histogramme (HistogramProcessing) :

² **HSL ((en) Hue, Saturation, Lightness)** teinte-saturation-luminosité désignent des systèmes de description des couleurs.

Chapitre III : Détection de plaque d'immatriculation de véhicule

Un histogramme est un graphe qui représente le taux de quoi et que ce soit. Les histogrammes ont des barres qui montrent le taux d'occurrence des données dans l'ensemble des données.

Histogramme représente le taux de valeurs d'intensité de pixels. Dans un histogramme de l'image, l'axe des abscisses représente les intensités de niveau de gris et l'axe des ordonnées représente la fréquence de ces intensités.

Les histogrammes prennent ont un rôle fondamental du traitement de l'image, dans la région tels que le développement, la segmentation et l'explication.

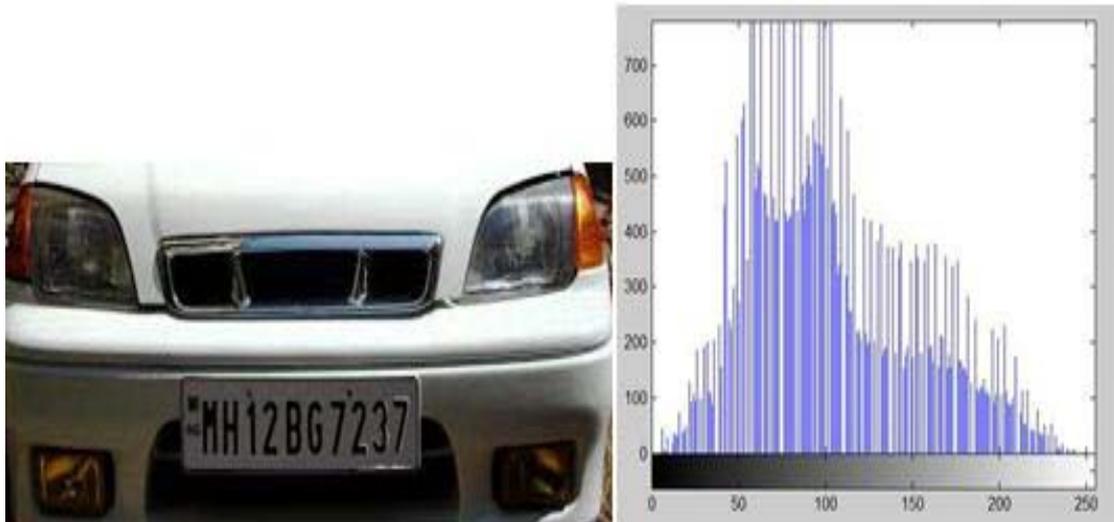


Figure III.4 : Un exemple de détection de la plaque d'immatriculation en utilisant les histogrammes.

III.4. Méthodes utilisé pour la détection des plaques d'immatriculation à grande échelle :

Dans les villes de taille moyenne, plus de 10000 photos sont capturées à chaque grande route urbaine en utilisant des caméras haute définition par heure. Très souvent, ces photographies impliquent des informations importantes pour trouver un crime de voiture avec plaque d'immatriculation. [6]

À temps présent, le policier sélectionne ou trouve un certain type de voiture, par exemple PASSAT noir, par «yeux de l'homme/humain ». Ce processus de récupération d'image n'est pas efficace. De l'expérience montre qu'il en coûterait plus de 4 heures pour une personne qui est très familiarisée avec les types de voitures pour sélectionner sur 15000 photos. Pour cela les chercheurs ont développé quelques auto-méthodes de détection de voiture et la reconnaissance de plaque d'immatriculation. En utilisant les caractéristiques de la bande de voiture à des images statiques. [6] Ces méthodes peuvent prendre en compte des caractéristiques spécifiques du véhicule comme le rapport de largeur/hauteur, la distance entre le capteur et la source, la forme ...etc.

❖ Méthodes d'accélération du processus de détection de PLs :

- **L'utilisation d'une méthode efficace de reconnaissance des plaques d'immatriculation des véhicules basée sur réseaux de neurone artificiel et fenêtre coulissante concentriques : [7]**

Dans cet article, ils utilisent la technique de segmentation nommée fenêtres coulissantes concentriques (SCW). Cette méthode est pour analyser les images de la route qui contiennent souvent des véhicules et extrait la plaque d'immatriculation à partir des propriétés naturelles en trouvant des bords horizontaux et verticaux de la région du véhicule.

À la base d'une nouvelle technique adaptative de segmentation de l'image qui est utilisée pour la détection et la vérification de couleur des régions candidates en utilisant HSI un modèle de couleur sur la base d'utilisation de la teinte (hue) et l'intensité dans le modèle de vérification de couleurs, LP verte et jaune et LP blanche respectivement.

- **L'utilisation des algorithmes parallèle: [10]**

Le traitement de l'image de voiture est très long. Un algorithme parallèle exécuté sur la haute performance d'ordinateurs est une bonne approche pour réduire le temps. L'architecture du dernier super-ordinateur est de plus en plus complexe. Beaucoup de supercalculateurs dans Top500- liste sont des architectures hétérogènes. Tels que le Tianhe-2 - MilkyWay-2 (TH1VB-FEE Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P) et le Titan - Cray XK7 (Opteron 6274 16C 2.200GHz, Cray Gemini interconnexion, NVIDIA K20x).

- **L'utilisation des approches de la programmation hétérogène : [11]**

La programmation sur superordinateur hétérogène est très difficile. Il existe deux approches pour résoudre ce problème. La première est que les maîtres programmeurs avancés sur les connaissances de l'architecture de superordinateur. La seconde consiste à construire un sympathique environnement de programmation pour faciliter la difficulté de programmation parallèle. Cette approche construit un environnement en nuage pour soutenir la programmation dans une langue plus facile, tel que Julia. Julia est un langage de programmation dynamique de haut niveau pour le calcul technique, avec une syntaxe qui est familière aux utilisateurs d'autres environnements de technique informatique. Il fournit une exécution parallèle distribuée, précision numérique, et une vaste bibliothèque de fonctions mathématiques.

- **L'utilisation de la forme de correspondance et la reconnaissance d'objet en utilisant les contextes de forme: [13]**

L'utilisation des descripteurs de forme, nommés contextshapes, décrivant la distribution du reste de la forme par rapport à un point donné sur le contour. Chercher la correspondance entre deux formes est alors équivalent à trouver pour chaque point sur une forme le point sur l'autre forme qui a un contextshape similaire. Les auteurs prolongent la correspondance à la forme entière en estimant une transformation qui aligne la carte de contextshapes d'une forme sur l'autre.

- **L'utilisation de ladétection de la plaque d'immatriculation basée sur une Morphologie à partir de scène complexes: [14]**

Une autre approche utilise les opérations morphologiques sur l'image au niveau de gris. Cette approche se concentre sur les propriétés d'apparence locale des régions des plaques d'immatriculation telles que la luminosité, la symétrie et l'orientation. Les régions candidates sont comparées avec une image de plaque d'immatriculation donnée basée sur la similarité de ces propriétés d'apparence. En particulier, l'algorithme de recherche de projection est utilisé pour détecter les régions des caractères de la plaque d'immatriculation à travers des projections verticales et horizontales.

- **L'utilisation d'une nouvelle méthode pour reconnaître le numéro de plaques d'immatriculation de véhicules : [8]**

En premier, les plaques d'immatriculation sont localisées en utilisant les principales caractéristiques. Ensuite, chacun des sept caractères dans les plaques d'immatriculation est segmenté et enfin, le système de reconnaissance de caractères extrait certains caractéristiques de ces caractères et utilise une caractéristique principale pour atteindre le résultat de reconnaissance robuste.

- **L'utilisation de ladétection de plaque d'immatriculation de voiture à partir de scènes complexes : [15]**

Les couleurs et les textures des plaques d'immatriculation ont été utilisés pour identifier les plaques d'immatriculation de véhicule (PIV), mais cela semble être inefficace lorsque la plaque a des couleurs et des motifs différents. Par exemple, les machines à vecteurs de support (SVM) sont utilisées pour analyser les propriétés de couleur de texture des PIV. Les valeurs de couleur des pixels bruts qui composent le modèle de couleur de texture sont données directement à la SVM. Ensuite, les régions de la plaque d'immatriculation sont identifiées en appliquant l'algorithme (CAMShift, a continuously adaptive mean shift algorithm) sur les résultats de l'analyse de la couleur et de la texture.

- **L'utilisation de ladétection de plaque d'immatriculation de véhicule basé sur les parallélogrammes et les histogrammes : [16]**

La méthode de détections des plaques d'immatriculation de véhicule basée sur le parallélogramme et l'histogramme a été présentée en 2008. La technique est constituée de trois étapes principales :

1. Application d'une technique de segmentation nommée fenêtres coulissantes concentriques (Sliding concentric windows, *SCW*) sur l'image pour détecter la région candidate,
2. Raffinement en utilisant le modèle de couleur *HSI* (*Hue, Saturation and Intensity*/ Teinte, Saturation et Intensité) sur cette base en utilisant la teinte (*Hue*) et l'intensité du modèle de couleur *HSI* pour vérifiant les plaques d'immatriculation verte, jaune et blanche, respectivement,
3. Enfin, la vérification et la détection de la région de la plaque d'immatriculation, qui contient des caractères alphanumériques prédéterminés de la PI en utilisant l'histogramme de position.

