Complex networks for image and video Analysis

Youssef Mourchid

ROYAUME DU MAROC
Université Mohammed V - RABAT -
جامعة محمد الخامس – الرباط –
Faculté des sciences
كلية العلوم
CENTRE D'ETUDES DOCTORALES - SCIENCES ET TECHNOLOGIES

N° d'ordre ……

THESE

En vue de l'obtention du : ***DOCTORAT***

Structure de Recherche : Laboratoire de Recherche en Informatique et Télécommunications
Discipline : Sciences de l'ingénieur
Spécialité : Informatique et télécommunications

Présentée et soutenue le 28/09/2019 par :

Youssef MOURCHID

Réseaux Complexes pour l'Analyse des Images et des Vidéos

JURY

Salma MOULINE	PES, Faculté des Sciences, Université Mohammed V de Rabat	Présidente
Mohammed EL HASSOUNI	PES, Faculté des Lettres et des Sciences Humaines, Université Mohammed V de Rabat	Directeur de thèse
Belaïd AHIOD	PH, Faculté des Sciences, Université Mohammed V de Rabat	Rapporteur/Examinateur
Rochdi MESSOUSSI	PES, FSK-Université Ibn Tofail de Kenitra, Maroc	Rapporteur/Examinateur
Khadija RHOULAMI	PH, Faculté des Lettres et des Sciences Humaines, Université Mohammed V de Rabat	Rapporteur/Examinateur
Hocine CHERIFI	PES, Université de Bourgogne, Dijon, France	Examinateur
Benjamin RENOUST	PA, l'Institut de Science de la databilité, Université d'Osaka, Japan	Co-encadrant

Année Universitaire : 2018-2019

À mes parents
À mes frères
À mes soeurs
À mes amis

Twenty years from now you will be more disappointed by then things that you didn't do than by the ones you did do. So throw off the bowlines. Sail away from the safe harbor. Catch the trade winds in your sails. Explore. Dream. Discover.
Mark Twain

Remerciements

Cette thèse de doctorat a été menée pour l'obtention du grade de Docteur de l'Université Mohammed V. Les travaux présentés dans ce mémoire ont été effectués au Laboratoire de Recherche en Informatique et Télécommunications (LRIT) de la Faculté des Sciences de Rabat (FSR) au Maroc sous la direction du Professeur **Mohammed EL HASSOUNI** et en collaboration avec le Laboratoire d'Informatique de Bourgogne (LIB) de l'Université de Bourgogne sous le co-encadrement du Professeur **Hocine CHERIFI**, et l'Institue de Science de la Databilité à l'Université d'Osaka sous le co-encadrement de du Professeur **Benjamin RENOUST**.

En premier lieu, je tiens à remercier mon directeur de thèse Professeur **Mohammed EL HASSOUNI**, Professeur d'Enseignement Supérieur à la faculté des lettres et sciences humaines de Rabat, avec qui j'ai travaillé depuis mon projet de fin d'études du Master. Ses conseils précieux ont bien mener à bien cette thèse. Au-delà de ses qualités scientifiques, il est avant tout un homme généreux, avec qu'il est agréable de discuter, d'échanger des idées, et de plaisanter. J'espère que ces immenses qualités continueront à m'influencer pour longtemps.

Je suis également reconnaissant au Professeur **Hocine CHERIFI**, Professeur à l'Université de Bourgogne de Dijon, France, pour l'encadrement de mes travaux de recherche. Ses conseils, sa disponibilité, les nombreuses discussions et échanges ont grandement contribué à la réussite de cette thèse. Je le remercie aussi d'avoir accepté de participer en tant qu'examinateur dans le jury final.

Je suis particulièrement redevable au Professeur **Benjamin RENOUST**, Professeur Associé à l'université d'Osaka, Japan pour son engagement dans le co-encadrement de ma thèse. Ces conseils et son talent m'inspireront certainement tout au long de ma carrière. Cette thèse doit énormément à sa grande disponibilité, son dynamisme, ses encouragements, son ouverture d'esprit et bien sûr ses qualités scientifiques exceptionnelles.

Je tiens à remercier Professeur **Salma MOULINE**, Professeur d'Enseignement Supérieur à la faculté des sciences de Rabat, d'avoir accepté de présider le jury de ma thèse.

Mers remerciements vont également à Professeur **Belaïd AHIOD**, Professeur Habilité à la faculté des sciences de Rabat, d'avoir accepté de rapporter ce mémoire de thèse.

Je tiens aussi à remercier Professeur **Rochdi MESSOUSSI**, Professeur d'Enseignement Supérieur à la faculté des sciences de Kenitra, d'avoir accepté de juger la qualité de mon travail en tant que rapporteur.

Je voudrais également remercier Professeur **Khadija RHOULAMI**, Professeur Habilité à la faculté des lettres et sciences humaines de Rabat, qui m'a fait l'honneur de rapporter cette thèse.

Je remercie mes amis qui m'ont apporté leur soutien et leurs encouragements tout au long de cette aventure. Dans le désordre, Ahmed(s), Haytam, Mohamed, Ilyass(s), Ayoub, Selma, Leila, Mourad, Noureddine, Oussama, Hicham, Mounir(s), Hassan, Zakariya.

Je garde le meilleur pour la fin, ma famille qui a supporté toutes les difficultés morales et matérielles pour me soutenir au terme de mes études. J'adresse ma profonde gratitude et mon immense reconnaissance à mes raisons d'être, ma mère et mon père, qui m'ont éduqué et orienté. Merci de m'avoir encouragé et soutenu dans mes choix. Nul mot et nulles expressions refléteront le grand amour et la profonde gratitude que je porte pour vous. Mes plus sincères remerciements vont aussi à mes frères et soeurs. Aucun mot ne saurait retranscrire ici sans l'affaiblir le bonheur qu'ils m'ont toujours apporté, ni l'ampleur de ce que je leur dois.

Table des matières

Résumé

L'analyse des données visuelles est toujours une problématique d'intérêt dans différents domaines d'application (surveillance, automobile, sécurité, environnement, médecine,). Dans le cadre du travail réalisé, nous nous sommes intéressés au développement des méthodologies pour analyser les données visuelles, notamment, les images et les vidéos (films). Dans ce travail de thèse, on s'adresse à deux problématiques de la littérature, la segmentation pour les images et l'analyse des histoires pour les vidéos, plus précisément les films. Différentes techniques ont été développées dans la littérature pour l'analyse du contenu visuel, mais qui souffrent de certaines limites selon la nature des données traitées, la précision, la robustesse, la sémantique, etc. En dehors de ces méthodes, il s'est avéré que les approches fondées sur les réseaux complexes sont un très bon outil pour résoudre les deux problématiques mentionnées. Dans ce sens, le travail de cette thèse est divisé en deux parties : La première partie vise à proposer un framework qui est basé sur les approches graphes, notamment les algorithmes de détection des communautés, pour résoudre le problème de la segmentation des images. Quant à la seconde partie, elle propose un modèle multi-couches basé sur les graphes pour analyser les histoires des films. Ce modèle exploite le script du film pour construire les liens entre les différentes entités qui composent l'histoire du film. Une extension de ce modèle a été proposée en utilisant les informations textuelles (script et sous-titres) et les informations visuelles (contenu multimédia, reconnaissance faciale, détection des captions) du film. Ces informations permettent d'extraire une structure plus riche du film. Les propriétés des graphes construits ont été exploitées par la suite pour analyser l'histoire du film (ex. propriétés topologiques, centralités, la structure communautaire).

Mots clés : analyse visuelle, réseaux complexes, détection des communautés, multi-couches, segmentation des images, analyse des vidéos, films, analyse des histoires, traitement automatique du texte (TAL), reconnaissance faciale.

Abstract

Visual data analysis is always a problem of interest in different fields of application (surveillance, automobile, security, environment, medicine, etc.). In this work, we focus on the development of methodologies for analyzing visual data, including images and videos (movies). We were interested in developing tools dedicated to the analysis of visual content from images and videos (movies). In this thesis, we address two issues, image segmentation and analysis of video stories, especially movie stories. Several techniques were developed in the literature for visual content analysis, which almost suffer from certain limitations depending on the processed data, precision, robustness, semantics, etc. Beyond these proposed methods, it has been shown that complex network approaches are a very good tool to address the two mentioned issues. In this context, this work is divided into two parts : The first part aims to propose a framework based on graph approaches, in particular, community detection algorithms to address the problem of image segmentation. As for the second part, it proposes a multi-layer model based on graphs to analyze movie stories, this model exploits the script of the movie to build interactions between the different entities that constitute the movie story. An extension of this model was proposed next, using the textual information (script and subtitles) and the visual information (multimedia content, facial recognition, dense captioning) of the movie. These informations can extract a richer structure of the movie. Graph properties were exploited next, to analyze the movie story (eg. topological properties, centralities, community structure).

Keywords : visual analytics, complex networks, community detection, multilayer, image segmentation, video analysis, movies, story analysis, natural language processing (NLP), face recognition.

Liste des abréviations

IRM *Imagerie par résonance magnétique*

URL *Uniform Resource Locator*

GN *Girvan-Newman*

EV *Eigen vectors*

LPA *Label Propagation algorithm*

RGB *Red Green Blue*

HSV *Hue Saturation Value*

LBP *Local Binary Pattern*

GLCM *Gray-Level Co-Occurrence Matrix*

CTM *Compression-based Texture Merging*

ACP *Analyse en Composantes principales*

NCUT *Normalized CUT*

MCL *MArkov Cluster Algorithm*

RAG *Region Adjacency Graph*

BSD *Berkeley Segmentation Dataset*

SSDS *Semantic Segmentation Data Set*

PRI *Probabilistic Rand Index*

VOI *Variation of Information*

EMD *Earth Mover's Distance*

KL *Kullback-Leibler*

MD *Mean Distance*

HOG *Histogram of Oriented Gradients*

FMCDRN *Fast multi-scale community detection algorithm using the criterion from Ronhovde and Nussinov*

FGMDO *Fast greedy modularity optimization algorithm*

TABLE DES FIGURES

Liste des tableaux

Liste des algorithmes

Chapitre

1 Introduction générale

Sommaire

1.1 Contexte et motivation

Le plus important des cinq sens que possède l'être humain, c'est le système visuel, et pour cause, presque 80% des informations captées par notre cerveau proviennent de ce système. Avec l'aide du cerveau, le système visuel cherche à interpréter l'information qu'il reçoit selon des schémas préalablement appris. Il est principalement constitué de l'oeil et plus particulièrement la rétine et des nerfs optiques. L'image est alors analysée au sein même de la rétine en zones de contraste et le résultat de ce traitement est envoyé au reste du système visuel par le nerf optique. L'image sous toutes ses formes occupe une place si importante dans notre société actuelle : peinture, photographie, films, imagerie médicale, publicité, vidéo surveillance, *etc.* Malgré que l'image peut être très riche en informations, notre cerveau arrive à dissocier son contenu la plupart du temps, en reconnaissant les différents éléments représentés. Cependant, si nous arrivons à faire cela sans peine après des années d'apprentissage depuis notre enfance, il est plus compliqué d'imiter ce phénomène artificiellement. La vision artificielle (qui peut être appelée aussi par la vision par ordina-

teur) est un problème de traitement et d'analyse d'images et des vidéos correspondant à une branche de l'intelligence artificielle. Celle-ci a pour but d'apprendre à la machine de "comprendre" ce qu'elle "voit" en reproduisant les caractéristiques supposées de la vision humaine. Marr et Hildreth (1980) a proposé à la fin des années 70, un paradigme de la vision par ordinateur qui est défini par les quatre points suivants :

- Pour délimiter et structurer les objets d'une image, il faut connaître ses contours.
- On peut accéder à la troisième dimension à partir d'une image 2D et de connaissances sur le monde 3D.
- On peut estimer le mouvement des séquences d'images.
- Pour l'extraction du relief, on peut utiliser L'ombre.

Les applications du traitement et de l'analyse de l'image sont diverses. Les problèmes qui en découlent sont donc extrêmement nombreux. Parmi les exemples de problèmes les plus couramment rencontrés :la restauration et le débruitage, qui sont nécessaires pour améliorer le rendu d'images dont la qualité d'acquisition a été insuffisante ; la reconstruction tomographique, qui est utilisée en imagerie médicale par Imagerie par Résonance Magnétique(IRM) ; la reconnaissance, le suivi ou la recherche de points d'intérêts d'un objet, qui sont requis pour un contrôle qualité ou une vidéo-surveillance automatique et finalement le problème le plus connu, *la segmentation des images*. La compréhension de l'image a été un domaine de recherche actif au cours de ces dernières années. Cet intérêt a été soutenu par de nombreuses applications pratiques, telles que la détection et la reconnaissance des visages, la détection humaine, la récupération d'images et la catégorisation (voir plusieurs exemples dans la figure 1.1). Par exemple, un succès remarquable de la vision par ordinateur, repose sur plusieurs travaux innovants, qui exploitent une vaste collection de données d'images numériques pour former des algorithmes d'apprentissage automatique, permettant de détecter des visages dans des images (Viola et Jones, 2004; Rowley, 1999). Certaines applications, telles que la reconnaissance des gestes, la vidéo surveillance, *etc*, nécessitent l'analyse de la dynamique visuelle. Pour ce but, des séquences

temporelles d'images, de vidéos, sont utilisées. Une vidéo est une séquence d'images, capturées à une vitesse temporelle régulière (fréquence d'images). Les premières recherches sur l'analyse automatique de la vidéo étaient axées sur les applications de surveillance, qui aident les humains à prévenir les situations dangereuses (Forsyth et Ponce, 2003).

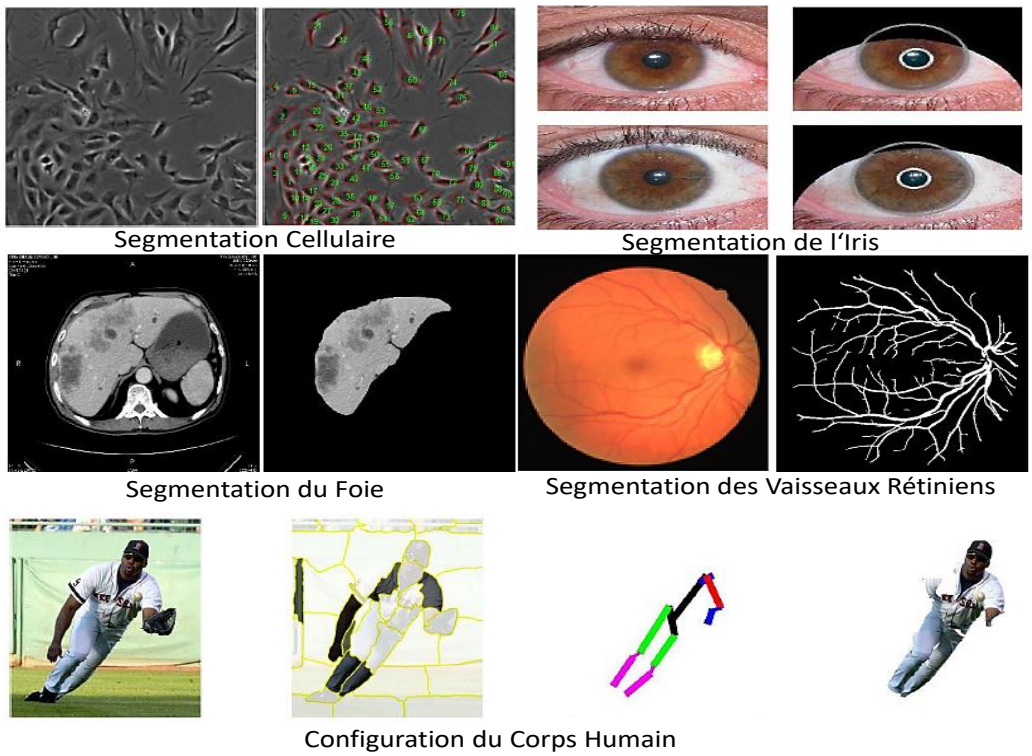

FIGURE 1.1 – Illustration des différentes applications de la segmentation des images.

Les données vidéo sont souvent longues et, par conséquent, prennent beaucoup de temps à être analyser. Identifier les parties les plus pertinentes dans une vidéo par rapport au but recherché est une tâche importante. Le résumé d'une vidéo, recherche des moyens pour représenter une vidéo sous une forme plus compacte, en ne conservant que les données pertinentes. Les applications connues du résumé vidéo (Truong et Venkatesh, 2007) incluent la recherche de base de données semi-automatisée, la surveillance vidéo, la détection de moments forts dans le sport, les moments d'influence dans les vidéos égocentriques et les points culminants dans les films (voir plusieurs exemples dans la figure 1.2). En particulier, les films fournissent une grande quantité de données vidéo réalistes et constituent un "laboratoire d'expérimentation" utile pour les chercheurs en vision par ordinateur (Sivic et Zisserman, 2003; Laptev et Pérez, 2007). Une propriété intéressante

des données du film est la participation de centaines voire de milliers de personnes à la création du film. Par conséquent, chaque seconde du film contient un sens implicite : les cinéphiles doivent regarder un film plusieurs fois pour parvenir à une compréhension complète. Les données du film sont donc assez difficiles. Ces énormes quantités de contenu audiovisuel ont largement dépassé la capacité que possède l'être humain de les visualiser et ont rendu difficile la recherche de contenus intéressants pour un utilisateur. Le besoin d'outils efficaces permettant aux utilisateurs de choisir le contenu qu'ils vont regarder est donc manifeste. Parmi les solutions proposées pour ce problème est de de faire une analyse sémantique de la vidéo sélectionnée. Cette analyse permet de répondre à ce besoin en fournissant un aperçu général et rapide de l'ensemble du contenu audiovisuel de la vidéo originale et en présentant les parties intéressantes pour l'utilisateur. Dans ce contexte, *l'analyse des films* fournit une assistance permettant aux professionnels d'économiser du temps et de réduire considérablement le coût de la recherche/annotation des films. Par conséquent, nous considérons les données du film comme un moyen de développer des outils plus sensibles pour l'analyse de la vidéo.