- **L'utilisation de l'algorithme de reconnaissances de plaque d'immatriculation pour un système d'applications de transport intelligent : [18]**

Une approche basée sur la logique floue et les réseaux de neurones pour la reconnaissance des plaques d'immatriculation de véhicule a été proposée. Cette méthode utilise la logique floue pour la segmentation et les réseaux de neurones cellulaires à temps discret (DTCNN, discrete-time cellular neural networks) pour l'extraction des caractéristiques.

- **L'utilisation de la marque automatique et la reconnaissance du modèle à partir d'images de Frontal des voitures : [9]**

Savoir la classe générique d'un véhicule ce n'est pas la précision suffisante pour trouver des voitures de la criminalité. Les véhicules doivent être classés en particulier par 'la marque et le modèle' (MMR, Marque et reconnaissance Model).

- **L'utilisation de l'Approche d'objet qui consiste à récupérer un texte dans la vidéo : [12]**

Quand les bases d'images sont grandes, les techniques de mise en correspondance deviennent inopérantes car trop gourmandes en calcul. La méthode bag-of-visual-words consiste alors à agréger les détecteurs locaux selon un vocabulaire prédéfini (le dictionnaire). Inspiré de la technique connue depuis longtemps dans le domaine du texte (bag-of-Word), elle a été proposée en 2003 pour la vidéo.

- **L'utilisation du couple phase de détection/ phase de reconnaissance : [3]**

LAPI est couplée à une phase de détection/reconnaissance de véhicule afin de renforcer les performances de la détection de plaque. Afin d'accélérer et d'améliorer les performances de la phase de détection des plaques, une technique courante est d'utiliser un a priori sur les zones potentielles de détection et de réduire la recherche à celles-ci. Ces zones sont déterminées par l'analyse des localisations des détections précédentes. Cette technique est appliquée dans le cadre des caméras ayant une prise de vue large (plusieurs voies). L'application de cette technique permet d'accélérer les traitements et donc d'augmenter le nombre de plaque lisible à la seconde. Cela est utile dans le cas où les dispositifs doivent travailler dans un environnement proche du temps réel tout en ayant une puissance de calcul modérée. Toutefois, elles s'exposent également au risque de ne pas détecter les véhicules qui ne passent pas dans les zones de pré-détection. C'est pourquoi elles ne sont pas appliquées dans le cas des dispositifs qui nécessitent d'identifier toutes les plaques à des fins de sécurité. Dans ces cas-là, soit le traitement est fait hors ligne, en temps différé, ou bien en temps réel mais avec un système plus coûteux car nécessitant une puissance de calcul élevée.

- **L'utilisation de la plaque d'immatriculation du véhicule, marque et modèle de reconnaissance : [19]**

L'algorithme qui définit la tâche de détection comme un problème stimulant (boosting problem). Sur plusieurs itérations, le classificateur AdaBoost sélectionne le classificateur le plus performant parmi un ensemble de faibles classificateurs, chacun agit sur une seule caractéristique, et une fois formés après apprentissage, il les combine d'une manière pondérée. Ce classificateur est alors appliqué à des sous-régions d'une image en cours de numérisation pour les zones

susceptibles de contenir une plaque d'immatriculation. Une optimisation basée sur une cascade de classificateurs, est utilisé afin d'accélérer le processus de numérisation.

III.5. Domaine d'application de la lecture des plaques d'immatriculation : [1]

La lecture automatique de plaques d'immatriculations peut également être utilisée pour :

- ✓ Les passages de frontière ou l'accès à des zones sécurisées et sensibles ;
- ✓ Les stations-service (enregistrement quand un client part sans payer) ;
- ✓ Le contrôle d'accès des parkings ou routes privées: ouverture automatique, ou enregistrement de l'entrée (votre plaque est imprimée sur les tickets de parking de l'aéroport de Roissy-CDG);
- ✓ Un outil de marketing pour enregistrer les modes de consommation ;
- ✓ Les systèmes de gestion de la circulation, qui calculent la vitesse de circulation en mesurant le temps entre les passages devant deux points de lecture ;
- ✓ La gestion de stationnement (parking, parc automobile, zone urbaine de stationnement) ;
- ✓ Le contrôle d'accès au péage
- ✓ Comparer les plaques d'immatriculations au Fichier des véhicules volés (FVV) ;
- ✓ Comparer les plaques d'immatriculations au Système d'Information Schengen ;

Conclusion :

Dans ce chapitre nous avons présentés un aperçu général sur le domaine d'application de la lecture des plaques d'immatriculation de véhicule et de mettre les points sur quelques systèmes qui permettent la lecture, pour cela on a évoqué un certain nombre de notions et d'éléments concernant la phase de la détection et la phase de la reconnaissance du domaine de la lecture des plaques d'immatriculation, les méthodes de la détection des plaques d'immatriculation de véhicule aux différents niveaux de traitement de l'image en générale et celle utilisé pour la détection des plaques d'immatriculation à grand échelle.

Après avoir acquistous les concepts du domaine de détection d'immatriculation, le prochain chapitre sera consacré pour la détection d'immatriculation de véhicule basée sur la technique de MapReduce et la réalisation de notre application.

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

Introduction:

Le système de reconnaissance de plaques d'immatriculation est une technologie de traitement d'image utilisée pour identifier les véhicules grâce à leurs plaques d'immatriculation. Cette technologie est utilisée dans diverses applications de sécurité et de circulation.

Les systèmes existants de détection et de reconnaissance de plaque d'immatriculation peuvent être divisés en deux grandes catégories: la voie et sur la route [1]. La plaque ne serait pas détectée si la voiture n'est pas en position correcte, et pour corriger ce problème une recherche a été faite et testée pour modifier le processus d'acquisition d'image en utilisant l'enregistrement vidéo.

Dans ce chapitre on a utilisé une méthode existante pour détecter le rectangle encadrant la plaque d'immatriculation d'un véhicule dans une séquence vidéo qui est un algorithme qui base sur MapReduce tout en passant par un processus de traitement d'image et elle est Considérée comme étant l'une des plus importantes méthodes de détection d'objet, et elle est notamment connue pour avoir introduit plusieurs notions reprises par de nombreux chercheurs en vision par ordinateur.

Dans ce quatrième chapitre on a mis en pratique cette méthode pour la détection de plaque d'immatriculation dans une vidéo tout en la fragmentant en plusieurs images chargées dans le HDFS sous plusieurs frames en utilisant la bibliothèque OpenCV avec le langage java qui implémente l'algorithme précédent . Cet algorithme est une méthode de détection de rectangles qui ont les caractéristiques d'objet à détecter dans une image numérique et en manipulant avec le Map/Reduce et dans notre cas l'objet c'est la plaque d'immatriculation. Tout en illustrant ça avec des captures d'écrans et un schéma de hadoop qui explique l'architecture de notre système utilisé dans notre application et le résultat obtenu.

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

IV.1. Architecture du système:

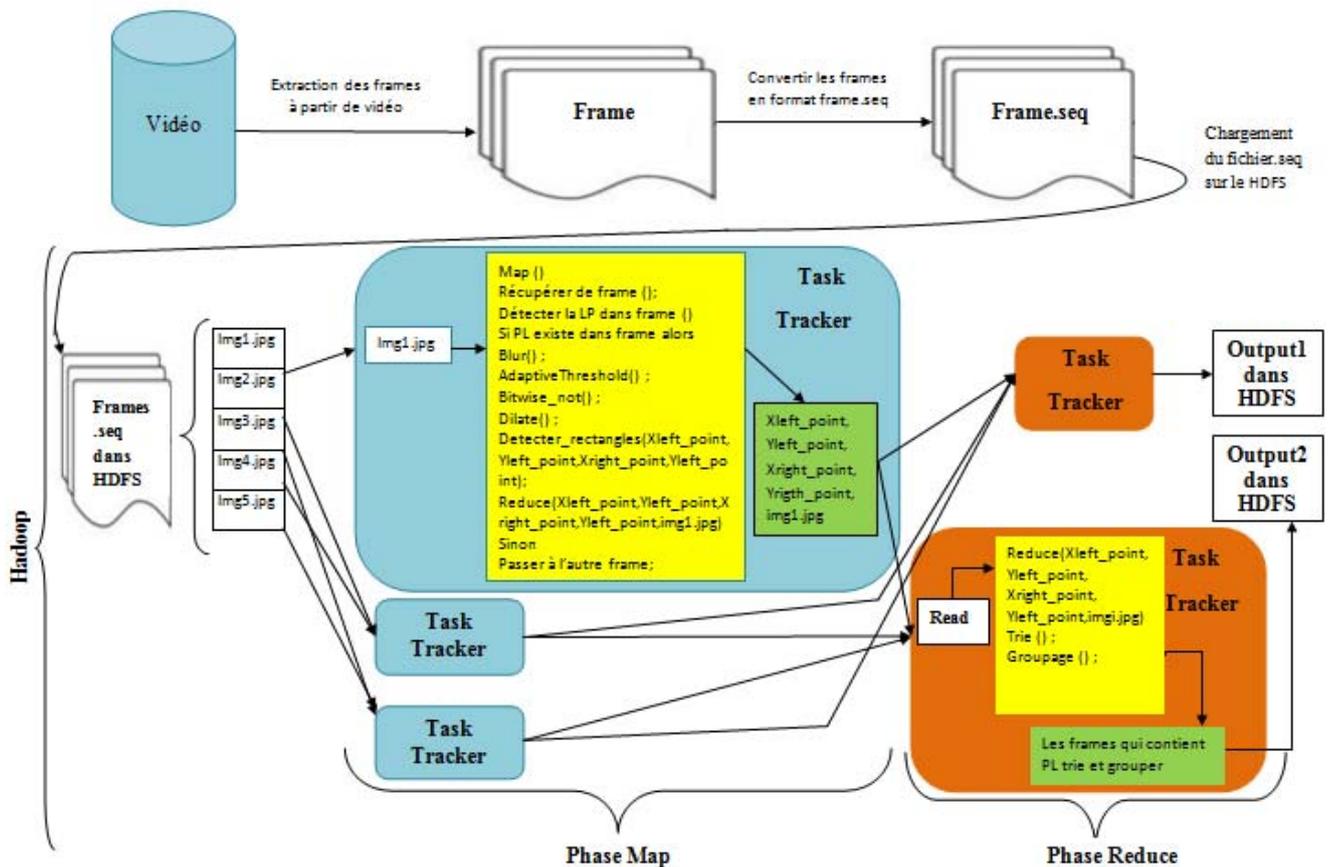


Figure IV.1 : L'architecture du système.