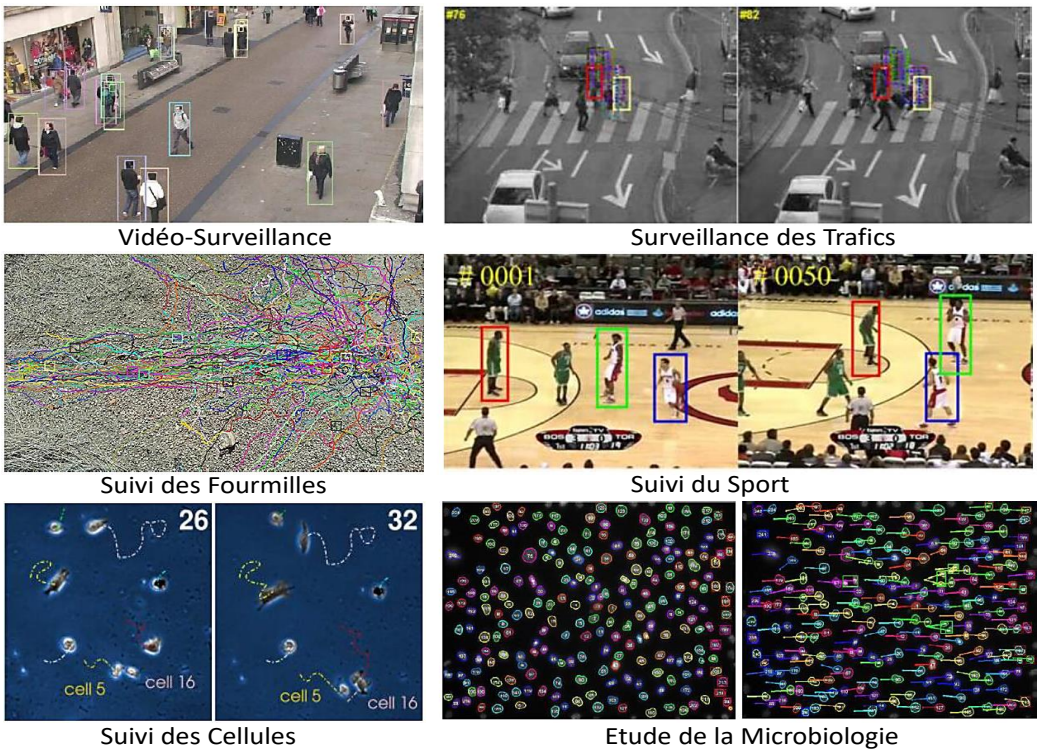

FIGURE 1.2 – Illustration des différentes applications de l'analyse des vidéos.

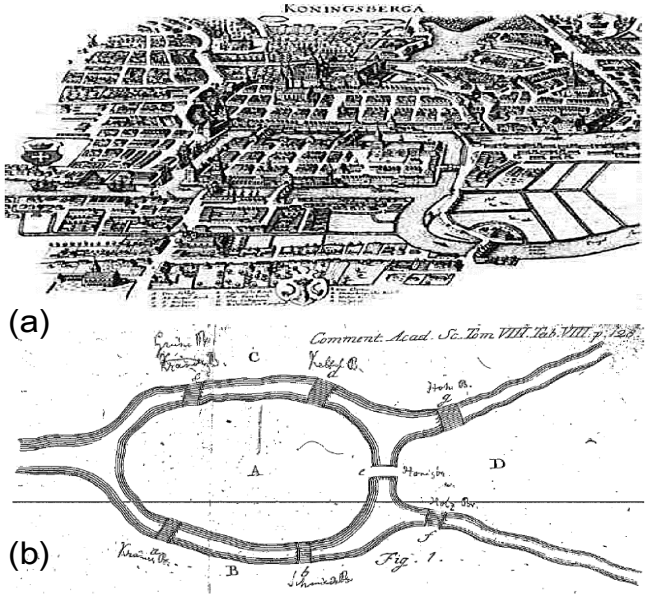

FIGURE 1.3 – (a) Plan de Königsberg au XVIIIème siècle. (b) Illustration des ponts de Königsberg issue de (Euler, 1741). Peut-on se promener, en revenant à son point de départ, en passant une fois et une seule fois par tous les ponts ?

Les graphes font également partie des outils mathématiques couramment utilisés en traitement et analyse d'images et des vidéos pour, notamment, la segmentation pour les images et faire l'analyse pour les films. Bien que l'on considère souvent la théorie des graphes comme remontant également aux années 1960 avec les bases posées par Berge (2001), leur première apparition est due à Leonhard Euler en 1736. Ce dernier, considéré par la plupart comme le mathématicien le plus productif dans la théorie des graphes et de la topologie, c'est celui qui a apporté une réponse au problème des sept ponts de Königsberg dans (Euler, 1741). Königsberg est une ville traversée par un fleuve, le Pregel, et au milieu duquel se trouvent deux îles reliées entre elles et aux berges par sept ponts (comme illustré dans la figure 1.3). Le problème était le suivant : peut-on se promener, en revenant à son point de départ, en passant par tous les ponts juste une seule fois ? Leonhard Euler prouva à l'aide des prémices de la théorie des graphes sa réponse : non.

Le champ applicatif des graphes s'est grandement répandu dans de nombreux domaines, depuis les travaux de Leonhard Euler. Les graphes sont désormais susceptibles de représenter divers réseaux (routiers, ferrés, de canalisations, électriques, informatiques,...) et d'être utilisés dans la résolution des problèmes, à savoir la recherche du plus court

chemin, de la tournée d'un livreur la plus courte, du goulot d'étranglement du réseau, *etc.* Comme déjà annoncés, les graphes sont également très utilisés en traitement et analyse d'images et des films, puisque comme nous le verrons en détails dans le chapitre 4, une image peut être représentée sous forme d'un graphe. Différentes approches basées sur les graphes ont donc pu être appliquées dans ce domaine et d'autres furent développées spécialement pour. Puisque la segmentation d'images est devenue un problème récurrent dans le traitement et l'analyse d'images, il n'est donc pas étonnant que la segmentation de graphes le soit également. Dans le chapitre 6, on va voir comment on peut représenter un film sous forme d'un graphe, plus précisément un graphe multi-couches *(en anglais multilayer)* qui contient les différentes entités qui compose le film. Différents travaux ont été proposés pour ce but, mais qui sont plus au moins limités. C'est dans ce contexte s'inscrit ce mémoire et le manuscrit qui en résulte.

1.2 Problématiques et objectifs

1.2.1 Segmentation des images

La segmentation des images est un problème fondamental en vision par ordinateur. Il s'agit de diviser l'image en régions uniformes et homogènes correspondant à des parties significatives de l'image. Le domaine d'application de la segmentation d'images varie de l'identification d'objets allant des données de télédétection (Chen *et al.*, 2017; Liu *et al.*, 2016) à la détection de cellules cancéreuses. Par exemple, il peut être utilisé pour diagnostiquer l'imagerie médicale (Noble et Boukerroui, 2006) ou pour extraire des points d'intérêt pour des images afin d'identifier les caractéristiques locales d'une image (Zou et Bai, 2018). Dans la littérature, de nombreux algorithmes de segmentation d'images ont été proposés. Ils peuvent être classés en deux catégories principales : la détection des contours et les approches basées sur les régions. Les méthodes de détection des contours reposent sur l'utilisation de discontinuités pour détecter les contours d'une segmentation d'image. Plusieurs méthodes (Gonzalez et Woods, 1992; Bao *et al.*, 2005) sont proposées dans cette catégorie, qui reposent sur les changements brusques d'intensité ou de la couleur de l'image. Dans les approches basées sur les régions, qui constituent une autre catégorie

populaire de méthodes de segmentation, la segmentation est réalisée de manière itérative, jusqu'à ce que certains critères d'uniformité soient satisfaits. Les principales méthodes de cette catégorie reposent sur le seuillage (Otsu, 1979), la croissance régionale (Wani et Batchelor, 1994) et les graphes (Peng *et al.*, 2013). Récemment des techniques basées sur les graphes ont été proposés pour représenter les composants de l'image en structures mathématiquement saines, ce qui facilite le problème de la segmentation et rend le calcul plus rapide et efficace. Le problème des méthodes de segmentation d'images basées sur des graphes repose sur le partitionnement en plusieurs sous-graphes, de telle sorte que chaque sous-graphe représente un objet d'intérêt significatif dans l'image. L'idée d'utiliser des graphes comme approche de la segmentation a été proposée par Wu et Leahy (1993). À partir de ce moment, l'étude des techniques d'optimisation sur le graphe a beaucoup attiré l'attention de la recherche (Felzenszwalb et Huttenlocher, 2004; Shi et Malik, 2000; Li et Wu, 2014; Abin *et al.*, 2014; Linares *et al.*, 2017). Cependant, ces méthodes sont généralement sensibles au bruit et utilise soit la couleur ou la texture comme mesure pour calculer la similarité entre les régions de l'image, ce qui conduit à une sur-segmentation en négligeant les régularités à l'intérieur de l'image. De plus, la plupart de ces méthodes reposent sur des algorithmes qui ont un coût de calcul élevé.

Pour surmonter ces limitations. Nous proposons un framework en utilisant un graphe pondéré de régions pour représenter une image, les régions représentent des noeuds du graphe. Un lien entre deux noeuds est considéré si les deux régions sont adjacentes, puis le poids est calculé à l'aide des caractéristiques de l'image (couleur et texture).

Généralement les graphes sont structurés en communautés, qui représentent des groupes de noeuds fortement connectés entre eux mais faiblement reliés aux noeuds d'autres groupes Blondel *et al.* (2008). En tenant compte de l'importance de la détection communautaire, il n'est pas surprenant que de nombreuses méthodes de détection communautaire ont été développées en utilisant des techniques de différentes disciplines. Il est donc possible d'envisager ces méthodes d'identification d'objets dans une image. Plus spécifiquement, une image peut être représentée sous forme d'un graphe et des approches de détection de communautés peuvent être envisagées pour identifier les régions (objets dans une image), qui correspondent aux communautés dans le graphe.

1.2.2 Analyse des vidéos (films)

Que ce soit à travers des livres ou des films, les histoires créent leur propre univers, en mettant en jeu des personnages dans un monde virtuel. Les interactions des différents éléments d'une histoire forment un monde imaginaire susceptible d'attirer l'attention des lecteurs ou des téléspectateurs. Les livres laissent à leurs lecteurs l'assemblage d'éléments d'histoire par leur propre imagination pour que chacun puisse construire sa propre vision du monde représenté. Les films sont très différents dans ce point, car ils offrent aux téléspectateurs un monde entièrement construit qu'ils ne peuvent qu'en profiter avec émerveillement. Lorsque certains réalisateurs aiment jouer avec la perception progressive des téléspectateurs de leur univers créé, d'autres présentent vraiment des mondes imaginaires riches comme dans les films de science-fiction. On pourrait souhaiter de comprendre si les interactions des éléments de l'histoire peuvent donner une vue globale sur elle, en caractérisant un genre ou un réalisateur. Les interactions entre les éléments d'une narration ont souvent été capturées grâce à la modélisation par graphe (Park *et al.*, 2012; Waumans *et al.*, 2015; Tan *et al.*, 2014; Renoust *et al.*, 2016a). elle a été utilisée pour soutenir la narration d'un large ensemble d'histoires, des livres (Waumans *et al.*, 2015), des séries télévisées (Tan *et al.*, 2014), des reportages d'actualité (Renoust *et al.*, 2016a) et des films (Park *et al.*, 2012), qui sont au centre de nos préoccupations. Les graphes sont des objets visuels qui peuvent non seulement être intelligemment visualisés pour expliquer les récits, mais leur structure peut également être interrogée sous l'angle de la topologie (Waumans *et al.*, 2015; Rital *et al.*, 2005).

Ces travaux sont limités. En effet, ils se concentrent principalement sur une seule facette de l'histoire, principalement les personnages en jeu. Les journalistes enquêtent souvent sur une histoire en articulant les 5 questions : Qui ?, Où ?, Quoi ?, Quand ? et comment / pourquoi ? (Chen *et al.*, 2009; Kipling, 1998). L'analyse des histoires par les graphes tente de répondre à la question "Comment / Pourquoi ?" en articulant les 4 autres éléments d'un graphe. Depuis que la question quand ? est fourni avec des données (selon la narration), les travaux précédents ont principalement porté sur Qui ? ou Où ?, dans un réseau mono-couche. Nous souhaitons introduire une approche plus globale pour aborder Qui ? ou Où ? et Quoi ? dans le même graphe, en reposant sur la modélisation du réseau

multi-couche des histoires. Bien entendu, le graphe peut être créé manuellement (Mish, 2016), mais les approches automatisées peuvent même s'adapter à des archives plus volumineuses (Renoust *et al.*, 2016a; Waumans *et al.*, 2015). Comme il est toujours difficile d'obtenir toutes les informations nécessaires à partir des données vidéo elles-mêmes (Demirkesen et Cherifi, 2008; Pastrana-Vidal *et al.*, 2006), nous pouvons nous fier à l'analyse de texte et du contenu de la vidéo. Heureusement, au tout début de leur création, les films sont écrits sous forme de scripts. Un script est généralement extrêmement bien structuré et contient tous les composants nécessaires pour analyser automatiquement un film (Jhala, 2008) (scènes, dialogues, personnages, *etc.*). Toutefois, le rôle du script peut se trouver affecté, voir annulé, si un problème d'accès à la langue du film vient se poser. En effet, les barrières de langues ont mené à la création et à la mise en place de techniques permettant de rétablir l'accès au sens des dialogues, dont le sous-titrage. Différents types de sous-titres sont utilisés, en fonction de la langue des dialogues et des éventuels besoins des spectateurs. Lorsque la langue des sous-titres est la même que celle présentée dans les dialogues, on parle de « sous-titrage intra-langue ». Notre travail se concentre sur ce type de sous-titrage. Étant donné que le script ne contient pas d'informations temporelles, des processus supplémentaires sont nécessaires pour synchroniser le script avec le film. Le sous-titre est généralement créé à partir du dialogue du script et contient des informations temporelles permettant d'afficher du texte à l'écran lorsque les personnages parlent. Cependant, les sous-titres n'incluent pas d'informations relatives aux personnages, scènes, plans et actions. Par conséquent, si les sous-titres sont alignés avec le script, nous pouvons extraire et analyser des informations détaillées pour le film.

1.3 Contributions de la thèse

Le but de cette thèse est de contribuer à l'état de l'art dans la segmentation d'images et l'analyse des films à l'aide des méthodes basées sur des graphes. A cette fin, trois contributions sont introduites :

- Notre première contribution est de proposer un framework pour la segmentation d'images, basé sur des algorithmes de détection de communauté. Premièrement,

nous proposons de modéliser une image avec un graphe d'adjacence pondéré des régions qui se base sur les propriétés topologiques et visuelles des images (texture, couleur). Deuxièmement, nous utilisons des algorithmes de détection de communauté développés dans le paradigme des réseaux complexes afin de résoudre le problème de la segmentation. Troisièmement, un processus itératif est proposé afin d'éviter les problèmes de sur-segmentation.

- La deuxième contribution sera cette fois sur l'analyse de la vidéo, notamment les films. Dans ce travail on propose d'exploiter l'information textuelle pour extraire automatiquement les graphes des films à partir des scripts. Un script est généralement bien structuré et contient tous les composants nécessaires pour analyser automatiquement un film (scènes, dialogues, personnages, *etc.*). Nous contribuons avec un nouveau modèle multi-couche, en se basant sur le dialogue du film, qui permettra d'articuler des personnages, des lieux et des mots-clés. Le graphe multi-couche capture une structure plus riche du film. Il complète l'analyse du graphe des caractères et apporte de nouveaux outils d'analyse topologique.

- Dans la troisième contribution, Nous étendons la deuxième approche sur plusieurs directions. Au lieu d'analyser des films indépendamment, nous étendons notre modèle à plusieurs films d'une même saga (une histoire plus grande composée de plusieurs histoires). Nous ne limitons pas notre modèle sur juste le dialogue du film, mais nous exploitons également les sous-titres (pour avoir la notion du temps) et le contenu multimédia du film en intégrant l'analyse à partir de l'image (via un sous-titrage des images et une analyse des visages).

1.4 Organisation du manuscrit

Hormis l'introduction générale qui constitue le chapitre 1, nous débutons dans le chapitre 2 par des définitions générales sur les réseaux complexes, puis, au fur et à mesure, nous présentons de nouvelles définitions plus spécifiques à la segmentation et l'analyse de graphes pour finalement nous orienter vers les deux applications (image et film) qui se trouvent dans les deux parties de ce manuscrit. La première partie sera consacrée aux

travaux précédents (chapitre 3) sur la segmentation des images puis l'approche proposée à ce but (chapitre 4). La deuxième partie se focalise sur l'état de l'art des méthodes proposées pour l'analyse des films (chapitre 5), puis on va présenter nos deux approches proposées (chapitres 6 et 7) pour analyser l'histoire des films. Nous conclurons avec une conclusion générale, qui présentera notamment les perspectives que nous donnons à ces travaux. Enfin figure en annexe les résultats complémentaires que nous avons effectués pour évaluer et comparer les différentes approches des contributions qui nous ont été utiles dans le cadre de nos recherches.

Chapitre

2 Les réseaux complexes

Sommaire

2.1 Introduction

Nous présentons dans ce chapitre le socle théorique utilisé tout au long de nos contributions. Nous commençons par la présentation des concepts généraux sur les réseaux complexes. Différentes études ont montré que les graphes (réseaux complexes) de terrain exhibent des propriétés structurelles communes, parmi ces propriétés on trouve qu'ils sont souvent composés de sous-graphes denses faiblement interconnectés, appelés communautés. Nous exposons dans la suite de ce chapitre les fonctions de qualité nommées modularité et stabilité qui sont fréquemment utilisées dans l'état de l'art, comme critères d'évaluation de la qualité de découpage d'un graphe en communautés. Nous décrivons ainsi quelques méthodes de détection de communautés utilisées pour optimiser la modularité avant de souligner les principales limites de l'optimisation de la modularité. Ensuite, nous exposons les différentes approches alternatives pour parer ces limites.

2.2 Définitions et notations

De manière générale, les systèmes complexes sont des systèmes difficiles à comprendre, à modéliser et à prévoir. Le terme système désigne généralement un ensemble de composants inter-reliés et interdépendants (réels ou abstraits) qui forment tout un ensemble. La principale caractéristique de ces systèmes complexes est que leur comportement ne peut pas être déduit du comportement de composants individuels. Les systèmes complexes se caractérisent par un grand nombre de composants qui peuvent interagir de différentes manières. Une variété de systèmes complexes peut être décrite par des réseaux qui montrent des relations, des dépendances et des interactions entre leurs composants constitutifs. Les réseaux sont composés de noeuds (également appelés sommets, sites ou acteurs) et de liens (également appelés arêtes) reliant les noeuds. D'un point de vue mathématique et informatique, les réseaux complexes sont des structures des graphes décrivant des systèmes larges et complexes du monde réel. Un graphe est un outil mathématique permettant de représenter de manière synthétique des objets (les noeuds) et leurs relations (les arêtes). D'après Gondran et Minoux (1984), la théorie des graphes aurait été fondée en 1736 par Euler, qui a posé le premier problème de cheminement dans son *Solutio problematis ad geometriam situs pertinentis*. Des noms illustres y sont attachés, tels Cauchy (1813) et Poincaré (1900).

On considère un graphe $G = (V, E)$ où V est l'ensemble des sommets et $E \subseteq V \times V$ est l'ensemble des arêtes. On s'intéressera dans ce document, sauf indication contraire, aux graphes non orientés, lesquels décrivent une relation symétrique entre les sommets. Les sommets sont les objets, au sens général du terme, qui sont en relation dans le graphe. On notera N l'ordre (nombre de sommets) de G, avec $N = |V|$. Deux sommets v_i et v_j sont adjacents s'ils sont les extrémités d'une même arête du graphe, c'est-à-dire si $(v_i, v_j) \in E$. Les arêtes décrivent les relations entre les sommets du graphe.

La notation ci-dessus peut être facilement étendue pour les graphes qui ont des couches (en anglais Multilayer). Tout d'abord, le concept d'aspect peut être défini comme une caractéristique d'une couche représentant une dimension de la structure de la couche (par exemple, le type d'une arête, l'heure à laquelle une arête est présente. Ensuite, un réseau

multi-couches est défini comme un quadruple :

$$M = (V_M, E_M, V, L) \tag{2.1}$$

tel que V est l'ensemble des noeuds du réseau. Notez que les noeuds ne doivent pas nécessairement être homogènes, c'est-à-dire qu'ils peuvent représenter différentes entités. $V_M \subseteq V \times L_1 \times ... \times L_d$ est l'ensemble des combinaisons noeud-couche, c'est-à-dire l'ensemble des couches dans lesquelles un noeud $v \in V$ est présent. $E_M \subseteq V_M \times V_M$ est l'ensemble des arêtes contenant l'ensemble des paires de combinaisons possibles de noeuds et de couches élémentaires. $L = \{L_a\}_{a=1}^{d}$ est l'ensemble des couches élémentaires définies par un nombre d'aspects d tels qu'il existe un ensemble de couches élémentaires L_a pour chaque aspect a. Notez que si d = 0, le graphe multi-couches M se réduit à un graphe mono-couche.

Nous rappelons dans la suite quelques définitions de base.

Un graphe $G = (V, E)$ est ***non dirigé*** si $(v_i, v_j) = (v_j, v_i)\,,\ \forall v_i,\ v_j \in V$ tel que (v_i, v_j) dénote l'arête entre le noeud v_i et v_j.

Un graphe $G = (V, E)$ est ***dirigé*** si $(v_i, v_j) \neq (v_j, v_i)\,,\ \forall v_i,\ v_j \in V$ tel que (v_i, v_j) dénote l'arête entre le noeud v_i et v_j.

Un graphe $G = (V, E)$ est ***simple*** s'il n'admet aucune boucle ni plusieurs arêtes entre une même paire de noeuds.

Un graphe ***multiple*** est un graphe qui n'est pas simple et qui présente plusieurs arêtes entre une paire de noeuds.

Un graphe $G = (V, E)$ est ***pondéré*** si un poids est associé à chaque arête du graphe.

Une ***matrice d'adjacence :*** l'interconnexion entre les noeuds au sein d'un graphe peut être décrite par une matrice d'adjacence A_G de dimension $n \times n$. Pour un graphe non pondéré nous avons $A_{i,j} \in \{0, 1\}; i, j = 0, ..., n$.

Un ***sous graphe*** $G_X = (X, E_X)$ du graphe $G = (V, E)$ est un graphe tel que $X \subseteq V$ et $E_X \subseteq E$.

Un ***sous graphe induit*** $G_X = (X, E_X)$ est un sous graphe de $G = (V, E)$ induit par un ensemble de noeuds tel que $X \subseteq V$ et $E_X = \{(u, v) \in E, u \in X, v \in X\}$.

Un ***graphe complet :*** un graphe est complet si tous les noeuds sont deux à deux connectés entre eux, tel que $|E| = \frac{1}{2}|N|(|N|-1)$ dans un graphe non dirigé ou $|E| = |N|^2$ dans un graphe dirigé.

Une ***k-clique*** X est un ensemble de noeuds dans le graphe $G = (V, E)$ tel que le sous graphe induit $G_X = (X, E_X)$ forme un graphe complet avec un ordre k (tel que $|X| = \frac{k(k-1)}{2}$ dans un graphe non dirigé)

Un ***graphe biparti :*** un graphe dont l'ensemble des sommets est divisé en deux sous-ensembles disjoints V_1 et V_2 et tel que chaque arête connecte un sommet de V_1 à un sommet de V_2.

Un ***graphe assortatif :*** est un graphe dont la corrélation entre les degrés de ses sommets et ceux de leurs voisins est élevée. Autrement dit, les sommets ayant un degré élevé (respectivement un degré faible) sont connectés entre eux. Dans le cas contraire, le graphe est disassortatif.

Un ***graphe connexe :*** un graphe est connexe s'il existe un chemin entre tout couple de noeuds.

Une ***composante connexe :*** une composante connexe d'un graphe est un sous-graphe connexe maximal.

Un ***chemin :*** un chemin est une suite $(v_1, ..., v_k)$ de noeud de G tels que deux noeuds consécutifs quelconques v_i et v_{i+1} sont connectés par un lien : $\forall i, 0 \leq i \leq k-2, (v_i, v_{i+1}) \in E$. La longueur du chemin correspond au nombre de liens parcourus k.

Le ***plus court chemin moyen :*** est un concept de topologie de réseau défini comme le nombre moyen des pas sur les chemins les plus courts pour toutes les paires de noeuds de graphe possibles. C'est une mesure de l'efficacité de l'information ou du transport de masse sur un réseau.

Une ***distance géodésique :*** la distance géodésique $dist_G(u, v)$ entre deux noeuds u et v de G est la longueur du plus court chemin entre eux.

Un ***diamètre :*** le diamètre D_G d'un graphe G est le plus long des plus courts chemins du G.

Une ***densité :*** la densité d_G d'un graphe est le rapport entre le nombre de liens divisés

par le nombre de liens possibles :

$$d_G = \frac{2m}{n \times (n-1)} \tag{2.2}$$

Un ***cycle*** dans un graphe $G = (V, E)$ est un chemin partant du noeud v_1 et se terminant au noeud v_1 dans un graphe tel que $(v_1, e_1, \ldots, v_1)$. Un graphe ***acyclique*** est un graphe qui ne contient aucun cycle.

Un ***arbre*** $T = (V, E)$ est un graphe connexe acyclique. Dans un arbre, il existe exactement un seul chemin entre deux noeuds. Une arête dans un arbre est appelée ***branche*** et chaque noeud v_i avec un degré égale à 1 est appelé ***feuille***.

Un ***coefficient de clustering :*** le coefficient de clustering d'un noeud cc(v) est la probabilité que deux noeuds soient liés, s'ils ont au moins un voisin commun. Le coefficient de clustering d'un noeud est donné par la formule suivante :

$$cc(v) = \frac{\textit{le nombre de triangles}}{\textit{le nombre de triades}} \tag{2.3}$$

Le coefficient de clustering de tout le graphe cc_G correspond à la moyenne des valeurs locales :

$$cc_G = \frac{1}{n} \sum_{i=1}^{n} cc(v_i) \tag{2.4}$$

où b est le nombre de noeuds de G, $cc_G = 1$ si G est un graphe complet.

2.2.1 Mesures de centralité

Les chercheurs ont proposé plusieurs définitions pour quantifier la notion d'importance d'un noeud dans un graphe, les plus connues sont sous le nom de mesures de centralité (Koschützki *et al.*, 2005). Une mesure de centralité est une fonction qui attribue à chaque noeud une valeur positive indiquant à quel point il est "central". La signification de la centralité dépend de la mesure utilisée. Il existe trois catégorie de mesures de centralité :

- Centralité locale : elle calcule l'importance d'un noeud en se basant uniquement sur sa connexion avec les voisins directs (ex. centralité de degré).

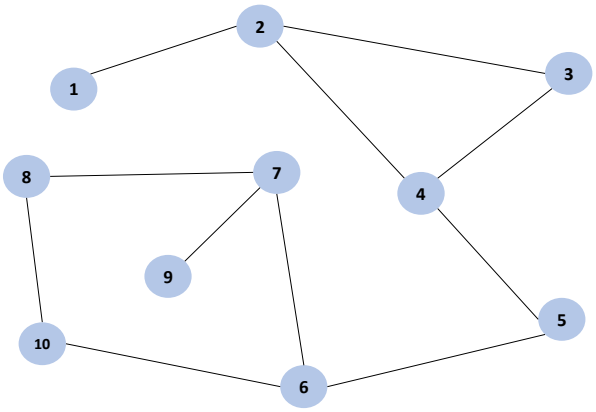

FIGURE 2.1 – Exemple d'un graphe pour l'illustration des mesures de centralités

- Centralité semi-locale : dans cette catégorie, l'importance d'un noeud ne dépend qu'une partie du graphe (ex. le travail de (Chen *et al.*, 2012)).

- Centralité globale : ce sont des centralités qui quantifient l'importance d'un noeud par rapport à tout le graphe. Trois principales mesures sont été réalisées pour cette mesure : la centralité de proximité, la centralité d'intermédiaire et la centralité de vecteurs propres.

Par la suite nous présentons les différentes mesures de centralité globales, et nous calculons leurs valeurs sur l'exemple du graphe présenté dans la figure 2.1.

Soit G un graphe non-dirigé non-pondéré et n le nombre de noeuds (Figure 2.1) :

2.2.1.1 Centralité de degré $C_d(v)$

Elle est considérée comme la notion la plus simple et la plus intuitive de la notion de centralité. L'idée de base est que l'importance d'un noeud au sein d'un graphe dépend du nombre de noeuds avec lesquels il est connecté directement. Le calcul de cette mesure se fait d'une manière locale, elle est définie comme la fraction des noeuds incidents au noeud v :

$$C_d(v) = d(v) \tag{2.5}$$

Centralité de degré										
v	1	2	3	4	5	6	7	8	9	10
Cd(v)	1	3	2	3	2	3	3	1	2	2
Rang	9	1	5	1	5	1	1	9	5	5

TABLEAU 2.1 – Résultats de la centralité de degré du graphe (Figure 2.1).

la centralité de degré permet de classer les noeuds d'une manière simple et efficace avec la complexité de $\mathcal{O}(n)$, cependant, elle devient moins pertinente dans le cas où un noeud avec peu de voisins importants a un degré d'importance plus élevé que celui d'un noeud ayant plusieurs voisins importants. Le tableau 2.1 montre les valeurs de C_d pour tous les noeuds du graphe (Figure 2.1).

2.2.1.2 *Centralité de proximité* $C_c(v)$

La centralité de proximité *(en anglais closeness)* esst une mesure de centralité globale basée sur l'intuition de la position stratégique d'un noeud dans un graphe, et s'il est voisin des autres noeuds. Si on prend l'exemple d'un réseau social, cette mesure correspond à l'idée qu'un acteur est central s'il peut contacter facilement un grand nombre d'acteurs avec un minimum d'efforts (l'effort ici est relatif à la longueur des chemins). Formellement, la centralité de proximité est l'inverse de la moyenne des distances géodésique (i.e. taille du chemin le plus court) vers tous les autres noeuds :

$$C_c(v) = \frac{n-1}{\sum_{v_i \in E, v_i \neq v_j} d_g(v_i, v_j)} \tag{2.6}$$

où $d_g(v_i, v_j)$ est la distance géodésique entre deux noeuds v_i et v_j. La complexité de calcul de cette centralité est : $\mathcal{O}(nlog(n) + m)$, où m est le nombre de liens (Okamoto *et al.*, 2008). Plus que la valeur de la centralité est élevée, plus que la probabilité que le noeud soit proche de l'ensemble des autres noeuds est forte et plus il est central. Le tableau 2.2 montre les scores des noeuds du graphe (Figure 2.1) en terme de C_c.

Centralité de proximité										
v	1	2	3	4	5	6	7	8	9	10
Cc(v)	1/34	1/26	1/27	1/21	1/19	1/19	1/23	1/31	1/29	1/25
Rang	10	6	7	3	1	1	4	9	8	5

TABLEAU 2.2 – Résultats de la centralité de proximité du graphe de l'exemple.

Centralité d'intermédiarité										
v	1	2	3	4	5	6	7	8	9	10
Cb(v)	0	8	0	18	20	21	11	0	1	6
Rang	8	5	8	3	2	1	4	8	7	6

TABLEAU 2.3 – Résultats de la centralité d'intermédiarité du graphe de l'exemple.

2.2.1.3 *Centralité d'intermédiarité* $C_b(v)$

La centralité d'intermédiarité *(en anglais betweenness)* est une mesure de centralité globale. Elle mesure l'utilité d'un noeud dans la transmission de l'information au sein d'un réseau. Un noeud peut être considéré central s'il se retrouve sur beaucoup de plus courts chemins entre d'autres paires de noeuds. Elle est définie formellement par :

$$C_b(v) = \sum_{s,t \in V} \frac{\sum(s,t|v)}{\sum(s,t)} \tag{2.7}$$

où $\sum(s,t)$ est le nombre des plus courts chemins liant s à t, et $\sum(s,t|v)$ est le nombre de ces chemins passant par v. La complexité de cette centralité est $\mathcal{O}(n.m + (n)^2 log(n))$ (Brandes, 2001). Le tableau 2.3 montre la centralité d'intermédiarité des noeuds du graphe (Figure 2.1) avec leur classement.

2.2.1.4 *Centralité des vecteurs propres* $C_{ev}(v)$

La centralité des vecteurs propres *(en anglais Eigenvector)* est nommée aussi centralité spectrale. L'idée est que la centralité d'un noeud dépend de la centralité des noeuds voisins. Il s'agit d'une extension de la centralité de degré dans laquelle on ne donne pas le même poids aux noeuds voisins. Formellement, la centralité spectrale d'un noeud est considérée

Centralité de vecteurs propres										
v	1	2	3	4	5	6	7	8	9	10
Cev(v)	0.171	0.413	0.363	0.463	0.342	0.363	0.292	0.121	0.221	0.242
Rang	9	2	3	1	5	3	6	10	8	7

TABLEAU 2.4 – Résultats de la centralité de vecteurs propres du graphe de l'exemple.

comme étant dépendante de la combinaison linéaire des centralités des noeuds voisins :

$$x_v = \frac{1}{\lambda} \sum_{t \in \Gamma(v)} x_t = \frac{1}{\lambda} \sum_{t \in V} a_{v,t} x_t \tag{2.8}$$

ici, λ est un réel strictement positif. L'équation 2.8 peut être réécrite sous forme vectorielle comme suit : $x = \frac{1}{\lambda} Ax$ qui est équivalent à $\lambda x = Ax$ avec $x = \{x_{v_1}, x_{v_2}, .., x_{v_n}\}$ est le vecteur de centralité du vecteur propre de tous les noeuds. Il existe en général, plusieurs valeurs propres pour lesquelles une solution du vecteur propre existe. Cependant, l'exigence que toutes les entrées du vecteur propre soient positives, implique que seule la valeur propre la plus élevée soit retenue, qui est λ. La complexité de la centralité des vecteurs propres est $\mathcal{O}(n^2)$. Dans le reste de ce document, nous notons cette centralité par C_{ev}. Le PageRank de Google est considéré comme une variante de cette centralité. Les résultats de cette centralité sur le graphe de l'exemple sont présentés dans le tableau 2.4.

2.2.1.5 Centralité de degré des voisins $C_l(v)$

C'est une centralité de semi-locale qui a été proposée par Chen *et al.* (2012). L'idée est que l'importance d'un noeud dépend de l'importance des noeuds voisins. Au contraire de la centralité des vecteurs propres, le calcul de la centralité n'est pas itératif. Il suffit que chaque noeud calcule la somme de degré des voisins. Cette centralité est formulée comme suit :

$$C_l(v) = \sum_{t \in \Gamma(v)} d(t) \tag{2.9}$$

sa complexité de l'ordre de $\mathcal{O}(n(k^2))$ où k est le degré moyen du réseau. Nous donnons les valeurs de cette centralité ainsi que le classement des noeuds dans le tableau 2.5.

Centralité de degré des voisins										
v	1	2	3	4	5	6	7	8	9	10
Cl(v)	3	6	6	7	6	7	6	3	5	5
Rang	4	2	2	1	2	1	2	4	3	3

TABLEAU 2.5 – Résultats de la centralité de degré des voisins du graphe de l'exemple.

2.3 Structure communautaire

La variété de l'origine des différents réseaux complexes ne les a pas empêchés d'avoir des structures similaires. Différentes études ont montré que les graphes de terrain exhibent des propriétés structurelles communes et non triviales (Albert *et al.*, 1999; Faloutsos *et al.*, 1999). Parmi les propriétés communes des graphes, on trouve également qu'ils sont souvent composés de sous-graphes denses faiblement interconnectés, appelés communautés (Girvan et Newman, 2002). Donner une définition formelle d'une communauté n'est pas une tâche évidente. Toutes les définitions proposées sont restrictives (Coscia *et al.*, 2011). Néanmoins, la définition la plus adoptée est celle liée à la topologie du réseau, qui considère qu'une communauté est un sous-graphe dont les noeuds sont densément inter-connectés et faiblement connectés au reste du réseau (Girvan et Newman, 2002). Prenons par exemple un groupe d'amis dans un réseau social, un ensemble de protéine qui a la même fonction biologique ou un ensemble de pages web traitant des sujets ayant la même thématique. La figure 2.2 présente un exemple de graphe avec trois communautés. Il peut y avoir aussi des communautés chevauchantes où chaque noeud peut appartenir à plusieurs communautés en même temps. Dans cette thèse, nous nous limitons à l'étude de la détection de communautés disjointes.