IV.2. Plateforme utilisée:

➤ Ubuntu : [2]

Ubuntu est un système d'exploitation open source libre et gratuit basé sur la distribution Linux Debian. Son nom « Ubuntu » provient d'un ancien mot africain qui signifie « Humanité » et également qui signifie « je suis ce que je suis grâce à ce que nous sommes tous ». Dans le même ordre d'esprit, les utilisateurs sont encouragés à étudier son fonctionnement, le modifier, l'améliorer et enfin de le redistribuer.

Ubuntu est construit sur l'architecture et l'infrastructure Debian, pour fournir des systèmes d'exploitation au serveur Linux, bureau, téléphone, tablette et d'exploitation de la télévision. Par défaut, Ubuntu est installé avec une pluralité de logiciels libres tels que LibreOffice, Firefox, ...etc. Plusieurs autres logiciels libres peuvent être installés, notamment grâce à la

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

logithèque Ubuntu (par exemple : GIMP, VLC), contrairement aux logithèques des autres systèmes d'exploitation les plus populaires, tel que Mac OS et Windows.

Ubuntu se mis à jour de façon prévisible tous les six mois, et chaque version reçoit un soutien gratuit pour les neuf mois avec des correctifs de sécurité, à fort impact des corrections de bugs à faible risque et conservateurs. La première version était en Octobre 2004.

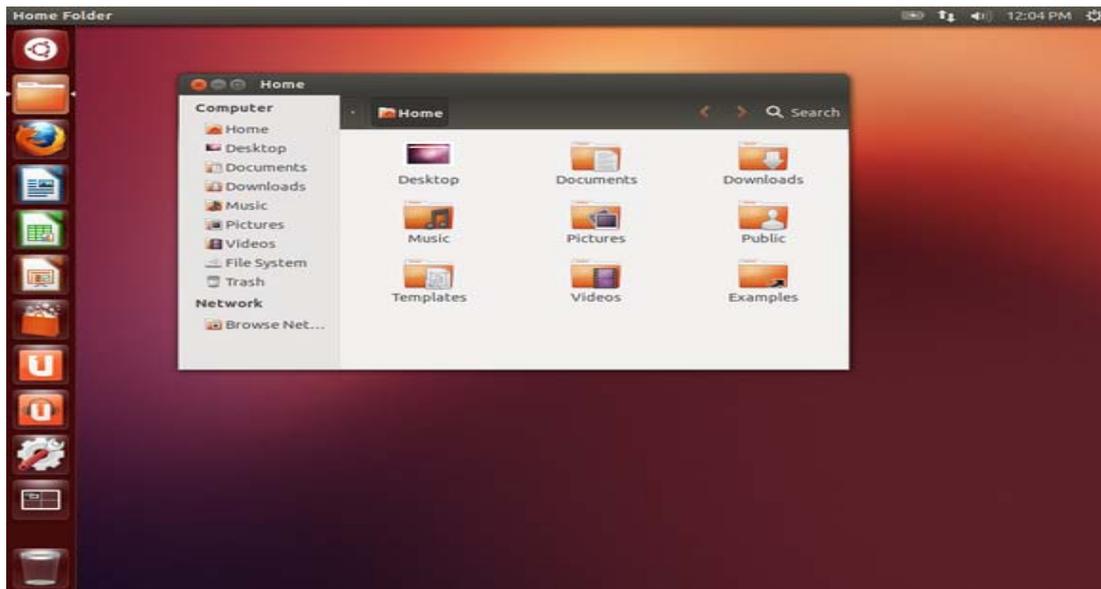


Figure IV.2 : La page principale de l'environnement Ubuntu.

➤ Virtual machine : [3]

Une machine virtuelle (VM) est une émulation d'un système informatique particulier. Les machines virtuelles fonctionnent sur les bases de l'architecture d'ordinateur et les fonctions réel ou hypothétique d'ordinateur, et leur mise en œuvre peut impliquer du matériel spécialisé, des logiciels, ou une combinaison des deux.

Une machine virtuelle (VM) est un système d'exploitation (OS) environnement ou application qui est installée sur le logiciel qui imite le matériel dédié. L'utilisateur final a la même expérience sur une machine virtuelle comme ils auraient sur du matériel dédié. Elle est une implémentation logicielle d'une machine (par exemple, un ordinateur) qui exécute des programmes comme une machine physique. Les machines virtuelles sont séparées en deux grandes classes, en fonction de leur utilisation et le degré de correspondance à toute machine réelle:

_ Une machine virtuelle du système fournit une plate-forme de système complet qui prend en charge l'exécution d'un système d'exploitation complet (OS). Ces émuler généralement une architecture existante, et sont construits avec le but soit de fournir une plate-forme pour exécuter des programmes où le matériel réel ne sont pas disponibles pour l'utilisation, ou

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

d'avoir plusieurs instances de machines virtuelles conduisant à une utilisation plus efficace des ressources informatiques, à la fois en termes de consommation d'énergie et de l'efficacité des coûts (connu comme la virtualisation matérielle, la clé à un environnement de cloud computing), ou les deux.

_Une machine virtuelle de processus (aussi, la langue machine virtuelle) est conçu pour exécuter un seul programme, ce qui signifie qu'il prend en charge un seul processus. Ces machines virtuelles sont généralement étroitement adaptées à un ou plusieurs langages de programmation et construit dans le but de fournir une portabilité des programmes et la flexibilité (entre autres). Une caractéristique essentielle d'une machine virtuelle est que le logiciel fonctionne à l'intérieur est limité aux ressources et aux abstractions fournies par la machine virtuelle, il ne peut pas sortir de son environnement virtuel.

➤ VirtualBox : [4]

VirtualBox est un produit pour l'entreprise ainsi que l'utilisation à la maison. Non seulement VirtualBox extrêmement riche en fonctionnalités, c'est un produit de haute performance pour les clients de l'entreprise, il est également la seule solution professionnelle qui est librement disponible en tant que logiciel Open Source sous les termes de la GNU General Public License (GPL).

À l'heure actuelle, VirtualBox fonctionne sur Windows, Linux, Macintosh, et les hôtes Solaris et prend en charge un grand nombre de systèmes d'exploitation invités, y compris mais sans s'y limiter à Windows, Solaris, et OpenBSD.

VirtualBox est activement développé avec les versions fréquentes et a une liste toujours croissante de fonctionnalités, pris en charge l'exécution sur les systèmes et plates-formes d'exploitation invités. VirtualBox est un effort communautaire soutenu par une société dédiée, tout le monde est invité à contribuer en Oracle assure que le produit répond aux critères toujours de qualité professionnelle.