Connaître la structure communautaire d'un réseau aide non seulement à mener plusieurs applications cibles mais aussi aider à la réalisation des traitements complexes. Nous décrivons dans ce qui suit quelques exemples de ces applications

Parallélisme des calculs : avec l'explosion du volume des données ces dernières années, la structure communautaire sert à réduire la complexité de calcul de certaines opérations sur des grands graphes de terrain. En effet, le partitionnement d'un graphe en communautés permet d'effectuer des calculs séparés moins coûteux sur chaque commu-

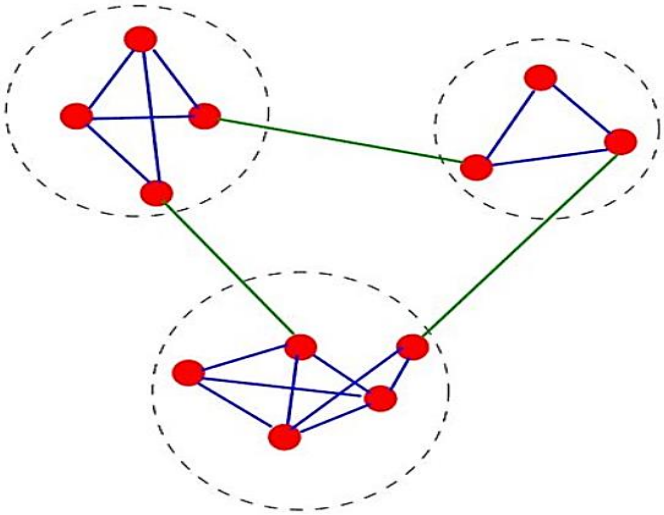

FIGURE 2.2 – Exemple d'un graphe avec trois communautés entourées par des cercles pointillés.

nauté avant d'agréger les résultats.

Compréhension du réseau : La grande taille des graphes complexifie la tâche de leur compréhension. Par exemple Facebook contient plus de 900 millions d'utilisateurs, Google indexe plus d'un trillion d'URLs, le nombre de clients dans le réseau téléphonie mobile dans un opérateur est environ 200 millions. L'identification des communautés permet de réduire cette complexité et de découvrir des relations entre les entités du réseau. Un travail dans ce contexte a été proposé par Blondel *et al.* (2008). Ils proposent d'étudier les relations entre les clients d'un opérateur Belge dans un réseau téléphonique. Le réseau a été modélisé par un graphe de 2.6 millions noeuds représentant les utilisateurs. Les liens entre les neouds sont pondérés par la durée cumulative dans ce réseau. La détection des communautés a produit 2 groupes principaux qui correspondent aux deux communautés francophone et flamande de la population belge (voir la figure 2.3).

Visualisation : La visualisation constitue une aide précieuse pour la compréhension et l'analyse des réseaux. Néanmoins, les outils de visualisation actuels ne permettent pas de traiter des graphes de terrain à cause de leur taille. La visualisation pourrait se faire facilement au niveau macroscopique à l'aide des communautés. La visualisation via les communautés aide donc à réduire la complexité du graphe d'une manière à qu'il soit in-

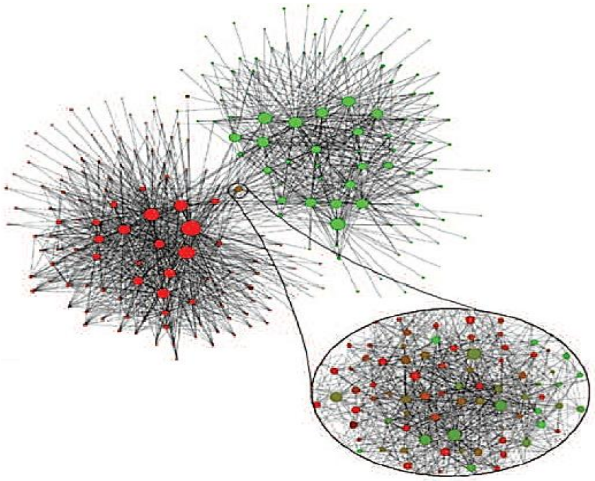

FIGURE 2.3 – Structure communautaire d'un réseau social de communication téléphonique (Blondel *et al.*, 2008). Les points colorés indiquent les sous-communautés au niveau hiérarchique. La coloration du rouge au vert représente la fraction des langues parlées dans chaque communauté (rouge pour les francophones et vert pour les flamands). Les deux grandes communautés sont linguistiquement homogènes, avec plus de 85% des personnes qui parlent la même langue. La communauté qui se trouve entre les deux graphes (partie zoomée) possède une répartition de langues équilibrée.

terprétable par l'oeil humain, par exemple la segmentation des images par la modélisation par graphes, permet de détecter les communautés dans l'image qui correspondent à des objets.

Identifier les communautés permet d'avoir une vue mésoscopique du réseau complexe et aide à comprendre sa structure. Cela aide également à mener des opérations plus complexes sur les réseaux comme la visualisation, la compression, *etc.* Beaucoup d'autres applications peuvent se servir de la tâche de détection de communautés. Parmi les principales méthodes de détection de communautés proposées dans la littérature, on peut citer celles qui optimisent une fonction de qualité pour évaluer la qualité d'une partition donnée, comme la modularité, la stabilité, la coupe ratio, la coupe min-max ou la coupe normalisée (Shi et Malik, 2000; Ding *et al.*, 2001; Newman et Girvan, 2004), les techniques hiérarchiques comme les algorithmes de division (Newman et Girvan, 2004), les méthodes spectrales (Von Luxburg, 2007) ou l'algorithme de Markov et ses extensions (Satuluri et Parthasarathy, 2009). Ces techniques de partitionnement de graphes sont très utiles pour détecter des composantes fortement connectées dans un graphe. Nous présenterons

dans cette section les méthodes importantes du domaine en mettant l'accent sur celles qui optimisent un critère de qualité de la partition, notamment la modularité et la stabilité puisque nos contributions utilisent ces critères.

2.4 Les approches de détection de communautés

Soit un graphe G=(V,E), on cherche la partition $P = \{c_1, ..., c_p\}$ de G en p communautés tel que $\cup_{i=1}^{p} c_i = V$; $\forall c_i \in P$, $c_i \neq \emptyset$; et $\forall i, j$; $i \neq j \Rightarrow c_i \cap c_j = \emptyset$.

Différents méthodes pour la détection des communautés ont été proposées, on trouve des techniques basées sur la marche aléatoire, méthodes basées sur la diffusion, des méthodes spectrales, *etc.* (Pour plus de détails, voir (Fortunato, 2010; Plantié et Crampes, 2013)). Ces méthodes s'appuient uniquement sur les données structurelles des réseaux sans considérer tous les autres attributs particuliers aux noeuds. Depuis le travail fondateur de (Girvan et Newman, 2002), la majorité des approches consistent à trouver une partition des noeuds du graphes tout en optimisant un critère de qualité d'un partitionnement, défini à partir de la structure du graphe. Les critères les plus connus et utilisés sont la modularité et la stabilité. Nous décrivons par la suite le principe de chaque critère.

2.4.1 Modularité : qualité d'une partition

La modularité est basée sur le fait que la structure communautaire n'existe pas dans les réseaux aléatoires. Ainsi, pour une partition trouvée, on veut non seulement que la valeur de la fraction de liens qui se trouve à l'intérieur d'une communauté c soit élevée, mais aussi que pour la même partition sur un graphe aléatoire produit une faible valeur de cette fraction. La fraction des arêtes au sein des communautés dans un réseau aléatoire se calcule à partir du degré des noeuds comme suivant : pour une partition donnée P, si une arête choisie au hasard, la probabilité a_c qu'une de ses extrémités de celle-ci mène à la communauté c, est égale au nombre des arêtes ayant une extrémité dans la communauté c, divisé par le nombre totale d'arêtes du réseau m. La probabilité qu'une arête soit connectée à un noeud dans la communauté c'est la proportion de demi-arêtes dans c, soit la somme des degrés des noeuds de c divisée par le multiple de nombre de liens : $a_c = \frac{\sum_{i \in c} d(n_i)}{m}$. La

probabilité que les deux extrémités d'un lien soient dans la communauté c'est donc a_c^2. Ainsi, la modularité est définie par l'équation 2.10 :

$$Q(P) = \sum_{c \in P} (e_c - a_c^2) \tag{2.10}$$

D'une autre manière, la modularité est exprimée comme suit : on considère que A soit la matrice d'adjacence d'un graphe G, tel que les éléments A_{ij} représentent les poids des arêtes entre le pair des noeuds i et j, et valent 0 ou 1 dans le cas d'un graphe non-pondéré. L'équation 2.11 peut alors être reformulée par :

$$Q(P) = \frac{1}{2m} \sum_{c \in P} \sum_{i,j \in c} (A_{ij} - \frac{d(i)d(j)}{2m}) \tag{2.11}$$

La modularité est comprise entre -1 et 1. Une bonne modularité a une valeur positive et la qualité de la partition augmente avec la modularité. Bien que plusieurs fonctions de qualité aient été proposées pour évaluer la qualité d'une partition d'un graphe donné (Mancoridis *et al.*, 1998), la modularité reste la fonction la plus utilisée.

Prenons en compte la taille de certains réseaux du monde réel, beaucoup d'efforts sont déployés pour trouver des algorithmes efficaces capables de traiter des réseaux de plus en plus grands, tels que les méthodes d'optimisation de la modularité. Cependant, il a été démontré que les réseaux ont souvent plusieurs niveaux d'organisation (Simon, 1991), conduisant à différentes partitions pour chaque niveau que l'optimisation de la modularité ne peut pas les traiter. Des méthodes ont été fournies pour adapter l'optimisation de la modularité à l'analyse multi-échelle (multi-résolution) à l'aide d'un paramètre de réglage (Reichardt et Bornholdt, 2006; Arenas *et al.*, 2008). Pourtant, la recherche d'une fonction de qualité de partition qui reconnaît la nature multi-résolution de réseaux fondés sur des bases théoriques appropriées a retenu moins l'attention. Récemment, la stabilité (Delvenne *et al.*, 2010) a été introduite en tant qu'une nouvelle mesure de qualité pour les partitions communautaires. Nous étudions dans la prochaine section son utilisation comme critère d'optimisation pour l'analyse multi-échelle. Nous montrons comment la stabilité peut être utilisée au lieu de la modularité dans les méthodes d'optimisation de la modularité, Nous présentons ensuite les algorithmes d'agglomération glouton utilisant la stabilité comme

fonction d'optimisation qui permettent une analyse multi-échelle, en utilisant le temps de Markov comme un paramètre de résolution.

2.4.2 Stabilité : qualité d'une partition

Une grande attention a été accordée à la modularité mais peu d'attention a été accordée aux nouvelles mesures de la qualité des partitions. Récemment, la stabilité a été introduite dans (Delvenne *et al.*, 2010) en tant qu'une nouvelle mesure de la qualité d'une partition unifiant certaines heuristiques de clustering connues, et en utilisant la modularité et le temps de Markov comme paramètre de résolution interne. La stabilité d'un graphe considère le graphe comme une chaîne de Markov où chaque noeud représente un état et chaque lien une transition d'état possible. Soit le n le nombre de noeuds, m le nombre de liens, $A(n \times n)$ est la matrice d'adjacence qui contient les poids des liens (le graphe peut être pondéré ou non), d est un vecteur de taille n donnant pour chaque noeud son degré et $D = diag(d)$ correspond à la matrice diagonale. La stabilité d'un graphe considère le graphe comme une chaîne de Markov où chaque noeud représente un état et chaque lien une transition d'état possible. La distribution en chaîne est donnée par la distribution stationnaire $\pi = \frac{d}{2m}$. Soit $\prod$ sa matrice diagonale correspondante $\prod = diag(\pi)$. La transition entre les états est donnée par la matrice stochastique $M = D^{-1}A$ $(n \times n)$. Supposons qu'on a une partition des communautés, soit H la matrice indicateur de taille $n \times c$ qui associe chaque noeud à sa communauté correspondante. La matrice d'auto-covariance à un temps de Markov t est définie par l'équation 2.12

$$R_t = H^T(\prod M^t - \pi^T\pi)H \tag{2.12}$$

La stabilité à un certain temps t, notée par Q_{S_t} est donné par la trace de R_t. La mesure de stabilité globale Q_S considère la valeur minimale du Q_{S_t} dans le temps à partir du temps 0 jusqu'à une limite supérieure donnée t :

$$Q_s = \min_{0 \leq t \leq \tau} trace(R_t) \tag{2.13}$$

Ce modèle peut être étendu pour traiter les valeurs réelles de t en utilisant l'interpo-

lation linéaire :

$$R_t = (c(t) - t).R(f(t)) + (t - f(t)).R(c(t)) \tag{2.14}$$

tel que c(t) retourne le plus petit entier supérieur à t et f(t) renvoie le plus grand entier inférieur à t. Ceci est utile pour rechercher par exemple des valeurs de temps comprises entre 0 et 1. En effet Il a été démontré dans (Delvenne *et al.*, 2010) que l'utilisation du temps de Markov avec des valeurs comprises entre 0 et 1 permet de détecter des partitions plus fines que celles détectées au temps 1 ou plus.

2.4.3 Détection de communautés par maximisation de modualrité

la maximisation de la modularité est un problème NP-difficile (Brandes *et al.*, 2007). Plusieurs méthodes ont été proposées pour la calculer, certaines utilisent les techniques de l'algorithmique génétique (Li et Song, 2013; Pizzuti, 2012; Cai *et al.*, 2011), du recuit-simulé (Reichardt et Bornholdt, 2006; Guimera *et al.*, 2004), d'autres utilisent l'optimisation extrémale (Duch et Arenas, 2005). Cependant, les heuristiques les plus appliquées sont fondées sur le principe de la classification hiérarchique. Deux approches contradictoires sont largement expérimentées : 1) Les approches agglomératives (ou ascendantes) selon lesquelles on part d'un ensemble des singletons, puis à chaque itération, on fusionne deux communautés qui promettent une modularité maximale. 2) Les approches séparatrices (ou descendantes) dans lesquelles, on commence au début par un graphe entier, puis on cherche à scinder une communauté en deux a chaque itération, de telle sorte que la modularité soit maximisée.

Dans les deux cas, l'algorithme produit une hiérarchie de communautés. On retient généralement la partition qui a la modularité maximale. Nous décrivons ci-dessous quelques algorithmes de ces deux familles sans exhaustivité.

2.4.3.1 Algorithme de Girvan-Newman(GN)

C'est une méthode divisive proposée par (Girvan et Newman, 2002). Elle s'appuie sur la mesure de centralité d'intermédiaire des liens comme une heuristique de sélection des liens à supprimer. Étant donné un lien (u, v), cette centralité est calculée par l'équation

2.15 :

$$C_b((u,v)) = \sum_{i,j \in n, i \neq j} \frac{\sigma_{ij}((u,v))}{\sigma ij} \tag{2.15}$$

où $\sigma_{ij}((u,v))$ représente le nombre de plus courts chemins entre v_i et v_j passant par l'arête (u,v) et σ_{ij} le nombre total de plus courts chemins entre v_i et v_j. L'objectif de cette centralité d'intermédiarité est de repérer les arêtes centrales du graphe qui se ressemblent à des ponts connectant les communautés entre elles. Cela est naturellement vrai puisqu'une arête inter-communautaire serait traversée par une fraction élevée de plus courts chemins entre les noeuds appartenant à différentes communautés. Considérons un graphe G, l'algorithme itère m fois en coupant à chaque fois l'arête qui a le maximum d'intermédiarité. Cela permet de construire une hiérarchie de communautés dont la racine est l'ensemble du graphe et les feuilles sont les communautés composées de noeuds isolés. La partition qui a la modularité la plus élevée est retenue. Cet algorithme nécessite un calcul itératif de la centralité d'intermédiarité ce qui le rend coûteux en temps avec une complexité $\mathcal{O}(m^2 n)$. Il est donc difficile de l'appliquer sur les grands graphes.

2.4.3.2 Méthode de Louvain

C'est une méthode agglomérative proposée par Blondel *et al.* (2008). Elle implante une méthode d'optimisation gloutonne locale de la modularité. A l'état initial, chaque noeud est affecté à une communauté différente des autres. La méthode applique ensuite une itération de successions de deux phases :

1. Phase d'affectation des noeuds : pour chaque déplacement d'un noeud x vers la communauté des voisins directs, on évalue le gain de la modularité s'il est maximisé. x reste dans sa communauté si aucun gain n'est pas trouvé.

2. Phase de compression : Pour compresser le graphe obtenu on remplace chaque communauté par un seul noeud. Un lien est considéré entre deux noeuds c_x, c_y dans le nouveau graphe, s'il existe un lien entre un noeud de la communauté représentée par c_x et un noeud de la communauté représenté par c_y. Finalement, la somme des

poids des liens reliant des noeuds de deux communautés est considérée comme le poids de liens entre deux communautés.

La méthode de Louvain s'arrête s'il y a plus de possibilités de réaffectation de noeuds ou si un maximum de modularité reste stable. Théoriquement la complexité de cette approche n'est pas encore étudiée, mais expérimentalement, elle est évaluée à $\mathcal{O}(nlog\ \mathrm{n})$, ce qui explique sa rapidité pour la détection des communautés.

2.4.3.3 Méthode Spectrale

C'est une méthode séparatrice à base des vecteurs propres (en anglais Eigen vectors (EV)) (Newman, 2006). Dans ce travail, la modularité est exprimée en fonction des vecteurs propres du graphe, et elle devient "la matrice de modularité". Newman (2006) propose une méthode spectrale pour trouver la partition du graphe. Cette méthode commence par effectuer une décomposition spectrale de la matrice de modularité, puis elle répète cette étape pour chaque sous-graphe. Quand il n'y a aucun gain de la modularité, la méthode EV quitte le sous-graphe indivisible correspondant. Cette méthode se termine lorsque l'ensemble du réseau est décomposé en des sous-graphes indivisibles. La complexité de cette méthode est de l'ordre $\mathcal{O}(n^2log\ \mathrm{n})$.

2.4.3.4 Limites de la modularité

Les méthodes basées sur l'optimisation de la modularité font implicitement les hypothèses du travail suivantes :- Le meilleur partitionnement d'un graphe en communautés est celui qui produit la modularité maximale ; - Si un graphe a une structure communautaire alors on peut trouver une décomposition pour laquelle la modularité est maximale ; - Pour un graphe à structure communautaire, les communautés correspondantes à des valeurs élevées de modularité sont structurellement similaires.