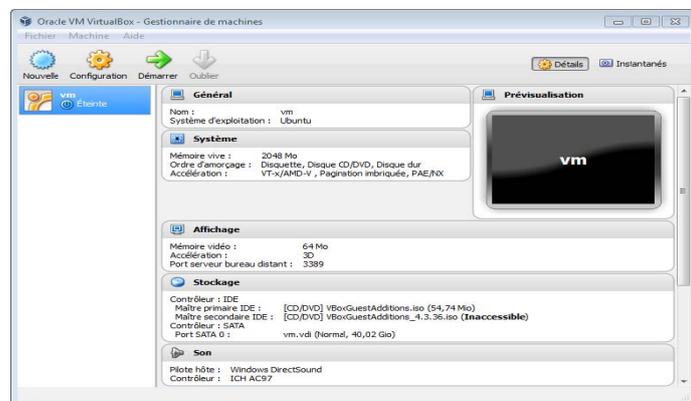
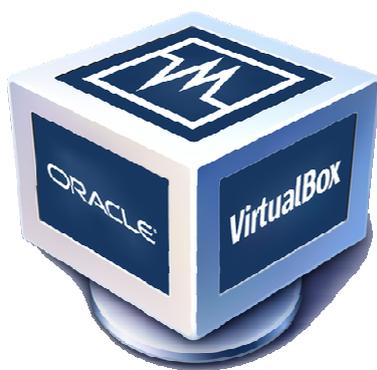


Figure IV.3 : La page principale de VirtualBox

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

➤ OpenCV :

Présentation d'OpenCV :

OpenCV est une librairie open source de traitement et analyse d'images et vidéos avec des interfaces pour les principaux langages de programmation C, C++, Java, C#, Python ... etc. Il est optimisée pour les applications temps réelles, c'est un supercalculateur qui permet de traiter des images en 1600x1200 et qui donnera un super résultat, mais au prix d'une surcharge de calcul importante. [5]

OpenCV est un framework très puissant pour la vision artificielle, il est utilisé aussi bien dans les laboratoires de recherche que dans l'industrie. Il fournit une API bas et haut niveau et une interface pour le langage de programmation parallèle IPP. [6]

La bibliothèque OpenCV nous offre une démonstration très intéressante pour une détection d'objet. En outre, elle nous offre des programmes (ou fonctions) qu'ils ont utilisé pour former des classificateurs pour leur système de détection d'objet, de sorte que nous pouvons créer nos propres classificateurs d'objets à l'aide de ces fonctions. [6]

Principaux fonctions d'OpenCV: [5]

- Manipulation et acquisition de vidéos
- Manipulation d'images (chargement, sauvegarde, copie, conversion...)
- Manipulations de matrices et algèbre linéaire
- Structure de données utilitaires variées (listes, files, ensembles, graphes...)
- Traitement d'images (filtrage, détections de discontinuités, morphologie mathématique...)
- Analyse d'images (composantes connexes, ajustement de primitives, transformée de distance...)
- Vision (calibration de caméra, stéréovision, recherche d'association...)
- Reconnaissance de forme (modèles de markov, ACP...)
- Extraction de caractéristiques
- Interface graphique (affichage d'images, de vidéos, gestion des évènements...)

➤ JAVA : [2]

Java est un langage de programmation objet, dont la syntaxe ressemble au langage C++. Il a été mis au point à partir de 1990 par la firme Sun Microsystems et officiellement présenté en 23 mai 1995. Le but de Java à l'époque était de constituer un langage de programmation pouvant être intégré dans les appareils électroménagers, afin de pouvoir les contrôler, de les rendre interactifs, et surtout de permettre une communication entre les appareils. Ce

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

programme de développement se situait dans un projet appelé Green, visant à créer une télécommande universelle (Star 7, présentée en 1992) comprenant un système d'exploitation capable de gérer l'ensemble des appareils électroménagers de la maison.

Etant donné que le langage C++ comportait trop de difficultés, James Gosling, un des acteurs du projet (considéré désormais comme le père de Java) décida de créer un langage orienté objet reprenant les caractéristiques principales du C++, en éliminant ses points difficiles, et en le rendant moins encombrant et plus portable (il devait pouvoir être intégré dans n'importe quel appareil...).

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications. Pour cela, divers plateformes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java.

Java a donné naissance à un système d'exploitation (Java OS), à des environnements de développement (eclipse/JDK), des machines virtuelles (MSJVM (en), JRE) applicatives multiplateforme (JVM), une déclinaison pour les périphériques mobiles/embarqués (J2ME), une bibliothèque de conception d'interface graphique (AWT/Swing), des applications lourdes (Jude, Oracle SQL Worksheet, etc.), des technologies web (servlets, applets) et une déclinaison pour l'entreprise (J2EE).

IV.3. L'algorithme de détection des rectangles des plaques d'immatriculation sans MapReduce:

Pour détecter la plaque d'immatriculation une méthode a été choisie qui est l'algorithme de détection de rectangle de PL tout en commençant par la présentation de PL à détecter et ce dernier se base sur le traitement d'images en utilisant la bibliothèque OpenCV, et chaque étape du traitement d'images a été expliquée ci-dessus et présentée dans l'algorithme exécuté et présenté sous forme de capture d'écran .

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

❖ Le processus de traitement d'image est comme suit :



Figure IV.4: Une image qui contient une plaque d'immatriculation de véhicule sans aucune modification sur laquelle on applique l'algorithme de détection des rectangles des plaques d'immatriculation.

➤ Application d'un lissage (Blur) :

Est aussi dit le filtre de retouche photo pour réduire le bruit dans l'image de la plaque d'immatriculation

Un système d'enregistrement d'image ne restitue pas l'image de manière parfaite : des informations parasites apparaissent et viennent s'ajouter de manière aléatoire aux détails de la scène d'origine. Cet effet est désigné sous le terme général « bruit », pour un appareil photo numérique, le bruit est principalement dû à la physique des capteurs et aux composants électroniques qui exploitent le signal provenant des capteurs. Le lissage a pour but de réduire l'influence du bruit, d'éliminer les éléments non significatifs dans une image numérique, afin d'obtenir une meilleure restitution de l'image pour son analyse [10] et aussi pour améliorer la visualisation pour les simplifier en but d'un traitement postérieur telque "Blur" le lissage des bords irréguliers de l'image.

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce



Figure IV.5: Filtrage d'une image qui contient une plaque d'immatriculation de véhicule avec l'opérateur Blur.

➤ La segmentation de l'image :

La segmentation d'image est très utile pour définir les objets et leurs limites [13] et elle se base sur plusieurs techniques telque :

" Adaptive Threshold" qui est une des plus larges et simple méthodes utilisée pour la segmentation des images à partir des images en niveau de gris qui peuvent être converties en images binaire. [14] Cette méthode a pour but de contrôler le pourcentage du noir au blanc de l'image et régler le seuil inférieur pour les zones plus blanches, ou supérieur, y compris les zones plus noires de l'image.

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce



Figure IV.6: Seuillage pour une image contient une plaque d'immatriculation de véhicule avec la méthode Adaptive Threshold.

➤ Transformation binaire de l'image :

L'image binaire doit contenir toutes les informations essentielles sur la position et la forme des objets d'intérêt (premier plan). L'avantage d'une image binaire est de réduire la complexité des données et simplifie le processus de reconnaissance et de classification. La façon la plus courante de conversion d'une image du niveau de gris vers une image binaire est de sélectionner la valeur de seuil [16] global calculé pour chaque bloc de l'image découpée en blocs [17], puis d'utiliser une des opérations de la base Bitwise : Bitwise_Not (Inverse la valeur de chaque bits), Bitwise_AND, Bitwise_OR, Bitwise_XOR qu'on applique lors de l'extraction d'une partie de l'image. [18]

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce



Figure IV.7: Transformation binaire d'une image qui contient une plaque d'immatriculation de véhicule avec l'opérateur Bitwise_Not.

➤ Application de la dilatation de l'image:

Dilatation est un opérateur très connue la morphologie des filtres non-linéaires pour des transformations simples basées sur la forme de l'image [20]. L'effet de la dilatation est d'élargir la figure, la hauteur et la largeur de la figure dilatée seront les sommes respectivement des hauteurs et des largeurs de la figure originelle et de l'élément structurant. [21]

On utilise les fonctions des bibliothèques OpenCV pour dilater légèrement les bords détectés. Cette fonction est utile car une fois que les caractères de plaque d'immatriculation sont dilatés, ils vont remplir par la suite le rectangle où ils se trouvent. [12]

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce



Figure IV.8: Application d'une dilatation sur une image qui contient une plaque d'immatriculation de véhicule.

➤ **Détection des rectangles :**

Est la détection des formes rectangulaires trouvées désignant la plaque d'immatriculation sur la base affichant (largeur * hauteur).

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce



Figure IV.9: Détection du rectangle de la plaque d'immatriculation de véhicule.

❖ L'algorithme avec OpenCV :

```
package test3opencv;  
  
import org.opencv.core.Core;  
import org.opencv.core.Mat;  
import org.opencv.core.MatOfByte;
```

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

```
import org.opencv.core.Scalar;
import org.opencv.core.Size;
import org.opencv.imgcodecs.*;
import org.opencv.imgproc.*;
import org.opencv.core.MatOfRect;
import org.opencv.core.Point;
import org.opencv.core.Rect;
import org.opencv.objdetect.CascadeClassifier;
import org.opencv.core.MatOfPoint;
import org.opencv.core.MatOfPoint2f;

import java.awt.image.BufferedImage;
import java.io.ByteArrayInputStream;
import java.io.File;
import java.io.InputStream;
import java.util.ArrayList;
import java.util.List;

import javax.imageio.ImageIO;
import javax.swing.ImageIcon;
import javax.swing.JFrame;
import javax.swing.JLabel;

public class DetectImmatriculation {
    public void run(String imageFile) {
        System.out.println("\nRunning Detection de PL");
        // Créer un détecteur de PL à partir du fichier en cascade dans les ressources
        System.loadLibrary(Core.NATIVE_LIBRARY_NAME);

        Mat image = Imgcodecs.imread(imageFile,Imgcodecs.CV_LOAD_IMAGE_GRAYSCALE);
        Mat image2 = Imgcodecs.imread(imageFile,Imgcodecs.CV_LOAD_IMAGE_COLOR);
        // Le filte de lissage (blur)
        Imgproc.GaussianBlur(image, image, new Size(11, 11), 0);
        saveImage (imageFile+"_blur",image);
        // Segmentation de l'image (adaptiveThreshold)
        Imgproc.adaptiveThreshold(image, image, 255, Imgproc.ADAPTIVE_THRESH_MEAN_C, Imgproc.THRESH_BINARY, 5, 2);
        saveImage (imageFile+"_adaptiveThreshold",image);
        // Transformation binaire de l'image (bitwise_not)
        Core.bitwise_not(image, image);
        saveImage(imageFile+"_bitwise_not",image);
        // Dilatation de l'image (dilate)
        Mat kernel = Imgproc.getStructuringElement(Imgproc.MORPH_DILATE, new Size(3, 3), new Point(1, 1));
        Imgproc.dilate(image, image, kernel);
        saveImage (imageFile+"_dilate",image);
        List<MatOfPoint> contours = new ArrayList<MatOfPoint>();
        Mat hierarchy = new Mat();
        Imgproc.findContours(image, contours, hierarchy, Imgproc.RETR_TREE, Imgproc.CHAIN_APPROX_SIMPLE);
        MatOfPoint2f approxCurve = new MatOfPoint2f();
        for (int i = 0; i < contours.size(); i++) {
            // Convertir les contours de MatOfPoint à MatOfPoint2f
            MatOfPoint2f contour2f = new MatOfPoint2f(contours.get(i).toArray());
            // Le traitement sur mMOP2f1 qui est dans le type MatOfPoint2f
            double approxDistance = Imgproc.arcLength(contour2f, true) * 0.02;
            if (approxDistance > 5) {
                // Trouver les Polygones
                Imgproc.approxPolyDP(contour2f, approxCurve, approxDistance, true);
                // Convertir le retour à MatOfPoint
                MatOfPoint points = new MatOfPoint(approxCurve.toArray());
                // Vérifie Rectangle - Points, région, convexité
            }
        }
    }
}
```


Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

IV.4. Algorithme de détection en utilisant MapReduce pour la détection de PL :

a) Ouverture d'une vidéo et récupération des images :

- ✓ cd /usr/local/ (se n'est pas obligé c'est juste pour dire que on sort de tous nœuds et qu'on se place dans le nœud local à partir de racine avec le chemin cd->usr->local)
- ✓ start-all.sh (lancement du Hadoop)
- ✓ jps
- ✓ ffmpeg -i /home/m1/Bureau/home/m1/Bureau/vidéo (chemin de la vidéo) -r 30/1 /home/m1/Bureau/frames (chemin de destination) \$ frame% 0004d.jpg (la fragmentation de la vidéo en une séquence d'images)
- ✓ compresser le fichier des photos générer son aucune commande,
 - aller au l'emplacement ou il y a le dossier qui contient le fichier frames générer;
 - Sélectionner le fichier frames ;
 - Clic droit est choisir compress ;
- ✓ cd /usr/local/tar-to-seq
- ✓ java -jar tar-to-seq.jar /home/m1/Bureau/vidéo.tar/home/m1/Bureau/vidéo.seq
- ✓ ls /home/m1/Bureau/
- ✓ hdfs dfsadmin -safemode leave (enlever le safe mode)
- ✓ hadoop fs -ls (pour lister tous les fichiers Hadoop (HDFS))
- ✓ hadoop fs -put /home/m1/Bureau/tar-to-seq/nomdelavidéo.seq /tracker/ (charger le fichier dans Hadoop)
- ✓ hadoop fs -ls /tracker/ (vérifier que le fichier.seq est dans Hadoop)
- ✓ bash run -nomduprogramme.sh (exécution du programme MapReduce)

a) L'algorithme:

```
import java.awt.image.BufferedImage;
import java.io.ByteArrayInputStream;
import java.io.IOException;
import java.awt.FlowLayout;
import java.awt.Image;
import java.awt.image.DataBufferByte;
import java.util.ArrayList;
import java.util.List;

import javax.imageio.ImageIO;

import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.ByteWritable;
import org.apache.hadoop.io.BytesWritable;
import org.apache.hadoop.io.DoubleWritable;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.lib.input.SequenceFileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
```

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

```
import org.opencv.core.Core;
import org.opencv.core.CvType;
import org.opencv.core.KeyPoint;
import org.opencv.core.Mat;
import org.opencv.core.MatOfKeyPoint;
import org.opencv.core.MatOfPoint;
import org.opencv.core.Point;
import org.opencv.features2d.DescriptorExtractor;
import org.opencv.features2d.DescriptorMatcher;
import org.opencv.features2d.FeatureDetector;
import org.opencv.imgcodecs.Imgcodecs;
import org.opencv.imgproc.Imgproc;

import org.opencv.core.MatOfByte;
import org.opencv.core.Scalar;
import org.opencv.core.Size;
import org.opencv.imgcodecs.*;
import org.opencv.imgproc.*;
import org.opencv.core.MatOfRect;
import org.opencv.core.Rect;
import org.opencv.core.MatOfPoint2f;

import java.io.File;
import java.io.InputStream;

public class LPDetection extends Configured implements Tool {

    public static class LPDetectionMapper extends Mapper<Text, BytesWritable, Text, Text> {

        public void map(Text key, BytesWritable value, Context context)
            throws IOException, InterruptedException {

            System.loadLibrary(Core.NATIVE_LIBRARY_NAME);
            BufferedImage targImage = ImageIO.read(new ByteArrayInputStream(value.getBytes()));

            byte[] pixels = ((DataBufferByte) targImage.getRaster().getDataBuffer())
                .getData();

            // Create a Matrix the same size of image
            Mat image = new Mat(targImage.getHeight(), targImage.getWidth(), Imgcodecs.CV_LOAD_IMAGE_GRAYSCALE);
            // Remplir la matrice avec des valeurs d'image
            image.put(0, 0, pixels);

            // Verifier l'image detecter, est d'une taille suffisante et comporte trois canaux de couleur (RVB)

            if (targImage != null && targImage.getWidth() > 1 && targImage.getHeight() > 1) {
                //int intkey= Integer.parseInt((key.toString().substring(10, 13)));

                Imgproc.GaussianBlur(image, image, new Size(11, 11), 0);
                Imgproc.adaptiveThreshold(image, image, 255, Imgproc.ADAPTIVE_THRESH_MEAN_C, Imgproc.THRESH_BINARY, 5, 2);
                Core.bitwise_not(image, image);
                Mat kernel = Imgproc.getStructuringElement(Imgproc.MORPH_DILATE, new Size(3, 3), new Point(1, 1));
                Imgproc.dilate(image, image, kernel);
            }
        }
    }
}
```

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

```
List<MatOfPoint> contours = new ArrayList<MatOfPoint>();
Mat hierarchy = new Mat();
Imgproc.findContours(image, contours, hierarchy, Imgproc.RETR_TREE, Imgproc.CHAIN_APPROX_SIMPLE);
MatOfPoint2f approxCurve = new MatOfPoint2f();

for (int i = 0; i < contours.size(); i++) {

    // Convertir les contours de MatOfPoint à MatOfPoint2f
    MatOfPoint2f contour2f = new MatOfPoint2f(contours.get(i).toArray());
    // Traitement sur mMOP2f1 qui est dans le type MatOfPoint2f
    double approxDistance = Imgproc.arcLength(contour2f, true) * 0.02;

    if (approxDistance > 5) {

        // Trouver les Polygones
        Imgproc.approxPolyDP(contour2f, approxCurve, approxDistance, true);

        /* Autre
           retour à MatOfPoint
        */
        MatOfPoint points = new MatOfPoint(approxCurve.toArray());

        // Vérifie Rectangle - Points, région, convexité
        if (points.total() == 4 && Math.abs(Imgproc.contourArea(points)) > 1000 && Imgproc.isContourConvex(points)) {
            double cos = 0;
            double mcos = 0;

            for (int sc = 2; sc < 5; sc++) {

                // Un moyen de faire vérifier l'angle de la figure
                if (cos > mcos) {
                    mcos = cos;
                }
            }

            if (mcos < 0.3) {

                // Délimitant contour du rect obtenez
                Rect rect = Imgproc.boundingRect(points);

                if (Math.abs(rect.height - rect.width) > 200) {
                    // Dessiner enfermant rectangle
                    context.write(new Text(key.toString()), new Text(rect.tl().x+", "+rect.tl().y+", "+rect.br().x
+", "+rect.br().y+"));
                }
            }
        }
    }
} //map()
}
```

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

```
|
public static class LPDetectionReducer extends Reducer<Text, Text, Text, Text> {

    public void reduce(Text key, Iterable<Text> values, Context context)
        throws IOException, InterruptedException {
        String ret="";
        for (Text val : values) {
            ret+=val;
        }
        context.write(new Text(key), new Text( ret));
    } // reduce()

} // LPDetectionReducer

public int run(String[] args) throws Exception {
    // Vérifiez les arguments d'entrée
    if (args.length != 2) {
        System.out.println("Usage: LPDetection <input HIB> <output directory>");
        System.exit(0);
    }

    // Initialiser et configurer l'emploi MapReduce
    Job job = Job.getInstance();
    // Définir la classe de format d'entrée qui analyse l'entrée HIB et génère des tâches de carte
    job.setInputFormatClass(SequenceFileInputFormat.class);
    // Définir les classes pilote, mapper et réducteur qui expriment le calcul
    job.setJarByClass(LPDetection.class);
    job.setMapperClass(LPDetectionMapper.class);

    // Initialiser et configurer l'emploi MapReduce
    Job job = Job.getInstance();
    // Définir la classe de format d'entrée qui analyse l'entrée HIB et génère des tâches de carte
    job.setInputFormatClass(SequenceFileInputFormat.class);
    // Définir les classes pilote, mapper et réducteur qui expriment le calcul
    job.setJarByClass(LPDetection.class);
    job.setMapperClass(LPDetectionMapper.class);

    job.setReducerClass(LPDetectionReducer.class);
    // Définissez les types pour les paires clé / valeur transmis à / de la carte et de réduire les couches
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(Text.class);
    // Définissez les chemins d'entrée et de sortie sur le HDFS
    FileInputFormat.setInputPaths(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    // Exécuter le travail de MapReduce et le bloc jusqu'à ce qu'il complets
    boolean success = job.waitForCompletion(true);
    // Retour succès ou l'échec
    return success ? 0 : 1;
}

public static void main(String[] args) throws Exception {
    ToolRunner.run(new LPDetection(), args);
    System.exit(0);
}
}
```