Or, des études récentes ont montré que La modularité a deux principales limites : une limite de résolution (Fortunato et Barthelemy, 2007) et une limite due au plateau, la première l'empêche de détecter des communautés de petite taille, tant que la deuxième est très irrégulière et qu'elle présente autour de son maximum global : il existe souvent de très nombreux maxima locaux, proches du maximum global, mais qui correspondent à

des communautés très différentes.

2.4.4 L'intérêt d'une vision multi-échelle

Pour parer à la première limite évoquée (celle de résolution), il est souvent nécessaire de faire appel à des algorithmes qui vont donner plusieurs solutions de partition. Une manière de procéder est de définir une notion d'échelle dans le graphe, et pour chaque échelle donnée, il faut chercher la partition qui découpe le mieux le graphe en communautés. Par cette manière, on s'assure de ne pas passer à côté de petites communautés non détectables autrement. Pour remédier à la seconde limite, il est nécessaire d'associer à une méthode multi-échelle une mesure de stabilité qui va permettre à l'utilisateur de choisir les échelles spécifiques pour lesquelles les partitions sont pertinentes.

Nous posons à présent le problème de détection multi-échelle de communautés, qui se décompose comme suit, pour la majorité des travaux existants (Fortunato, 2010) :

1. Définir une notion d'échelle pertinente. Cette échelle, qu'on notera de manière générique s, est continue et définie sur un intervalle réel.

2. Discrétiser s en M échelles discrètes $S = \{s_i\}_{i\in[1,M]}$.

3. Proposer une mesure de qualité dépendante de l'échelle et optimiser cette mesure pour chaque échelle s_i, afin d'obtenir la partition en communautés à cette échelle P_{s_i}. On obtient ainsi l'ensemble de partitions $\mathbb{P} = \{P_{s_i}\}_{s_i\in S}$.

4. Pour les M échelles proposées, détecter lesquelles sont vraiment signifiantes. Pour ce but, il est nécessaire d'avoir pour chaque échelle une mesure de pertinence.

2.4.5 Les méthodes multiéchelle pour la détection des communautés

2.4.5.1 La méthode de Reichardt et Bornholdt

Reichardt et Bornholdt (2006) proposent une approche multiéchelle en se basant sur une analogie entre la détection en communautés, et le modèle de Potts en physique statistique qui est basée sur les interactions entre spins. Nous présentons ici sa version initiale qui prend en compte que les matrices binaires. Le modèle de Potts est un cas général du

modèle d'Ising, pour chaque spin i (i.e. chaque noeud i), il associe une valeur de spin σ_i qui correspond dans l'analogie à la communauté du noeud i. Son principe exige que les spins adjacents aient des valeurs un peu similaires, alors que les spins non-adjacents aient des valeurs différentes. En d'autres termes, le modèle favorise (resp. pénalise) les arêtes connectant des spins de valeurs égales (resp. différentes) et les non-arêtes entre des spins de valeurs différentes (resp. égales). L'état physique d'un tel système est caractérisé par la liste $\{\sigma\}$ de tous les σ_i (i.e. une partition) , et son énergie vaut :

$$\mathcal{H}_\gamma(\{\sigma\}) = -\sum_{i<j}(A_{ij} - \gamma p_{ij})\delta(\sigma_i, \sigma_j), \tag{2.16}$$

où A_{ij} et l'élément (i,j) de la matrice d'adjacence, $\gamma \in \mathbb{R}^+$ et p_{ij} la probabilité d'avoir un lien entre les noeuds i et j pour un modèle nul au choix. Trouver l'état $\{\sigma\}$ qui minimise cette énergie revient à trouver les communautés dans le graphe. L'information du modèle aléatoire auquel on veut confronter l'existence de structures en communautés est donné par la probabilité p_{ij}. Par exemple, $p_{ij} = p$ si on veut confronter le modèle au modèle d'Erdös-Rényi. Mais, généralement pour les graphes complexes usuels, les auteurs proposent d'utiliser le modèle de Chung-Lu : $p_{ij} = d_i d_j / 2d_{tot}$ qui exige une distribution réaliste des degrés. Notons que pour le cas $\gamma = 1$ on se retrouve à un facteur multiplicatif près de l'expression de la modularité. De cette manière-là, $\mathcal{H}_\gamma(\{\sigma\})$ peut être considérée comme une modularité généralisée paramétrée par la donnée d'un modèle aléatoire de graphe et de γ. Le paramètre γ joue le rôle de paramètre d'échelle : on peut passer de la solution à une communauté contenant tous les noeuds pour $\gamma = 0$ à la solution à N communautés chacune contenant un seul noeud pour $\gamma \to \infty$. Une extension de ce modèle est proposée pour les graphes pondérés dans (Heimo *et al.*, 2008), et pour les graphes dirigés dans (Traag et Bruggeman, 2009).

2.4.5.2 Autres méthodes

Ronhovde et Nussinov (2010) proposent également une méthode multi-échelle proche de celle de Reichardt, qu'ils nomment "méthode de Potts absolue" parce qu'elle ne fait pas recours au modèle de graphe aléatoire comme la méthode de Reichardt. Ceci dit,

les deux méthodes restent conceptuellement similaires.

Le Martelot et Hankin (2011, 2013) qui ont proposé deux algorithmes optimisés pour la détection multiéchelle de communautés. En effet, ils considèrent que la détection d'une partition à une échelle $s + \gamma s$ peut être faite plus rapidement si on s'inspire du résultat obtenu à l'échelle s. Les auteurs proposent un algorithme qui permet d'optimiser les fonctions de qualité locales comme celles de Huang ou de Lancichinetti, et un autre algorithme pour optimiser les fonctions de qualité globales comme celles de Delvenne, de Pons, Reichardt, d'Arenas ou de Ronhovde.

2.4.6 Les approches alternatives

En étudiant les limites des approches fondées sur la maximisation de modularité, il s'avère qu'il est nécessaire d'adopter ou améliorer d'autres concepts pour mieux détecter les communautés. Différentes approches alternatives ont été proposées, parmi ces approches, il y a une catégorie de méthodes qui s'appuient sur un processus dynamique qui se déroule dans le réseau, pour révéler ses communautés. On trouve principalement deux méthodes :

2.4.6.1 InfoMap

C'est une méthode proposée par Rosvall et Bergstrom (2008). Comme Walktrap, InfoMap exploite le fait qu'un marcheur suivant aléatoirement les liens du graphe, a tendance à rester bloqué dans les communautés, Si on décrit un parcours aléatoire sur le graphe comme une séquence de numéros, qui peut être le numéro du noeud courant ou de la communauté courante, un bon partitionnement consiste à compresser au mieux cette séquence. La qualité du partitionnement représente la quantité d'informations utilisées pour le codage du graphe. Les auteurs optimisent ce critère en utilisant une méthode similaire à celle de Louvain (Blondel *et al.*, 2008). La complexité de cette méthode est de l'ordre $\mathcal{O}(nlog\ \mathrm{n})$.

2.4.6.2 LPA

L'algorithme de Propagation de Labels (LPA) est une approche basée sur la diffusion (Raghavan *et al.*, 2007) Elle s'adresse au problème de détection de communautés en utilisant un paradigme de communication qui s'appuie sur l'hypothèse que l'information est échangée de façon plus efficace entre les noeuds d'une même communauté. L'idée du LPA est simple : un label spécifique l_v est assigné à chaque noeud $v \in V$. Tous les noeuds mettent à jour de façon synchrone leurs labels en sélectionnant le label majoritaire chez les voisins directs. En cas où on a un choix multiple, un label est sélectionné aléatoirement L'algorithme itère jusqu'à ce qu'il atteigne un état stable où les noeuds ne modifient plus leurs labels. A la fin, les noeuds qui ont la même étiquette sont considérés comme une communauté. La complexité de chaque itération est de $\mathcal{O}(m)$, où m est le nombre de liens, si la propagation se fait de façon synchrone. Un avantage majeur de cette approche est son aspect parallélisme massif qui permet à cette méthode de passer à l'échelle. Cependant, il n'y a aucune preuve sur sa convergence. Aussi, en fonction de la topologie locale du réseau, certains noeuds peuvent avoir un problème d'oscillation entre les labels.

2.5 Conclusion

Dans ce chapitre, les notions des réseaux complexes ont été introduites. Les concepts de la théorie des graphes seront exploités dans les différentes contributions proposées dans cette thèse pour différentes applications. Les approches de détection de communautés ont été également définies avec les deux critères modularité et stabilité pour évaluer la qualité de découpage d'un graphe en communautés. Nous avons soulevé les deux limites principales dans l'optimisation de la modularité, pour ensuite citer les approches multi-échelles et les approches alternatives pour remédier à ce problème. Ainsi nous présenterons dans la partie suivante (c.f. chapitre 3), un état de l'art sur la segmentation des images, suivi d'une présentation de notre framework proposé basé sur les algorithmes de détection de communautés qui sont dédiés pour les réseaux complexes (c.f. chapitre 4).

Première partie

Segmentation des images

Chapitre

3 État de l'art

Sommaire

3.1 Introduction

La segmentation des images est un problème fondamental en vision par ordinateur. L'objectif de la segmentation d'images est de segmenter une image en plusieurs régions qui ne se chevauchent pas et qui sont jugées utiles en fonction d'un critère objectif. La segmentation des images est un problème étudié depuis longtemps. Depuis les premières approches de segmentation d'images publiées il y a plus de 40 ans, voir par exemple (Muerle, 1968), des milliers d'algorithmes ont été proposés, et ils peuvent être très différents en utilisant des modèles mathématiques ou des objectifs d'application différents. Le problème de la segmentation d'images peut être résolu par différentes approches. Le choix d'une approche de segmentation est lié à plusieurs facteurs : la nature de l'image, les conditions d'acquisition (bruit), les primitives à extraire (contours, textures,...) et bien évidemment les contraintes d'exploitation (fonctionnement en temps réel, type, mémoire vive et physique disponible). En effet dans la littérature, plusieurs approches de segmentation

d'images ont été proposées : a) Approche par contours ; b) Approche région ; c) Approche utilisant la théorie des graphes ; Approche basée sur les algorithmes de détection des communautés. Dans ce chapitre, on présente en premier temps les caractéristiques d'une image, puis un état de l'art sur les approches existantes pour la segmentation d'images, nous mettons l'accent surtout sur les approches fondées sur les graphes et les algorithmes de détection des communautés. Pour évaluer les performances des méthodes de segmentation, des bases de données des images ont été présentées dans ce chapitre ainsi que les métriques les plus utilisées dans l'évaluation des résultats de la segmentation.

3.2 Caractéristiques de la texture et la couleur

Deux propriétés basiques existent qui peuvent être considérées pour regrouper des pixels et définir le concept de similarité qui forme des régions : couleur et texture. L'importance des deux caractéristiques pour définir la perception visuelle est évidente dans les images correspondant à des scènes naturelles, qui présentent une grande variété de couleurs et de textures. Cependant, la plupart des méthodes de la littérature traite le problème de la segmentation soit en se basant sur la couleur, soit sur la texture, et peu de propositions considèrent les deux propriétés ensemble. Heureusement, cette tendance semble changée dans les faits, à l'origine de l'intuition voulant que l'utilisation des informations fournies par les deux fonctionnalités permette d'obtenir des résultats plus robustes et significatifs.

3.2.1 Couleur

L'humain sent intuitivement que la couleur est une caractéristique importante de leur expérience visuelle et qu'elle est utile, voire nécessaire, pour le traitement visuel dans le monde réel. Il est évident que l'oeil humain réagit plus rapidement et plus précisément à ce qui se passe dans une scène si elle est en couleur. La couleur est utile pour faire "ressortir" de nombreux objets quand ils seraient atténués ou même cachés dans une image en niveaux de gris.

Considérant que la couleur est sans aucun doute l'une des caractéristiques les plus intéressantes du monde naturel, la difficulté réside dans son traitement, qui peut être effectué

de différentes manières. Dans de nombreux cas, les composants RGB de base peuvent fournir des informations très précieuses sur l'environnement. Cependant, les modèles perceptuels, tels que CIE (L, a, b), YUV, ou HSV de (Maheswari et Korah, 2016) sont plus intuitifs et permettent donc d'extraire des caractéristiques selon le modèle de perception humaine. De plus, la complexité des scènes naturelles souligne la nécessité pour le système de sélectionner un bon espace colorimétrique extrêmement important pour les tâches de segmentation. Par conséquent, il est nécessaire de formuler la question suivante : quel est le meilleur espace colorimétrique à appliquer pour segmenter une image représentant une scène en extérieur. Notant que cette question n'a pas de solution unique ni parfaite. L'espace colorimétrique approprié pour un algorithme de segmentation ne convient pas pour d'autres. De ce fait, en raison de l'absence d'espace colorimétrique consolidé, nous traiterons le problème de la segmentation des couleurs en considérant l'utilisation d'un espace de couleur qui est proche du système visuel humain.

3.2.2 Texture

La répétition régulière d'un élément ou d'un motif sur une surface s'appelle texture. Il est utilisé pour identifier les différentes régions texturées et non texturées dans une image, pour les classer/segmenter et extraire les contours entre les principales régions de texture. La texture est un concept difficile à représenter. L'identification de textures spécifiques dans une image est obtenue principalement en modélisant la texture en tant qu'une variation bidimensionnelle du niveau de gris. La luminosité relative des paires de pixels est calculée de telle sorte que le degré de contraste, la régularité, la grossièreté et la directivité sont prises en considération.

Les descripteurs de texture représentent les caractéristiques d'une texture dans une image (Simon et Uma, 2018). La plupart des recherches sur la classification de la texture visent à trouver un descripteur de texture efficace, puissant et discriminant. Les descripteurs textuels extraient les caractéristiques et les représentent efficacement, ce qui permet d'obtenir des précisions de classification plus élevées, quel que soit le classificateur utilisé. La plupart des descripteurs de texture sont simples et sont basés sur des orientations, des arrangements spatiaux de pixels, une uniformité, un histogramme et des gradients. Dans

la littérature, les descripteurs de texture conventionnels tels que le motif binaire local (LBP), la matrice de co-occurrence de niveau de gris (GLCM), les textures de Law, les caractéristiques statiques, les modèles d'autocorrélation, *etc*, sont proposés.

3.3 Approches de segmentation

Le problème de la segmentation d'images peut être résolu par différentes approches. Dans la littérature plusieurs approches ont été proposées.

3.3.1 Approche contours

De façon générale, un contour est défini comme étant la frontière entre deux régions. La détection de contours consiste à repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse. Pratiquement, il s'agit de mettre en évidence les zones de transition et de détecter les différentes frontières qui séparent les régions dans une image. Les approches frontières elles même, peuvent être classées en plusieurs catégories. On peut distinguer normalement les modèles dérivatifs (Deriche, 1987), surfaciques (Haralick, 1987) et variationnelles (McInerney et Terzopoulos, 1996). Les méthodes de détection de contours donnent de bons résultats quand les contours de l'image sont bien définis. Cependant, dans le cas des images bruitées ou faiblement contrastées, les méthodes contours nécessitent une étape supplémentaire afin de fermer les bords des régions.

3.3.1.1 Calcul du gradient

Les contours dans une image sont caractérisés par une forte variation de contraste. La drivée (le gradient) est parmi les opérateurs qui permettent de caractériser les zones ou les niveaux de gris augmentent ou diminuent très vite. Il répond tout à fait à ce problème (Lechlek *et al.*, 2012). Le gradient d'une image permet de mesure les taux de changement de niveau de gris par unité de distance dans les directions des axes de coordonnés. On peut le définir comme un vecteur caractérisé par sa direction et son amplitude, tels que :

- La direction du gradient est orthogonale à la frontière qui passe au point considéré.

- L'amplitude est liée à la quantité de variation locale des pixels.

3.3.1.2 Approche de Canny

Canny (1987) a proposé un filtre déterminé analytiquement en se basant sur trois critères :

- Une bonne détection : l'opérateur donne une réponse au voisinage d'un contour.
- Une bonne localisation : optimisation de la précision avec laquelle le contour est détecté.
- Unicité de la réponse : le contour doit provoquer une réponse unique de l'opérateur.

Pour vérifier ces trois critères, Canny a proposé la solution suivante :

$$f(x) = a_1 e^{x/\sigma} sinwx + a_2 e^{x/\sigma} sinwx + a_3 e^{-x/\sigma} sinwx + a_4 e^{-x/\sigma} sinwx \qquad (3.1)$$

Où les coefficients a_i et w sont déterminés à partir de la taille du filtre. Le paramètre σ est un paramètre de grande importance, d'échelle qui indique en-deçà de quelle distance deux contours parallèles seront confondus en un seul. Canny montre que la dérivée d'une gaussienne est une bonne approximation de son filtre.

3.3.1.3 Algorithme de détection des contours et de segmentation d'images (EDISON) :

Ce système (Christoudias *et al.*, 2002) utilise l'algorithme "Meanshift" développé par Georgescu *et al.* (2003) et ses collègues (http ://www.caip.rutgers.edu/ riul/research/code/EDISON/). EDISON est un outil d'extraction des caractéristiques de bas niveau qui intègre la détection des contours basée sur la confidence et la segmentation d'images basée sur le "Meanshift". Il a été développé par le laboratoire Robust Image Understanding de l'Université Rutgers.

3.3.2 Approche régions

Les approches régions ont pour but de mettre en évidence les régions homogènes de l'image. Il s'agit de rechercher des ensembles de pixels partageant des propriétés com-

munes. Les régions sont différenciées entre elles par des propriétés élémentaires basées sur des critères locaux tels que le niveau de gris de chaque pixels, ou bien sur un attribut estimé dans le voisinage du pixel tel que la valeur moyenne, la variance ou des paramètres de texture. Les approches régions peuvent être classées en trois types de méthodes :

3.3.2.1 Méthodes de classification

Ces méthodes permettent de découper l'image en plusieurs classes. Chaque pixel est associé à une et une seule classe. Les méthodes de classification sont classées en :

1. Méthodes probabilistes, parmi celles-ci, on trouve les méthodes de mélange de lois (Celeux, 1985; Celeux et Govaert, 1992), Markovienne (Geman et Geman, 1987), les machines à vecteurs de support (Vapnik, 2013).

2. Méthodes déterministes : On trouve dans ce type de méthodes les réseaux de neurones (Huisman et Thijssen, 1998), k-moyennes (Pichler *et al.*, 1998), c-moyennes floues (Gath et Geva, 1989),*etc.*

L'algorithme Compression-based Texture Merging(CTM) :

La méthode de segmentation Compression-based Texture Merging (CTM) proposée par Yang *et al.* (2008) consiste à représenter le problème de segmentation d'images sous forme d'un problème de classification d'attributs couleur/texture. Premièrement L'image initiale doit être convertie dans l'espace couleur CIE Lab, représentant l'espace couleur le plus proche du système visuel humain selon les auteurs. Ensuite la distribution des attributs couleur/texture est modélisée par un mélange de distributions gaussiennes grâce à la représentation de l'image dans l'espace couleur CIE Lab. Contrairement à l'espace couleur (R,G,B), cet espace facilite la représentation de l'information texture sous la forme d'une gaussienne. De plus, les auteurs supposent que le mélange des composantes est dégénéré 4 ou bien quasi-dégénéré. Contrairement aux méthodes de segmentation de l'état de l'art. Ils montrent qu'un mélange de distributions peut être classé efficacement grâce à un algorithme de classification ascendant dérivé à partir d'une approche de compression de données avec perte. En effet, pour chaque composante couleur de l'espace CIE Lab, une fenêtre de taille fixe est centrée en chaque pixel qui constitue les valeurs de l'intensité, puis

un filtre gaussien est appliqué sur cette fenêtre. Finalement, la dimension du vecteur de descripteurs est réduite à huit à l'aide d'une analyse en composantes principales (ACP).

3.3.2.2 Méthodes de type croissance de régions

Ces méthodes consistent à faire croître chaque région autour d'un pixel de départ appelé un germe. Les régions sont créées successivement suivant deux phases : La phase du choix d'un nouveau germe (La phase d'initialisation) et la phase dans laquelle les pixels proches sont agrégés au germe selon un critère d'homogénéité jusqu'à convergence (Herlin *et al.*, 1994) (La phase itérative).

3.3.2.3 Méthodes de type division-fusion

Ces méthodes consistent à découper (division) itérativement l'image jusqu'à l'obtention de blocs homogènes selon un critère donné. Les blocs voisins sont ensuite regroupés en respectant un critère d'homogénéité. Ces méthodes peuvent faire appel à la théorie des graphes, au partitionnement de Voronoï (Bolon *et al.*, 1995), à une structure de données de type arbre quaternaire (Strasters et Gerbrands, 1991) ou aux approches pyramidales (Mathieu, 1996).

3.3.3 Approches utilisant la théorie des graphes

Les approches utilisant la théorie des graphes, consistent à créer un graphe à partir de l'image et d'exploiter les différentes propriétés de cette théorie pour trouver les différentes régions de l'image. L'idée générale est de représenter les pixels par les noeuds du graphe et les caractéristiques (distance, similarité,. . .) reliant ces pixels par des arêtes. Parmi les méthodes majeures dans ce cadre, on trouve la méthode de coupe minimale normalisée du graphe (NCUT) (Shi et Malik, 2000), et la méthode de flux (MCL) (Van Dongen, 2000).

3.3.3.1 Segmentation par coupe normalisée (NCUT)

Dans la segmentation par coupe normalisée, l'image est représentée par un graphe pondéré complet non-orienté $G = (V, E)$ avec V l'ensemble des noeuds et E l'ensemble des arêtes reliant ces noeuds. La segmentation d'une image consiste à découper le graphe correspondant pour avoir au final des coupures qui minimisent un certain critère (coût

minimal). Les noeuds représentent les pixels dans l'image avec des liens reliant ces différents pixels. Ces liens sont pondérés pas une matrice de poids W, tel que w_{ij} est le poids entre les deux pixels i et j. Le graphe ensuite est divisé en deux sous-graphes A et B où

$$A \cup B = G \texttt{ et } A \cap B = \varnothing \tag{3.2}$$

La coupure normalisée (NCUT), peut être écrite sous la forme suivante :

$$NCUT(A,B) = \frac{CUT(A,B)}{ASSOC(A,V)} + \frac{CUT(A,B)}{ASSOC(B,V)} \tag{3.3}$$

Avec $ASSOC(A,V)$ représente la somme des mesures des arêtes entre les noeuds du groupe A et tous les autres noeuds de V (les noeuds de A inclus par conséquent).

$$ASSOC(A,V) = \sum_{i \in A} \sum_{j \in V} w_{ij} \tag{3.4}$$

et CUT(A,B) dénote la somme des mesures des liens reliant les noeuds de A et B, ou bien la somme des mesures des arêtes que l'on enlèverait si on devait séparer les deux groupes A et B.

$$CUT(A,B) = \sum_{i \in A} \sum_{j \in \{V-A\}} w_{ij} \tag{3.5}$$

Cette définition normalisée de la coupure à été largement discuté dans (Shi et Malik, 2000).

3.3.3.2 *Algorithme FH (*Felzenszwalb et Huttenlocher*)*

Les auteurs dans (Felzenszwalb et Huttenlocher, 2004) utilisent une approche basée sur les graphes pour la segmentation des images. En considérant un graphe non orienté $G = (V, E)$ avec $v_i \in V$ l'ensemble des éléments (noeuds) à segmenter, et $(v_i, v_j) \in E$ les paires des noeuds voisins (liens). Chaque lien $(v_i, v_j) \in E$ est représenté par un poids non-négatif $w((v_i, v_j))$ qui mesure la dissimilarité entre les noeuds voisins v_i et v_j. Pour le cas d'une segmentation d'images les éléments de V dénotent les pixels de l'image alors que le poids d'une arête représente la mesure de dissimilarité entre chaque pair de pixels

connectés par cette arête (i.e., la couleur, la différence en intensité, texture ou autre attribut local). Dans cette méthode, la segmentation S est représentée par la partition de V en plusieurs composantes (ou régions) telle que chaque région $C \in S$ correspond à une composante connectée dans le graphe G'=(V,E'), où $E' \subseteq E$.

3.3.4 Approches utilisant les algorithmes de détection des communautés

3.3.4.1 Méthode de Li et Wu (2014)

Li et Wu (2014) proposent un algorithme de segmentation d'image basé sur l'optimisation de la modularité dans les graphes. En premier temps, ils commencent leur algorithme par une segmentation initiale en utilisant la méthode des superpixels qui permet de segmenter l'image en un ensemble de petits segments. Chaque segment représente une région dans le graphe, puis le graphe de régions est construit à l'aide des caractéristiques d'image pour calculer la similarité entre les régions. Cette similarité est attribuée en tant que pondération au graphe. Enfin, à partir du graphe de régions, Li et Wu (2014) appliquent un algorithme de détection des communautés basé sur l'optimisation de la modularité.

3.3.4.2 Méthode de Abin et al. (2014)

Dans le travail de (Abin *et al.*, 2014), les auteurs commencent par une segmentation initiale à l'aide de meanshift, puis ils construisent le graphe en utilisant la similarité entre deux régions en utilisant uniquement l'information de la couleur. Cette similarité est utilisée comme un poids pour les arêtes du graphe de régions. Finalement, ils appliquent un algorithme de détection des communautés pour obtenir l'image segmentée. Néanmoins, les auteurs n'ont pas utilisé un processus itératif pour éviter le problème de sur-segmentation, mais utilisent un algorithme de post-traitement pour fusionner des régions avec des zones inférieures à un seuil prédéfini avec d'autres régions. Si une région est inférieure au seuil t, elle est fusionnée avec la région adjacente la plus similaire du réseau.

Cependant, ces deux méthodes sont généralement sensibles au bruit et utilisent soit la couleur ou la texture comme mesure pour calculer la similarité entre les régions de l'image, ce qui conduit à une sur-segmentation en négligeant les régularités à l'intérieur

de l'image. De plus, la plupart de ces méthodes reposent sur des algorithmes de détection des communautés qui ont un coût de calcul élevé. Pour surmonter ces limitations, le framework proposé commence par une segmentation initiale pour diviser l'image en régions qui doivent être cohérentes et à conserver la plupart des informations nécessaires pour la segmentation. Ensuite, un graphe de régions adjacentes (RAG) est utilisé pour représenter l'image où chaque région représente un noeud dans le graphe. Un lien entre deux régions est considéré si elles sont adjacentes. Afin de pondérer le RAG, une combinaison de caractéristiques de texture et de la couleur est utilisée pour mesurer la similarité entre les noeuds. Finalement, en se basant sur des algorithmes de détection des communautés efficaces, qui établissent le meilleur équilibre entre le coût de calcul et les performances de segmentation, nous extrayons des communautés représentant les régions de l'image. Le processus est répété de manière itérative jusqu'à ce que la segmentation optimale soit atteinte.

3.4 Segmentation des images : Bases de données et critères d'évaluation

Berkeley Segmentation Dataset (BSD). Le benchmark publique (Arbelaez *et al.*, 2010) a deux versions. La première est appelée BSD300, qui comprend 300 images avec ses données de vérité de terrain (chaque image comporte au moins 4 annotations humaines), est divisée en un ensemble d'apprentissage contenant 200 images et un ensemble de test comprenant 100 images. La seconde est le BSDS500, une version étendue du BSDS300 qui comprend 200 nouvelles images de test. La taille de chaque image est de 481×321. La figure 3.1 présente quelques exemples tirés de la BSD. Notez que chaque image a plusieurs vérités du terrain annotées par différents observateurs humains.

Semantic Segmentation Data Set (SSDS) (Li *et al.*, 2013), qui comprend 100 images sélectionnées à partir de BSDS500, ainsi que les vérités de terrain au niveau sémantique générées à l'aide des vérités de terrain existantes de BSDS500, ainsi elle contient un outil de segmentation interactif. SSDS est aussi intégré avec le concept d'arbre de perception, qui permet aux vérités de terrain de s'adapter via une simple fusion de régions. La figure 3.2 présente des exemples d'images de BSDS500 et les segmentations de vérité

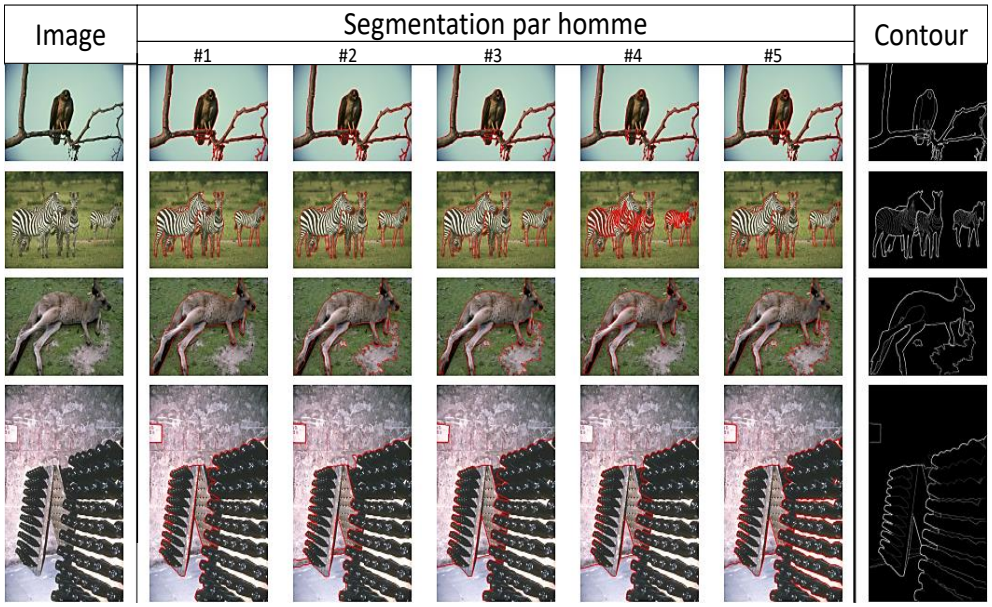

FIGURE 3.1 – La base de données de segmentation Berkeley.

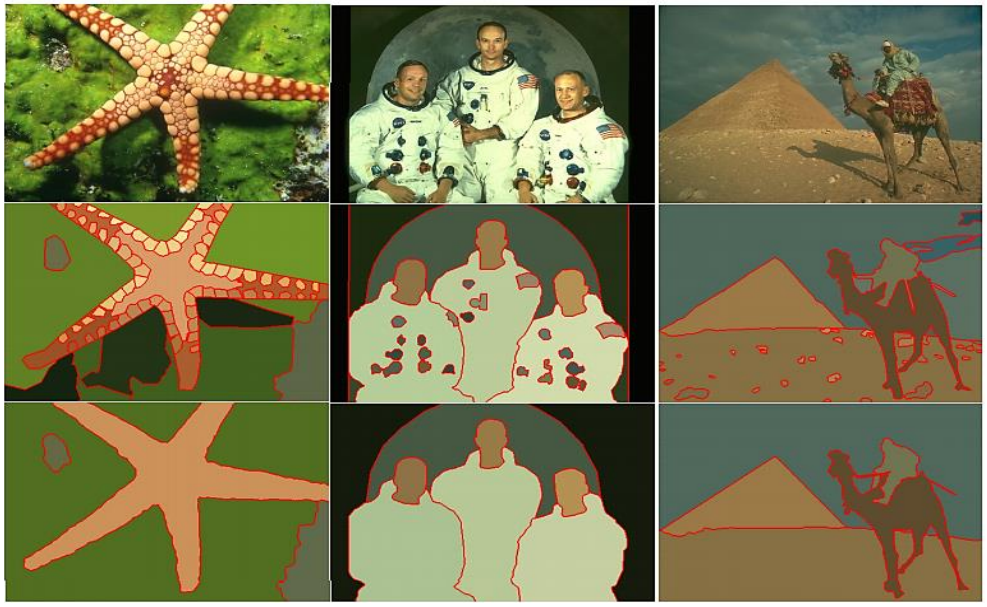

FIGURE 3.2 – Échantillons des images, haute : Les images originales, milieu : Segmentation de la vérité du terrain de BSDS500, bas : Segmentation de la vérité du terrain de SSDS.

de terrain correspondantes fournies par BSDS500 et SSDS. On peut voir que certaines segmentations de vérité de terrain fournies par BSDS500 ont une granularité fine, alors que SSDS donne une meilleure segmentation de vérité de terrain au niveau sémantique.

D'après plusieurs années, nombreux algorithmes de segmentation d'images ont été proposés, il était essentiel de les comparer en utilisant différents paramètres, afin de les comprendre sur un terrain expérimental pour choisir le plus convenable. Ce domaine a suscité un vif intérêt et de nombreuses mesures d'évaluation ont été développées. Zhang *et al.* (2008) ont présenté une bonne étude sur ce sujet.

Dans une perspective différente, de nombreux algorithmes d'évaluation ont été proposés. En règle générale, les chercheurs ont tendance à évaluer l'algorithme proposé avec plusieurs mesures d'évaluation différentes pour présenter pleinement ses performances. Par exemple, de nombreuses méthodes (Abin *et al.*, 2014; Li et Wu, 2014) utilisent le PRI, VOI , *etc*, comme une combinaison standard pour comparer avec d'autres algorithmes de référence standard. Selon ces travaux, un résultat de segmentation est jugé bon, au moins quantitativement, si la comparaison avec la vérité du terrain donne une valeur élevée pour PRI et une valeur faible pour VOI. Nous répertorions plusieurs indicateurs d'évaluation couramment utilisés dans la segmentation d'images.

The Probabilistic Rand Index (PRI) (Unnikrishnan *et al.*, 2007) mesure la fraction des paires de pixels dont les labels sont cohérents entre le résultat de la segmentation et la vérité du terrain. En pratique, les PRI peuvent être calculés sous une forme simple. Soient S_{ground} et S_{test} deux segmentations de la même image avec un nombre différent de segments, et n_{ij} le nombre de points dans le ith segment dans S_{ground} et jth segment dans S_{test}. N est le nombre total de pixels de l'image. La similarité entre les deux segmentations est la suivante :

$$PRI(S_{ground}; S_{test}) = [\binom{N}{2} - 1/2[\sum_i(\sum_j n_{ij})^2 + \sum_j(\sum_i n_{ij})^2 - \Sigma\Sigma n_{ij}^2]]/\binom{N}{2} \tag{3.6}$$

la valeur de PRI, qui mesure la similarité des deux segmentations, va de 0 (lorsqu'il n'y a pas une similarité entre S_{ground} et S_{test}) à 1 (lorsque les deux segmentations sont identiques).

Variation of Information (VOI) (Meila, 2005) est une métrique qui mesure la somme du gain d'informations et de la perte d'informations entre deux segmentations. La métrique

VOI est non négative, plus elle est petite, plus la similarité est grande. C'est défini par la formule ci-dessous :

$$VOI(C, C') = H(C) + H(C') - 2I(C, C') \tag{3.7}$$

où H (C) et H (C') désignent respectivement l'entropie des deux segmentations C et C' et I(C,C') désignent l'information mutuelle de C et C'. La valeur de VOI varie entre $[0, \infty]$, et plus la valeur est petite, plus les deux segmentations sont similaires.

Précision et rappel sont deux mesures intéressantes pour mesurer la qualité de la segmentation, car elles sont sensibles à la sous/sur -segmentation, la sous-segmentation conduit à de faibles scores de rappel, tandis que la sur-segmentation conduit à des faibles scores de précision. La précision mesure la fraction des frontières détectées des pixels qui sont similaires aux frontières de la vérité du terrain. Elle est définie comme suit :

$$\text{Precision} = \frac{|\text{S}_{\text{test}}| \cap |\text{S}_{\text{gt}}|}{|\text{S}_{\text{test}}|} \tag{3.8}$$

tel que S_{gt} est la segmentation de vérité du terrain et S_{test} est la segmentation du test et $|S|$ désigne le nombre des frontières de pixels dans la segmentation S.