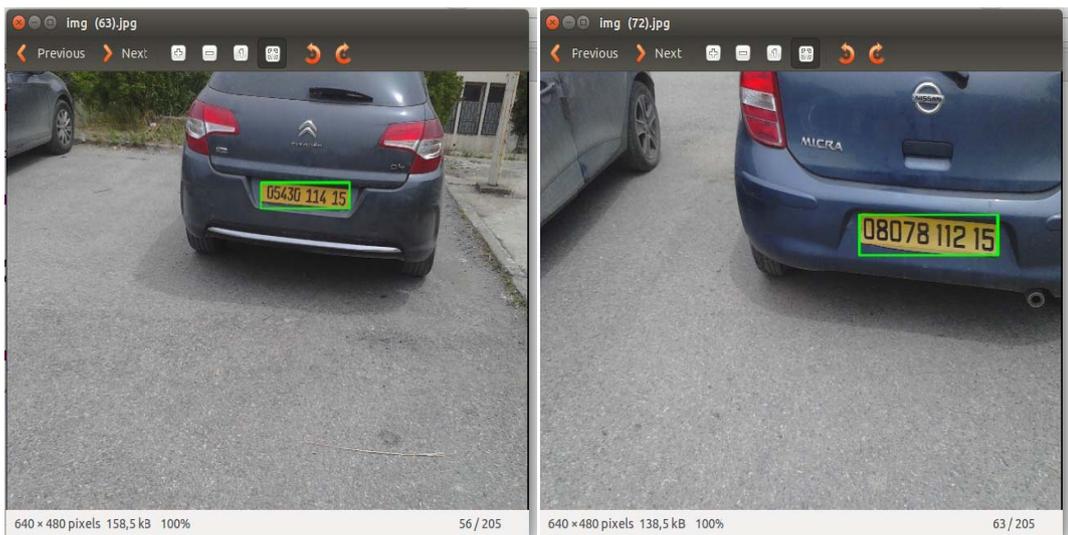
Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

IV.5. Exemples de résultat attendu :

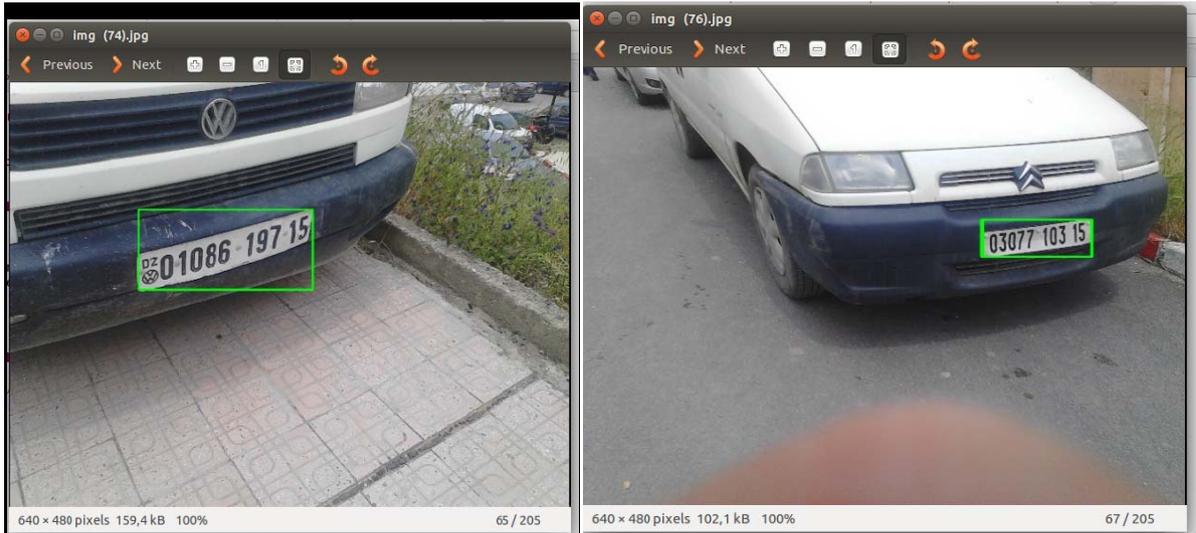
Exemple étape par étape:



Exemples directe:



Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce



IV.6. Les résultats en chiffres :

```
Input split bytes=107
Combine input records=0
Combine output records=0
Reduce input groups=0
Reduce shuffle bytes=6
Reduce input records=0
Reduce output records=0
Spilled Records=0
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=988
CPU time spent (ms)=6760
Physical memory (bytes) snapshot=340643840
Virtual memory (bytes) snapshot=3542728704
Total committed heap usage (bytes)=289591296

Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0

File Input Format Counters
  Bytes Read=4959845
File Output Format Counters
  Bytes Written=0

m1@m1-VirtualBox:/usr/local/hipi$ hadoop fs -ls /tracker/output/
16/07/12 03:26:50 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
cable
Found 2 items
-rw-r--r--  1 m1 supergroup          0 2016-07-12 03:25 /tracker/output/_SUCCESS
-rw-r--r--  1 m1 supergroup          0 2016-07-12 03:25 /tracker/output/part-r-00000
m1@m1-VirtualBox:/usr/local/hipi$
```

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

```
m1@m1-VirtualBox: /usr/local/hipi
Total time spent by all reduces in occupied slots (ms)=360865
Total time spent by all map tasks (ms)=1173839
Total time spent by all reduce tasks (ms)=360865
Total vcore-seconds taken by all map tasks=1173839
Total vcore-seconds taken by all reduce tasks=360865
Total megabyte-seconds taken by all map tasks=120201136
Total megabyte-seconds taken by all reduce tasks=369525760
Map-Reduce Framework
Map input records=84
Map output records=0
Map output bytes=0
Map output materialized bytes=6
Input split bytes=107
Combine input records=0
Spilled Records=0
Failed Shuffles=0
Merged Map outputs=0
GC time elapsed (ms)=10120
CPU time spent (ms)=50570
Physical memory (bytes) snapshot=336236544
Virtual memory (bytes) snapshot=1849331712
Total committed heap usage (bytes)=297222144
File Input Format Counters
Bytes Read=85564628
m1@m1-VirtualBox: /usr/local/hipi$
```

```
16/07/12 01:13:04 INFO mapreduce.Job: map 50% reduce 17%
16/07/12 01:13:37 INFO mapreduce.Job: map 51% reduce 17%
16/07/12 01:13:48 INFO mapreduce.Job: map 52% reduce 17%
16/07/12 01:13:58 INFO mapreduce.Job: map 53% reduce 17%
16/07/12 01:14:05 INFO mapreduce.Job: map 54% reduce 17%
16/07/12 01:14:12 INFO mapreduce.Job: map 55% reduce 17%
16/07/12 01:14:19 INFO mapreduce.Job: map 56% reduce 17%
16/07/12 01:14:26 INFO mapreduce.Job: map 57% reduce 17%
16/07/12 01:14:32 INFO mapreduce.Job: map 58% reduce 17%
16/07/12 01:14:40 INFO mapreduce.Job: map 59% reduce 17%
16/07/12 01:14:49 INFO mapreduce.Job: map 60% reduce 17%
16/07/12 01:14:53 INFO mapreduce.Job: map 61% reduce 17%
16/07/12 01:15:03 INFO mapreduce.Job: map 62% reduce 17%
16/07/12 01:15:10 INFO mapreduce.Job: map 63% reduce 17%
16/07/12 01:15:20 INFO mapreduce.Job: map 64% reduce 17%
16/07/12 01:15:23 INFO mapreduce.Job: map 65% reduce 17%
16/07/12 01:15:34 INFO mapreduce.Job: map 66% reduce 17%
16/07/12 01:15:37 INFO mapreduce.Job: map 67% reduce 17%
16/07/12 01:15:47 INFO mapreduce.Job: map 68% reduce 17%
16/07/12 01:15:50 INFO mapreduce.Job: map 100% reduce 100%
16/07/12 01:15:55 INFO mapreduce.Job: Job job_1468267150984_0002 failed with state FAILED due to: Task failed task_1468267150984_0002_m_000000
Job failed as tasks failed, failedMaps:1 failedReduces:0
16/07/12 01:15:55 INFO mapreduce.Job: Counters: 40
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=115228
FILE: Number of read operations=0
```

```
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=85564735
HDFS: Number of bytes written=0
HDFS: Number of read operations=4
HDFS: Number of large read operations=0
HDFS: Number of write operations=0
Job Counters
Failed map tasks=4
Killed map tasks=1
Killed reduce tasks=1
Launched map tasks=6
Launched reduce tasks=1
Other local map tasks=4
Data-local map tasks=2
Total time spent by all maps in occupied slots (ms)=1173839
Total time spent by all reduces in occupied slots (ms)=360865
Total time spent by all map tasks (ms)=1173839
Total time spent by all reduce tasks (ms)=360865
Total vcore-seconds taken by all map tasks=1173839
Total vcore-seconds taken by all reduce tasks=360865
Total megabyte-seconds taken by all map tasks=120201136
Total megabyte-seconds taken by all reduce tasks=369525760
Map-Reduce Framework
Map input records=84
Map output records=0
Map output bytes=0
Map output materialized bytes=6
Input split bytes=107
Combine input records=0
Spilled Records=0
Failed Shuffles=0
```

Chapitre IV : Détection de plaque d'immatriculation de véhicule basée sur MapReduce

Conclusion :

Des recherches menées ce système peut détecter les plaques d'immatriculation dans une vidéo. La chose la plus importante dans l'utilisation de la méthode Viola-Jones est la qualité des données d'apprentissage. La bonne qualité des données est d'avoir une variété de combinaisons de caractères, a un axe x de la pente, l'axe y et l'axe z de façon uniforme et pas trop extrême, mais il est tout aussi important l'éclairage de l'objet à détecter.

La conclusion qui peut être tirée est que plus la qualité des données d'apprentissage varié sera directement proportionnelle au niveau de la précision dans la détection du système de plaque d'immatriculation en utilisant la méthode Viola-Jones. Il y a tellement de choses qui peuvent être améliorées de cette recherche afin que la recherche future puisse être mieux faite, entre autres choses telles que la façon de détecter la meilleure image de la plaque d'immatriculation dans un ensemble de frame dans la vidéo.

On travaillant avec une grande quantité de donnée ce qu'on appelle les Big data le hadoop est le meilleur framework qui répond à la multiplication de donnée et il offre une simplicité pour l'exécution de tâche parallèle et pour garantir au futur une sclabilité avec notre application le Map/Reduce peut garantir ça, et moins chère à déployer et facile à réaliser avec le framework hadoop.

Conclusion générale

Le processus de la détection de plaque d'immatriculation à partir d'une séquence vidéo nécessite une étude profonde et des recherches intenses sur les Big data et le traitement d'image et plus précisément sur les framework qui manipulent et implémentent les volumes massive de données, ainsi que des méthodes qui permettent la détection des caractéristiques locales et globales afin de détecter la plaque d'immatriculation du véhicule à partir d'une image. Pour cela on a expliqué au début de ce mémoire les éléments de base sur les Big data, la solution software conçues pour simplifier et accélérer l'obtention d'informations synthétiques à partir de l'analyse métier, et la gestion des grands volumes de données à un champ d'application très vaste et varié ainsi que la plateforme de manipulation de ce grand volume de données qui est nommée Hadoop.

Cette plateforme est un Framework qui permet le traitement distribué de grands ensembles de données à travers des grappes d'ordinateurs utilisant des modèles simples de programmation. Elle est conçue pour évoluer à partir de serveurs uniques à des milliers de machines.

A fin de bien munir ces tâches, Hadoop se base sur un modèle de programmation massivement parallèle adapté au traitement de très grandes quantités de données dite MapReduce. Enfin, le Framework Java MapReduce fournit une API pour écrire des applications qui vont pouvoir traiter de larges quantités d'information telle que les séquences vidéo de surveillances capturées aux autoroutes, dans des parkings...etc.

L'objectif visé par le deuxième chapitre est l'étude des concepts de bases du domaine de traitement d'images qu'on présente dans ces trois majeurs points :

- Etude détaillée sur les vidéos : Définition, Composants et caractéristiques.
- Les systèmes de recherche d'information basés sur le contenu vidéo (CBVIR) et les méthodes d'indexation et de recherche.
- Définition détaillée sur les images et la détermination des caractéristiques locales et globales ainsi les détecteurs et les descripteurs de caractéristiques d'une image.

Le domaine d'application de la détection des plaques d'immatriculation de véhicule est très vaste, pour cela on a évoqué un certain nombre de notions et d'éléments concernant la phase de la détection et la phase de la reconnaissance des plaques d'immatriculation, les méthodes de la détection des plaques d'immatriculation de véhicule aux différents niveaux de traitement de l'image et les méthodes utilisées pour la détection des plaques d'immatriculation à grande échelle afin de se rapprocher plus vers notre objectif.

Les concepts du domaine de détection d'immatriculation simplifient l'implémentation de la détection de plaque d'immatriculation de véhicule. Notre méthode pour cela est basée sur la technique du framework MapReduce pour écrire un programme java qu'on exécute sous le framework Hadoop. Ce programme doit être à la hauteur de détecter l'existence ou non d'une plaque d'immatriculation à partir d'une séquence d'images qui seront traitées et générées dans un algorithme de détection de rectangles de PLs avec OpenCV.

Conclusion générale

Comme perspective pour élargir notre domaine d'application et publié ce genre d'application aux grandes entreprises ou a des secteurs concernés c'est possible d'utiliser la technologie d'aujourd'hui qui est les cloudprivéstout en disposant des web services pour faciliter le stockage des big data et le traitement et l'exécution faciles des programmes.

Bibliographie / Webographie

Chapitre I

- [1] : [piloter.org/Big data quoi pourquoi comment](http://piloter.org/Big-data-quoi-pourquoi-comment)
- [2] : [marketing.fr/Thematique/Strategies-1001/Dossiers/Le Big Data 255/Definition de Big Data, trois V : variete, volume, velocite](http://marketing.fr/Thematique/Strategies-1001/Dossiers/Le-Big-Data-255/Definition-de-Big-Data,-trois-V:-variete,-volume,-velocite)
- [3] : [piloter.org/business intelligence/big data definition](http://piloter.org/business-intelligence/big-data-definition)
- [4] : <https://www-01.ibm.com/software/fr/data/bigdata/>
- [5] : WordPress.com | Thème Big Data : l'analyse de données en masse
- [6] : [futura-sciences.com/d'informatique Cloud computing](http://futura-sciences.com/d'informatique/Cloud-computing)
- [7] : http://www.t-systems.fr/home/page-d-accueil/492094_1/blobBinary/White-Paper_Cloud+Computing-ps.pdf
- [8] : http://www.medinsoft.com/website/custom/module/cms/content/file/Ce_qu_il_faut_savoir_sur_le_Cloud....pdf
- [9] : salesforce.com/fr/cloudcomputing
- [10] : [lemagit.fr/definition/NoSQL base de donnees Not Only SQL](http://lemagit.fr/definition/NoSQL-base-de-donnees-Not-Only-SQL)
- [12] : [piloter.org/business intelligence/base nosql](http://piloter.org/business-intelligence/base-nosql)
- [13] : [open-source-guide.com/Solutions/Developpement et couches intermediaires/Big data/Hadoop](http://open-source-guide.com/Solutions/Developpement-et-couches-intermediaires/Big-data/Hadoop)
- [14] : wiki.apache.org/Hadoop
- [15] : journaldunet.com/solutions/analytics/hadoop
- [16] : [journaldunet.com/developpeur/outils/les solutions du Big data/les composants Hadoop au crible](http://journaldunet.com/developpeur/outils/les-solutions-du-Big-data/les-composants-Hadoop-au-crible)
- [17] : fr.slideshare.net/MPDAM/hadoop
- [18] : [blog.zenika.com/Hadoop et le MapReduce au service des gros volumes de donnees](http://blog.zenika.com/Hadoop-et-le-MapReduce-au-service-des-gros-volumes-de-donnees)
- [19] : [piloter.org/business intelligence/map reduce](http://piloter.org/business-intelligence/map-reduce)
- [20] : wikipedia.org/wiki/MapReduce

Bibliographie / Webographie

- [21] : Implémentation parallèle de certains algorithmes de fouille de données avec le framework MapReduce
Algorithmes : K-means et Apriori
Maria Malek
LARIS-EISTI
maria.malek@eisti.fr
- [22] : <http://searchcloudcomputing.techtarget.com/definition/MapReduce>
- [23] : MapReduce Algorithme de parallélisations des traitements
Khaled TANNIR
Doctorant CIFRE LARIS/ESTI
<http://blog.khaledtannir.net>
these@khaledtannir.net
2^{ème} SÉRI 2010-2011 Jeudi 17 mars 2011
- [24]:<http://mon.univ-montp2.fr/claroline/backends/download.php?url=L01hcFJIZHVjZS5wZGY%3D&cidReset=true&cidReq=BDNG2014>
- [24] : [lemagit.fr/definition/In Memory Base de donnees en memoire](http://lemagit.fr/definition/In-Memory-Base-de-donnees-en-memoire)
- [25]: [piloter.org/business intelligence/in memory](http://piloter.org/business-intelligence/in-memory)
- [26]:<http://blog.ippon.fr/2014/03/14/mapreduce-et-les-grilles-de-donnees-ou-hadoop-sans-hadoop/>
- [27]: Combining Local and Global Image Features for Object Class Recognition
Dimitri A. Işın, Marwan A. Mattar, Matthew B. Blaschke, Mark C. Benfield, Erik G. Learned-Miller