Le rappel calcule le pourcentage des frontières de pixels de la vérité de terrain qui sont détectées, il est défini comme suit :

$$\text{Recall} = \frac{|\text{S}_{\text{test}}| \cap |\text{S}_{\text{gt}}|}{|\text{S}_{\text{gt}}|} \tag{3.9}$$

F_α-measure est une mesure de qualité basée sur la précision et le rappel, elle mesure la moyenne harmonique de la précision et du rappel, elle est défini comme suit :

$$\text{F-measure} = \frac{\text{Precision.Recall}}{(1-\alpha).\text{Recall} + \alpha.\text{Precision}} \tag{3.10}$$

3.5 Conclusion

Dans ce chapitre nous avons décrit les différentes catégories de méthodes pour la segmentation des images, comme les approches contours, régions, graphes et finalement les

approches de détection des communautés qui font l'objectif de la première partie de ce mémoire sur la segmentation des images. Nous avons présenté par la suite quelques bases de données des images qui sont utilisées dans l'évaluation des méthodes de segmentation. Finalement, les métriques les plus utilisées pour évaluer les performances de ces algorithmes ont été entamées dans la dernière section. Le prochain chapitre sera consacré à la description du framework proposé pour la segmentation des images en se basant sur les algorithmes de détection des communautés.

Chapitre

4 Segmentation des images par les approches de détection des communautés

Sommaire

4.1 Introduction

Avec le développement de la théorie des réseaux complexes, la segmentation d'images en se basant sur les graphes a considérablement évoluée. L'identification des régions de pixels peut être remplie par des méthodes de détection des communautés sur les noeuds d'un graphe. La détection des communautés est un sujet très prolifique dans la littérature des réseaux complexes. Une grande variété d'algorithmes ont été développés jusqu'à présent pour traiter ce problème. Ce chapitre commence par la présentation du schéma global du framework proposé pour la segmentation des images en se basant sur les al-

gorithmes de détection des communautés. Premièrement la méthode commence par une segmentation initiale de l'image pour ensuite construire le graphe des régions adjacentes (Region Adjacency Graph RAG) (Trémeau et Colantoni, 2000). Les noeuds sont les régions initiales et les arêtes existent entre deux noeuds, s'ils sont adjacents. Puis l'arête est pondérée selon la similarité entre les caractéristiques de l'image (texture et couleur). Un algorithme de détection des communautés est appliqué ensuite sur le graphe pondéré des régions adjacentes pour partitionner le graphe à un ensemble des communautés. Ces communautés sont utilisées pour regrouper les régions adjacentes de l'image. En effet, tous les noeuds appartenant à la même communauté sont considérés comme appartenant à la même région et fusionnés en une seule région dans l'image. Le processus est itératif jusqu'à ce qu'il n'y ait plus aucune différence entre les structures communautaires de deux itérations successives. À la fin de ce chapitre, une évaluation quantitative et qualitative du framework proposé avec les méthodes de l'état de l'art a été présentée.

4.2 Schéma global du framework proposé

En raison des propriétés inhérentes de l'image, les problèmes de segmentation et de détection des communautés sont différents. L'utilisation d'algorithmes de détection des communautés uniquement pour segmenter une image en considérant les pixels comme des noeuds sur un graphe peut conduire à des faibles performances. L'échec d'une telle méthode peut s'expliquer par plusieurs raisons. Tout d'abord, lorsque nous segmentons une image, les pixels peuvent avoir différentes propriétés, par exemple des couleurs différentes, mais dans la détection de communauté, les noeuds peuvent partager des caractéristiques similaires. Deuxièmement, nous ne pouvons pas prendre des régularités et des informations pour des segments homogènes de l'image en utilisant juste un seul pixel. Troisièmement, en comparant avec les communautés, les images partagent certaines informations. Par exemple, deux régions adjacentes appartiennent probablement à la même communauté. Donc, pour résoudre les problèmes mentionnés, le schéma proposé prend en considération les propriétés inhérentes de l'image, ainsi que de l'optimisation efficace de la modularité/stabilité à l'aide des algorithmes de détection de communauté. Dans cette section, nous décrivons les étapes du schéma proposé afin que le lecteur puisse avoir une vue glo-

bale de tout le schéma avant d'approfondir dans les détails. Nous nous référons à la Figure 4.1 pour l'illustration des étapes du schéma général proposé. Les détails de chaque étape et certains points techniques sont expliqués dans les sections suivantes.

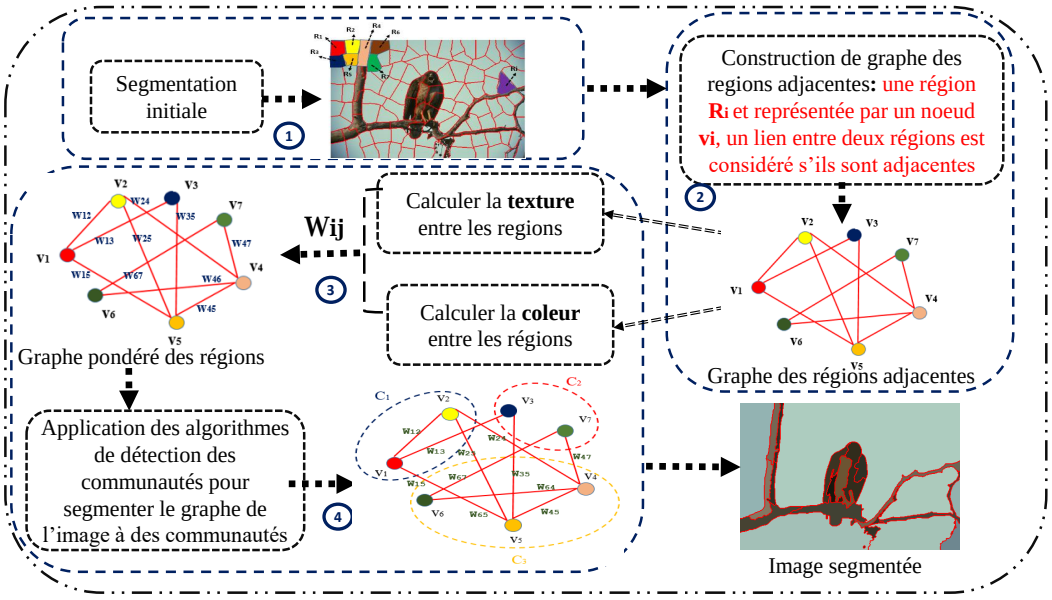

FIGURE 4.1 – Le schéma global du framework proposé pour une itération.

4.3 Segmentation initiale

Le but de la segmentation initiale est de segmenter l'image à des petites régions homogènes. Différentes techniques de bas niveau pour la segmentation peuvent être utilisées comme super-pixels, meanshift, levelset, et watershed.

4.3.1 Algorithme de super-pixels

L'algorithme de super-pixels divise une région en une multitude de régions plus petites. Il est généralement utilisé comme processus de segmentation initiale pour réduire le nombre de pixels et la complexité de calcul des tâches ultérieures. Plusieurs études existent pour l'extraction de super-pixels. Dans (Çığla et Alatan, 2010), les auteurs proposent une technique efficace permettant d'obtenir des super-pixels quasi-uniformes à faible coût de calcul. Les résultats obtenus par la méthode montrent son efficacité en termes de coûts de calcul, de compacité des segments, et des erreurs de sur-segmentation. Ils utilisent un

algorithme de K-moyennes connecté avec une contrainte de convexité pour extraire les super-pixels. Selon un certain nombre de régions souhaitées par l'utilisateur, l'image est divisée en régions rectangulaires (segments) à l'aide d'une grille. Ensuite, en minimisant une fonction de coût :

$$C_{x,y}(i) = \lambda_1.|I(x,y) - I_i| + \lambda_2.|(x - C_x^i)^2 + (y - C_y^i)^2)| \tag{4.1}$$

Tel que x et y sont les positions des pixels testés dans les différents segments ; λ_1 et λ_2 correspondent respectivement, à la pondération de la similarité d'intensité et les contraintes de convexité ; I_i l'intensité moyenne de i^{th} segment. C_x^i et C_y^i les positions du centre de i^{th} segment. L'algorithme de super-pixels teste les pixels qui se trouvent aux contours du segment et les attribue aux nouveaux segments. Dans cette contribution proposée, le code public qui est dans (Mori, 2005) est utilisé pour obtenir l'initialisation par l'algorithme de super-pixel.

4.3.2 Algorithme de Meanshift

Meanshift est un algorithme itératif non paramétrique (Comaniciu et Meer, 2002) qui peut être utilisé dans plusieurs tâches, telles que la recherche de modes, la classification, *etc.* L'un des avantages de Meanshift par rapport à d'autres techniques de pré-segmentation est que nous n'avons pas besoin de spécifier le nombre de segment (classes), car l'algorithme lui-même trouve le meilleur nombre de segments pour l'image. Pour démarrer l'algorithme MeanShift sur un ensemble de points de données X (pixels dans l'image), il faut :

- Une fonction N (x) qui détermine quels sont les voisins d'un point $x \in X$. Les points voisins sont les points situés à une certaine distance. La métrique de distance est généralement la distance euclidienne.

- Une fonction du noyau K (d) à utiliser dans Meanshift. K est généralement un noyau gaussien et d est la distance entre deux points de données.

Maintenant, avec les fonctions ci-dessus, la procédure de Meanshift pour un ensemble

de points de données X suit ces étapes :

1. Pour chaque point $x \in X$, trouver le voisinage des points N(x) de x.

2. Pour chaque point $x \in X$, calculer le mean shift m(x) à partir de l'équation 4.2 :

$$m(x) = \frac{\Sigma_{x_i \in N(x)} K(x_i - x) x_i}{\Sigma_{x_i \in N(x)} K(x_i - x)} \tag{4.2}$$

3. Pour chaque point $x \in X$, modifier $x \leftarrow m(x)$.

4. Refaire 1, pour n itérations ou jusqu'à les points ne se déplacent plus.

Le processus de segmentation initial est très important pour le fonctionnement de la méthode proposée. Puisque un seul pixel ne peut capturer aucune information sur la texture, utiliser des régions au lieu de pixels permet de capturer plus de détails dans l'image et d'empêcher la perte de variations locales. Il est montré que les régions ont l'avantage de s'adapter à la structure de l'image, étant plus grandes là où la couleur reste similaire sur une grande surface. De plus, les régions diminuent brusquement le nombre de noeuds dans un graphe de millions à des milliers de noeuds, d'où la complexité des calculs est automatiquement diminuée, sans affecter les performances de la segmentation. Pour toutes ces raisons, nous utilisons l'approche de segmentation initiale pour fournir de très petites régions de pixels contenant des informations et des régularités et qui seront utilisées pour calculer la similarité entre les régions. De plus, la méthode proposée permet d'utiliser diverses méthodes qui ont été adoptées pour produire cette sur-segmentation.

4.4 Construction du graphe de régions adjacentes

Contrairement aux réseaux conventionnels, les images contiennent des informations spatiales par rapport aux réseaux sociaux ou aux réseaux de citations. Les régions adjacentes dans l'image sont souvent considérées comme un seul segment dans l'image, par rapport aux autres régions qui sont éloignées. Nous construisons donc le graphe des régions adjacentes RAG en utilisant les informations spatiales de l'image. Soit G = (V, E) un graphe non orienté, où $vi \in V$ est un ensemble de noeuds qui correspond aux

régions de l'image Ri. E est un ensemble d'arêtes reliant les paires de noeuds voisins. En d'autres termes, une arête est créé entre deux noeuds si leurs régions correspondantes sont adjacentes dans l'image. Comme le montre la figure 4.2, le RAG est construit après la segmentation initiale, chaque région de l'image étant considérée comme un noeud du graphe.

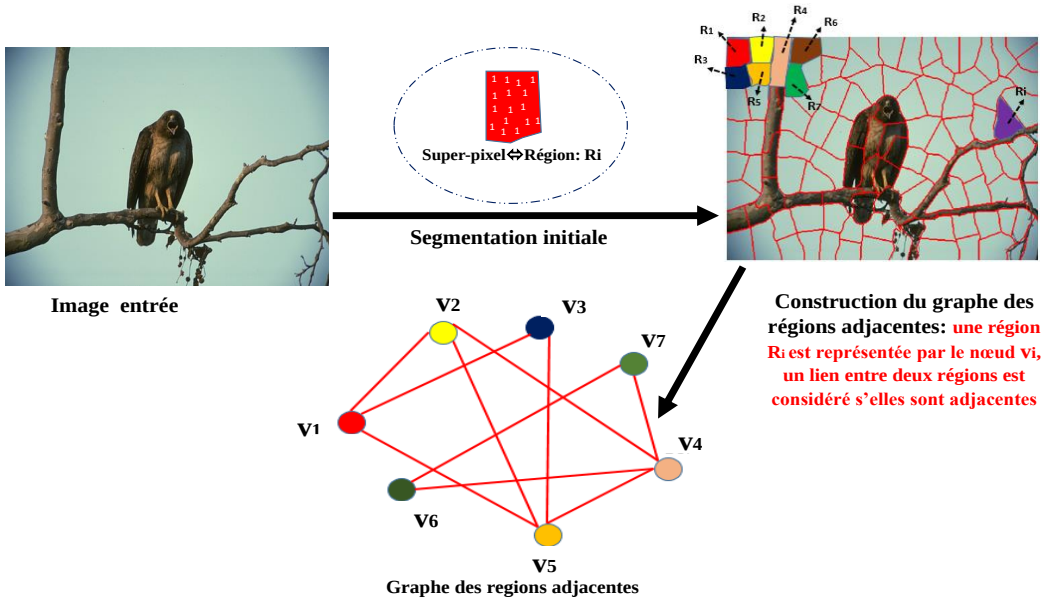

FIGURE 4.2 – La procédure de construction du graphe des régions adjacentes à partir de la segmentation initiale, chaque région Ri est représentée par un noeud dans le RAG.

4.5 Pondération du RAG

Dans cette étape, nous calculons le poids en utilisant la similarité entre les régions, plusieurs mesures peuvent être employées pour calculer la similarité entre les régions adjacentes du RAG.

4.5.1 Similarité par la couleur

La couleur dans la segmentation est une caractéristique importante. Chaque pixel d'une image couleur est représenté par un vecteur tridimensionnel. Nous supposons que la valeur d'intensité de pixel d'une région donnée suit une distribution gaussienne. Par conséquent, la distribution d'une région Ri est donnée par : $R_i \sim N(\mu_i; var_i)$ tel que μ_i est

la moyenne du vecteur de l'intensité du pixel calculé l'espace de couleur tridimensionnel des régions R_i, var_i représente la variance de R_i.

Pour mesurer la similarité entre les deux distributions, différentes mesures de distance sont proposées dans la littérature, comme la distance Earth Mover (en anglais Earth Mover's Distance (EMD)) (Puzicha *et al.*, 1999), la divergence Kullback-Leibler (KL) (Joyce, 2011), la distance moyenne (en anglais Mean Distance (MD)), *etc.* Nous avons choisi d'utiliser la distance moyenne dans ce travail, parce qu'elle est généralement est une bonne approximation de la distance d'Earth Mover avec une complexité plus basse. La distance moyenne peut être définie par l'équation 4.3

$$\mathrm{D_{MD}(R_i; R_j)} = (\mu_\mathrm{i} - \mu_\mathrm{j})^\mathrm{T}(\mu_\mathrm{i} - \mu_\mathrm{j}) \tag{4.3}$$

pour transformer la distance de la distribution couleur à une mesure de similarité, on utilise le noyau de la fonction de base radiale défini par l'équation 4.4 :

$$\mathrm{c_{ij}} = \exp\left(\frac{-\mathrm{D_{MD}(R_i; R_j)}}{2\sigma^2}\right) \tag{4.4}$$

tel que σ est un paramètre défini par l'utilisateur.

Plusieurs espaces de couleur peuvent être utilisés comme RGB, YUV, L*a*b, HSV, *etc.* Choisir un espace calorimétrique approprié pour segmenter une image couleur est une étape cruciale pour obtenir de meilleures performances de segmentation. Grâce à son accord avec le système visuel humain (Wyszecki et Stiles, 1982), nous avons choisi l'espace calorimétrique LAB. Il s'agit d'un espace de couleurs opposés sur 3 axes avec la dimension L pour la légèreté et A et B pour les dimensions de la couleur opposée.

4.5.2 Similarité par la texture

Utiliser uniquement la couleur pour mesurer la similarité ne permet pas d'obtenir un bon résultat de segmentation, car la couleur de certains objets homogènes décompose les régularités de l'image en différents segments. Par conséquent, nous proposons la caractéristique de texture comme solution pour remédier à ce problème. Beaucoup d'approches récentes utilisent les ondelettes comme caractéristiques, d'autres méthodes, telles que

(Bernstein et Amit, 2005), permettent d'entrainer des dictionnaires de structures locales à partir des images d'apprentissage. Dans ce travail, nous utilisons une caractéristique appelée Histogramme de Gradients Orientés (HOG) pour la texture. Elle est bien connue dans le traitement des images et la vision par ordinateur pour la détection d'objets dans l'image. Elle calcule le nombre d'occurrences d'orientation du gradient dans des parties localisées de l'image. Pour construire l'histogramme des gradients orientés, on procède comme suit :

- Premièrement nous avons besoin de calculer le gradient horizontal et vertical, puis on calcule l'histogramme du gradient pour une région de l'image

- Dans la deuxième étape, la région de l'image est divisée à des petites cellules de taille $C \times C$ pixels (C = 8). Pour chaque cellule, l'histogramme des directions du gradient est calculé. L'histogramme est essentiellement un vecteur de 9 bins.

- Dans la troisième étape, on utilise une méthode qui s'appelle normalisation des blocs, pour grouper les cellules individuelles dans des blocs et les normaliser pour assurer l'invariance face aux changements d'éclairage. Un bloc est représentée par 2×2 cellules, tel que chaque bloc a une taille de $2C \times 2C$ pixels (4 histogrammes).

- Finalement, le vecteur caractéristique finale est calculé pour toute la région R_i, tel que les histogrammes du vecteur gradient des blocs h_c est groupés dans un seul vecteur caractéristique HOG H_i :

$$H_i = [h_1, ..., h_c] \tag{4.5}$$

tel que h_c représente l'histogramme des vecteurs gradients pour un bloc, c est le nombre de blocs dans une région R_i. Pour calculer la similarité entre deux régions R_i et R_j, on utilise la mesure de similarité par cosinus qui est défini par l'équation 4.6 :

$$\mathrm{t_{ij}} = \cos(\mathrm{H_i}, \mathrm{H_j}) = \frac{\mathrm{H_i^T H_j}}{\|\mathrm{H_i}\|.\|\mathrm{H_j}\|} \tag{4.6}$$

tel que ||.|| est la norme L_2, H_i et H_j sont respectivement les vecteurs HOG des régions R_i et R_j.