Chapitre II

- [1] : Thèse pour obtenir le grade de
DOCTEUR DE L'UNIVERSITÉ JOSEPH FOURIER- Grenoble I
Discipline : Informatique
Mbarek CHARHAD
TITRE :
Modèles de Documents Vidéo basés sur le Formalisme des Graphes Conceptuels
pour l'Indexation et la Recherche par le Contenu Sémantique
Date de soutenance : 28 novembre 2005
- [3] : Cours Indexation Multidimensionnelle : KpyrRec, une amélioration de Kpyr
Exposer à : *LIFL – UMR CNRS 8022, Laboratoire d'informatique fondamentale de Lille*
Réalise par :
Thierry Urruty
Fatima Belkouch
Chabane Djeraba

Bibliographie / Webographie

- [4] : Livre : cour de vidéo
Matériels, Tournage et prise de vues, Post-production
René Bouillot, Gérard Galès
- [5] : TRAITEMENT *DES IMAGES* et *VISION PAR MACHINE*, Jean-Marc Vézien
Jean-Marc.Vezien@limsi.fr
- [6] : THÈSE PRÉSENTÉE À L'UNIVERSITÉ D'ORLÉANS POUR OBTENIR LE GRADE
DE : DOCTEUR DE L'UNIVERSITÉ D'ORLÉANS
Discipline : Sciences et Technologies Industrielles
PAR : Arnaud CAPRI
Titre : Caractérisation des objets dans une image en vue d'une aide à l'interprétation et
d'une compression adaptée au contenu : application aux images échographiques.
Soutenue le : 2007.
- [7] : www.lossendiere.com/.../IMAGENUM1/.../grain-caracteristique-image-numerique.
- [8] : UNIVERSITÉ DU QUÉBEC À MONTRÉAL
PLASTICITÉ, ICONICITÉ, PARATEXTE: UNE ANALYSE DES RELATIONS
TEXTE-IMAGE
MÉMOIRE PRÉSENTÉ COMME EXIGENCE PARTIELLE DE LA MAÎTRISE EN
COMMUNICATION
Par : FRANÇOIS BASTIEN
Janvier 2006
- [9] : MEMOIRE DE MAGISTER EN AUTOMATIQUE
Option : Traitement d'Images et Reconnaissance de Formes
Présenté par : M^{elle} BEDOUHENE Saïda ingénieur U.M.M.T.O
Thème : Recherche d'images par le contenu.
- [10] : wikipedia.org/wiki/Extraction_de_caractéristique_en_vision_par_ordinateur
- [11] : FACULTÉ POLYTECHNIQUE DE MONS
Développement d'une méthode de représentation d'images sur base d'un
Vocabulaire visuel
Travail de fin d'études :
Simon PIERARD
- [12] : Détection et mise en correspondance de points d'intérêt pour la propagation de germes
Master 2 Recherche
Spécialité Image, Information et Hypermédia
Par : Guillaume GALES
RAPPORT DE STAGE 2006-2007
- [13] : <http://morpheo.inrialpes.fr/people/Boyer/Teaching/M2PGI/c4.pdf>
- [14] : stracoverflow.com

Bibliographie / Webographie

Chapitre III

- [1] : Adsl Dépannage Déclarations à faire Définir vos besoins
Important Informatique et réseau
Lecture plaque immatriculation Protocoles VPN /Récupération de Données serveur
Sauvegardes /VidéoSurveillance
- [2] : Alphanumeric Vision
86 rue de Paris - Bâtiment Erable 91400 ORSAY
contact@alphanumericvision.fr
www.alphanumericvision.com
- [3] : Sétra Service d'études sur les transports, les routes et leurs aménagements
Rapport d'études
Systèmes et équipements de lecture automatique
de plaques d'immatriculation des véhicules
Principes et aperçu des applications
mai 2013
- [4] : https://fr.wikipedia.org/wiki/Lecture_automatique_de_plaques_minéralogiques
- [5]: IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,
SEPTEMBER 2008
License Plate Recognition From Still Images and Video Sequences: A Survey
- [6]: Cloud-Based Car Image Retrieval with Interactive Script
Changyou Zhang, Bo Huang, Hongcai Chen, Yu Cheng, Li Wang, and Ting Wang
- [7]: Sarfraz.M , Ahmed.M.J, et.al(2003),“Saudi Arabian licence plate recognition system”,
International Conference on Geometric Modeling and Graphics (GMAG'03).
- [8]: Ozbay.S, and Ercelebi.E(2005),” Automatic Vehicle Identification by Plate Recognition”,
Processing of world academy of science engineering and technology vol9,
ISSN 1307-6884.
- [9]: Pearce, G., Pears, N.: Automatic Make and Model Recognition from Frontal Images of
Cars. In: 8th IEEE International
Conference on Advanced Video and Signal-Based Surveillance, pp. 373-378 (2011)
- [10]: <http://www.top500.org/>
- [11]: <http://julialang.org/>
- [12]: Josef Sivic,Andrew Zisserman , Video Google: A Text Retrieval Approach to Object
Matching in Videos, international conference on computer vision 2003

Bibliographie / Webographie

- [13]: Belongie, S., Malik, J., & Puzicha, J. (2002).
Shape matching and object recognition using shape contexts. IEEE Transactions
On Pattern Analysis and Machine Intelligence, 24 (4), pp. 509-522.
Canny, J. (1986). A computational approach to edge detection.
IEEE Trans. Patt. Anal. Mach. Intel., PAMI-8 , pp. 679-698
- [14]: Hsieh, J.-W., Yu, S.-H., & Chen, Y.-S. (2002).
Morphology-based license plate detection from complex scenes.
Proceedings. 16th International Conference on Pattern Recognition, 3,
pp.176- 179.
- [15]: Gao, D.-S., & Zhou, J. (2000, August 21).
Car License Plates Detection from Complex Scene.
5th International Conference on Signal Processing Proceedings. WCCC-
ICSP 2000. , 2, pp. 1409- 1414
- [16]: Deb, K., Chae, H.-U., & Jo, C. K.-H. (2008, April 9-11).
Parallelogram and Histogram based Vehicle License Plate Detection, International
Conference on Smart Manufacturing Application, ICSMA. , pp. 349-353.
- [17]: Nijhuis, J., Ter Brugge, M., Helmholt, K., Pluim, J., Spaanenburg, L.& Venema,
R.(1995, November-December 27-01).
Car license plate recognition with neural networks and fuzzy logic, Proceedings,
IEEE International Conference on Neural Networks , pp. 2232 - 2236.
- [18] : Anagnostopoulos, C., Anagnostopoulos, I., Loumos, V., & Kayafas, E. (2006).
A License Plate-Recognition Algorithm for Intelligent Transportation System
Applications. IEEE Transactions on Intelligent Transportation Systems, 7
(3), pp. 377-392.
- [19] : Dlagnekov, L. (2005). Car license plate, make, and model recognition.
In Proc. IEEE Conf. On Computer Vision and Pattern Recognition .

Chapitre IV

- [1] : H.C Kuo, J.S. Lee, S.T. Kao. An Autonomous License Plate Detection Method.
Proceedings of the fifth International Conference on Intelligent Information Hiding and
Multimedia Signal; 2009:110-113
- [2]: <https://fr.wikipedia.org/wiki/Ubuntu>
- [3]: [https://fr.wikipedia.org/wiki/Virtual machine](https://fr.wikipedia.org/wiki/Virtual_machine)

Bibliographie / Webographie

- [4]: [https://fr.wikipedia.org/wiki/ VirtualBox](https://fr.wikipedia.org/wiki/VirtualBox)
- [5]: <http://perso.esiee.fr/~perretb/I5FM/TAI/OpenCVEnBref.pdf>
- [6]: <http://wcours.gel.ulaval.ca/2014/h/GEL3014/default/5notes/documents/opencv.pdf>
- [7]: [https://fr.wikipedia.org/wiki/ XML](https://fr.wikipedia.org/wiki/XML)
- [8]: <https://fr.wikipedia.org/wiki/Java>
- [9]: <http://philippe.petitpa.perso.sfr.fr/tutoHARR.pdf>
- [10]: <http://chamilo2.grenet.fr/inp/courses/ENSIMAG4MMTI/document/TP/TP1/tp1.pdf>
- [11]: <http://chamilo2.grenet.fr/inp/courses/ENSIMAG4MMTI/document/TP/TP1/tp1.pdf>
- [12]: <http://stackoverflow.com/questions/20276209/find-the-plate-rectangle-in-a-given-picture>
- [13]: http://www.tutorialspoint.com/java_dip/basic_thresholding.htm
- [14]: <https://arxiv.org/ftp/arxiv/papers/1005/1005.4020.pdf>
- [15]: <https://arxiv.org/ftp/arxiv/papers/1005/1005.4020.pdf>
- [16]: <https://arxiv.org/ftp/arxiv/papers/1005/1005.4020.pdf>
- [17]: <http://dept-info.labri.fr/~vialard/Traitement/cours/cours2.pdf>
- [18]: http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_core/py_image_arithmetics/py_image_arithmetics.html
- [19]: http://abaque.ca/a15/info6501/docs/EChap4_mmbinaire_FR.pdf
- [20]: http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html
- [21]: <https://dpt-info.u-strasbg.fr/~cronse/TIDOC/MM/deof.html>