Pour calculer le poids en utilisant la similarité entre les régions, Des caractéristiques telles que la couleur et la texture de l'image sont utilisées entre deux régions adjacentes du RAG. Pour calculer la matrice de similarité W (RAG pondéré), nous utilisons les équations (2) et (4) pour mesurer la similarité entre chacune des deux régions adjacentes. Ensuite, nous associons un poids entre eux. Contrairement au framework proposé dans (Linares *et al.*, 2017) où les auteurs ont construit un graphe en considérant chaque noeud comme un super-pixel connecté selon une fonction de pondération, qui est basée uniquement sur la couleur. Dans ce travail, la similarité est calculée en utilisant une combinaison des caractéristique LAB et HOG. Nous nous référons à (Sumengen *et al.*, 2006) pour la construction de la matrice de similarité (RAG pondéré). Dans (Sumengen *et al.*, 2006), les auteurs utilisent un modèle hybride combinant deux caractéristiques. Dans ce travail, nous choisissons les caractéristiques de texture (HOG) et de la couleur (LAB) pour calculer le poids comme il est défini dans l'équation 4.7 :

$$\begin{aligned} \mathrm{W} = \mathrm{w_{ij}} = \mathrm{a} \times \sqrt{\mathrm{t_{ij}} \times \mathrm{c_{ij}}} + (1 - \mathrm{a}) \times \mathrm{c_{ij}}; \\ (\mathrm{i}, \mathrm{j}) = 1, .., \mathrm{n} \end{aligned} \tag{4.7}$$

tel que n est le nombre de régions et a est un paramètre d'équilibrage.

4.6 Extraction des communautés

Dans cette étape, à partir du RAG pondéré, on extrait les communautés à l'aide des algorithmes de détection de communauté. Pour trouver la meilleure partition du graphe offrant une modularité ou une stabilité maximale, plusieurs algorithmes ont été proposés dans la littérature. Contrairement aux approches proposées dans (Li et Wu, 2014) et (Abin *et al.*, 2014) qui utilisent des algorithmes de détection des communautés qui ont un coût de calcul élevé et ne produisent pas toujours la meilleure segmentation, tels que l'algorithme de Newman-Fast et l'algorithme de l'optimisation de la modularité. Le framework proposé

utilise des algorithmes de détection des communautés efficaces.

Afin de choisir les algorithmes de détection des communautés appropriés, nous avons fait recours à des travaux de synthèse qui sont utilisés pour rechercher puis évaluer ces algorithmes. Selon (Orman *et al.*, 2011b), (Orman *et al.*, 2011a), (Lancichinetti et Fortunato, 2009), les algorithmes proposés par Ronhovde et Nussinov (Fast multi-scale community detection algorithm using the criterion from Ronhovde and Nussinov (FMCDRN)) (Ronhovde et Nussinov, 2010), Infomap (Rosvall et Bergstrom, 2008), l'algorithme d'optimisation rapide de la modularité gloutonne (Fast greedy modularity optimization algorithm (FGMDO)) (Clauset *et al.*, 2004) et Louvain (Blondel *et al.*, 2008) sont jugés capables de fournir une estimation raisonnable du nombre des communautés pour différentes tailles de réseaux, puis surpasse tous les algorithmes de l'état de l'art dans la détection des communautés. D'avantage, ce framework permet d'utiliser tous les algorithmes de détection des communautés existants dans l'état de l'art pour cette étape.

4.7 Processus de regroupement des communautés

Un processus itératif est utilisé dans cette étape pour construire la matrice de similarité et pour recalculer les poids entre les régions à chaque itération selon l'équation 4.7, car lorsque nous utilisons des algorithmes de détection des communautés, les régions continuent à s'élargissent à chaque itération, et la mesure de similarité donnée par l'itération précédente peut ne pas être convenable pour l'itération actuelle. Ainsi, la mise à jour des poids du RAG à chaque itération permet de réévaluer les pondérations entre les régions. Ce processus évite les nombreuses petites régions qui se créent dans l'image, et qui devraient être fusionnées dans la même communauté selon la perspective du système visuel humain. L'algorithme du framework proposé peut être résumé dans Algorithme 4.1.

Algorithme 4.1

Entrées: Une image couleur I

Sorties : l'ensemble des segments de l'image $C_i = \{C_{i1}, ..., C_{ic}\}$ avec $c \leq n$

Calculer l'ensemble initial de régions $R = \{R_1, ..., R_n\}$ tel que n est le nombre des régions

Initialiser l=0

initialiser la structure des communautés $C_0 = \{C_{01}, ..., C_{0n}\}$ chaque région est associé à un communauté

do

Construire le RAG ;

Affecter chaque noeud à une région associé une arête entre deux noeuds s'ils sont adjacents

Calculer la valeur de la texture et la couleur pour chaque région R_i ;

Calculer les poids pour le RAG (W) selon l'équation (10), $w_{ij} \neq 0$ seulement si R_i et R_j sont des régions adjacentes dans le RAG ;

Calculer la structure communautaire du RAG pondéré en utilisant un algorithme de détection des communautés $C_l = \{C_{l1}, ..., C_{lm}\}$ tel que m est le nombre courant des communautés ;

regrouper les régions qui appartiennent à les même communautés ;

l=l+1 ;

while *la structure communautaire change* ($C_l \neq C_{l-1}$)

4.8 Résultats et Discussions

4.8.1 Présentation de la démarche d'application et d'évaluation

4.8.1.1 Données

Afin de vérifier les performances du framework proposé pour la segmentation des images par rapport aux autres méthodes, une segmentation manuelle des images de la base de données est requise. La base de données de segmentation Berkeley 500 (BSDS500) (Arbelaez *et al.*, 2010), accessible au public, a été utilisée pour évaluer les performances du framework proposé. BSDS500 contient 500 images naturelles, dont 200 images sont pour l'apprentissage, 200 images pour le test et 100 images pour la validation. Les contours sont étiquetées pour chacune des 500 images de taille 481×321 par plusieurs personnes et sont moyennées pour former la vérité du terrain. En ce qui concerne l'évaluation de la performance de la segmentation, nous utilisons trois mesures différentes, à savoir la Probability Rand Index (PRI), la Variation de l'Information (VOI), la précision, le rappel et la F-mesure. Ces métriques ont des caractéristiques différentes, par exemple, le PRI a

tendance à sur-segmenter les images, tandis que VOI encourage la sous-segmentation. Par conséquent, la prise en compte globale de ces trois paramètres est nécessaire. Notez qu'une bonne segmentation correspond à une valeur élevée de PRI, de précision, de rappel et de F-mesure, mais à une valeur faible pour le VOI. Toutes les méthodes ont été implémentées sous Matlab 2016 avec un processeur Intel CPU i5 de 3.10 GHz et 4 Go de mémoire.

4.8.1.2 Méthodologie

Pour étudier les performances du framework proposé, nous étudions d'abord l'influence de certains paramètres utilisés pour calculer la similarité entre les régions. Nous réalisons des premières expériences sur 100 images de BSDS500 en utilisant empiriquement plusieurs valeurs du paramètre d'équilibrage **"a"** pour déterminer la valeur appropriée permettant d'obtenir les meilleurs résultats de segmentation. Deuxièmement, une comparaison entre les caractéristiques de l'image (couleur et texture) est effectuée. À cet égard, nous examinons la performance du framework quantitativement et qualitativement, dans les cas où juste la texture est seulement utilisée, la couleur seulement et pour le cas de la couleur et la texture sont les deux utilisées dans le processus de segmentation. Troisièmement, afin de choisir la méthode de segmentation initiale appropriée pour le framework, nous effectuons des expériences sur 100 images de BSDS500 et comparons les valeurs moyennes de PRI, VOI, Précision et Rappel pour chaque méthode. Ensuite, nous choisissons la méthode qui assure les meilleurs résultats de segmentation. Nous étudions ensuite l'influence des méthodes de détection des communautés, dans notre étude nous avons choisi les 4 algorithmes discuté dans la sous-section 4.6 : L'algorithme FMCDRN (Ronhovde et Nussinov, 2010), Infomap (Rosvall et Bergstrom, 2008), FGMDO (Clauset *et al.*, 2004) et Louvain (Blondel *et al.*, 2008). Nous considérons que toutes les images dans BSDS500 sont classées en quatre catégories, à savoir les personnes, les paysages urbains, les animaux et les paysages naturels. Nous prenons pour chaque catégorie cinq images choisies au hasard avec leurs vérités du terrain, et nous comparons les résultats des quatre méthodes qualitativement et quantitativement en utilisant la valeur moyenne des métriques PRI, VOI, Précision et Rappel pour chaque catégorie. Enfin, avec les paramètres et les méthodes appropriés décrits précédemment, nous effectuons une comparaison qua-

litative et quantitative du framework proposé avec des méthodes de l'état de l'art : (Li et Wu, 2014), (Abin *et al.*, 2014), CTM (Yang *et al.*, 2008) et EDISON (Christoudias *et al.*, 2002). Dans (Li et Wu, 2014) et (Abin *et al.*, 2014), nous conservons les mêmes paramètres que ceux utilisés par les auteurs. Dans la méthode EDISON (Christoudias *et al.*, 2002) qui repose sur l'implémentation de Meanhift dans les schémas d'extraction de limites et de filtrage du bruit, le paramètre principal d'EDISON est la taille de la région minimale. Nous avons donc défini la valeur de ce paramètre à 1000 pour éviter la création de petites régions. Dans CTM (Yang *et al.*, 2008), nous utilisons le modèle de mélange gaussien pour ajuster les textures de l'image. Pour trouver la segmentation optimale, nous utilisons le principe de la longueur minimale de description, qui produit la Longueur de description minimale sous un certain taux de distorsion. Nous utilisons le taux de distorsion $\epsilon = 0.2$.

4.8.2 Résultats des expérimentations

Nous avons discuté les points essentiels du framework proposé, mais jusqu'à présent, aucun résultat n'a été montré. Par souci d'exhaustivité et d'illustration, dans cette section, la performance du framework proposé est évaluée qualitativement et quantitativement en fournissant des expériences.

4.8.2.1 Influence des mesures de similarité

L'ajustement du paramètre d'équilibrage a :

Comme il est mentionné dans la section 4.5, durant chaque itération, on utilise une combinaison entre la caractéristique de la couleur LAB et la caractéristique de la texture HOG, pour calculer la similarité entre les régions en utilisant l'équation 4.7, avec a est un paramètre d'équilibrage. Si a = 0, cela signifie que les informations de texture ne sont pas prises en compte. Nous pouvons observer sur la figure 4.3 que la roue du véhicule et sa partie supérieure sont bien codées dans la similarité. Avec l'augmentation de la valeur de a, de plus en plus des informations dans l'image sont codés dans la similarité, en préservant en même temps les régularités. Toutefois, si a est trop grand dans notre cas a = 0,8, la roue et la partie supérieure du véhicule de l'image sont fusionnées en un seul segment. Nous faisons plusieurs tests pour déterminer la meilleure valeur du paramètre a, nous

avons varié a de 0 à 1 avec un pas de 0,2 car lorsque nous avançons par un pas de 0,1 ou 0,05, nous ne pouvons observer aucun changement dans le résultat de la segmentation. Les résultats montrent que a = 0.4 donne les meilleures performances en terme de métrique PRI et VOI. Dans toutes les expériences suivantes, nous utilisons a = 0.4.

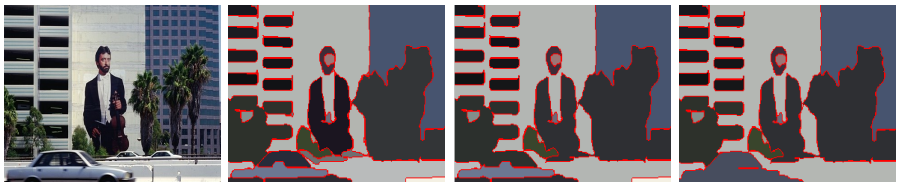

FIGURE 4.3 – Segmentation avec différentes valeurs de **a** par l'algorithme de FMCDRN.

Comparaison entre les caractéristiques de similarité :

Une question importante pour le framework proposé est d'évaluer l'influence des informations de couleur (LAB) et de la texture (HOG) dans le processus de segmentation. Nous comparons les performances du framework proposé dans les cas où HOG seul, LAB seul et pour le cas de HOG + LAB sont utilisées dans le processus de segmentation. Nos expériences ont été testées sur 100 images d'un ensemble de données de segmentation de Berkeley, en utilisant le meanshift comme une segmentation initiale et le FMCDRN comme méthode de détection des communautés. Le compromis entre la texture et la couleur est calculé par le poids w_{ij} dans l'équation (10). Pour obtenir la segmentation par la texture seule où la couleur seule, nous ajustons le paramètre d'équilibrage a avec des réglages manuels (c.-à-d. a= 0 pour la segmentation uniquement par la couleur et a = 1 pour la segmentation par uniquement la texture). Comme dans BSDS500, il existe plusieurs cartes de segmentation de vérité du terrain, 5 cartes de segmentation pour chaque image. Dans nos expérimentations, la valeur moyenne des métriques calculées est utilisée entre le résultat de la segmentation et toutes les cartes de segmentation pour chaque image. Comme illustré dans le tableau 4.1, qui présente les valeurs moyennes des PRI, VOI, Précision et Rappel pour 100 images, nous pouvons noter que les résultats de la texture toute seule et de la couleur toute seule sont généralement inférieurs aux résultats obtenus par la combinaison entre la texture et la couleur pour toutes les métriques.

TABLEAU 4.1 – Comparaison quantitative entre les caractéristiques texture et couleur.

	PRI	VOI	Précision	Rappel	F-measure
framework avec Texture (HOG)	0.708	1.808	0.698	0.571	0.628
framework avec Couleur (LAB)	0.715	1.794	0.685	0.589	0.633
framework avec Texture-Couleur(HOG+LAB)	0.828	1.695	0.788	0.621	0.694

La figure 4.4 montre des exemples de résultats visuels pour la segmentation de trois images de BSDS500. Nous pouvons observer que l'utilisation d'une seule des caractéristiques proposées conduit à une dégradation importante des performances. Par exemple, dans la première image, il est clair que l'utilisation de la couleur ou la texture pour calculer la similarité, la pyramide et le désert qui sont visuellement cohérents, sont brisées, la même chose pour le visage humain. Parce que même avec un paramètre de distance correctement choisi, l'utilisation d'une seule caractéristique, la couleur ou la texture, brise les régularités à l'intérieur de l'objet et conduit à une sur-segmentation dans l'image. En revanche, la combinaison de ces deux caractéristiques conduit à de meilleures performances visuelles et préserve bien ces régularités.

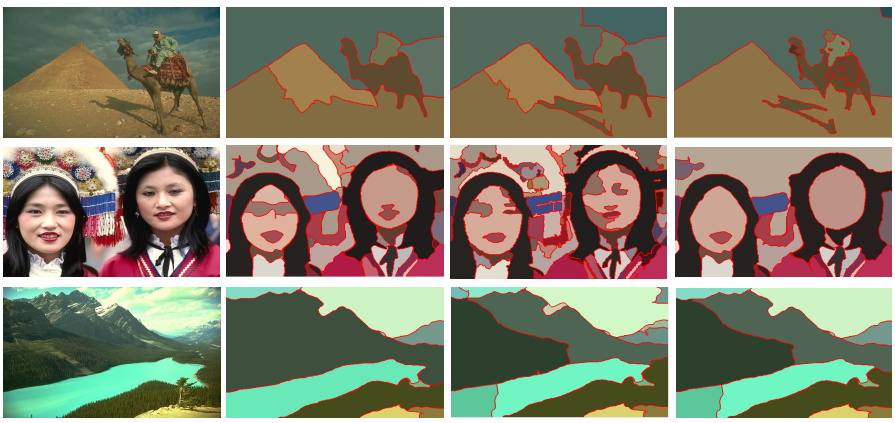

FIGURE 4.4 – Comparaison des résultats de la segmentation : a) Image originale ; b) La caractéristique HOG ; c) La caractéristique LAB ; d) HOG avec LAB.

4.8.2.2 Influence des méthodes de détection des communautés

Dans cette section, nous comparons les méthodes de détection des communautés proposées afin de choisir les meilleurs d'entre elles pour la comparaison suivante avec des méthodes alternatives. Dans la première expérimentation d'évaluations qualitatives, illustrée

dans les Figure 4.5, 4.6, 4.7 et 4.8, pour chaque catégorie Animaux, Personnes, Paysage naturel et Paysage urbain, nous testons le framework proposé pour chaque méthode de détection de communauté. Les résultats produisent des régions importantes et donnent des meilleurs résultats pour toutes les images sélectionnées pour chaque catégorie, par exemple, le visage humain dans la catégorie des personnes, les animaux dans la catégorie des paysages naturels, le château dans la catégorie des paysages urbains et les montagnes dans la catégorie des paysages naturels. Comme indiqué dans les figures, la méthode Infomap ne donne pas toujours la meilleure segmentation et sous-estime le nombre des communautés même si elle est plus rapide que les autres méthodes de détection des communautés. De plus, on peut observer sur les figures que la méthode FMCDRN donne la meilleure segmentation de l'image. Nous évaluons également la performance du framework proposé avec les quatre techniques de segmentation de manière quantitative. Les tableaux 4.2, 4.3, 4.4 et 4.5 présentent les valeurs moyennes des PRI, VOI, Précision et Rappel pour chaque catégorie. Là encore, les résultats de la méthode FMCDRN montrent leur efficacité pour la tâche de segmentation d'image en termes de PRI/VOI/Précision/Rappel/F-mesure.

TABLEAU 4.2 – Comparaison quantitative entre les méthodes de détection des communautés utilisées dans le framework proposé pour la catégorie **animaux**.

Algorithmes	PRI	VOI	Précision	Rappel	F-measure	Temps d'exécution (secondes)
framework+FMCDRN	**0.911**	**1.520**	**0.897**	**0.789**	**0.839**	4.324
framework+FGMDO	0.858	1.585	0.868	0.681	0.763	8.583
framework+Louvain	0.847	1.610	0.857	0.670	0.752	5.322
framework+Infomap	0.588	2.402	0.598	0.546	0.570	2.161

4.8.2.3 Comparaison avec les méthodes de l'état de l'art

En premier temps, nous faisons une comparaison qualitative du framework proposé avec certaines méthodes de segmentation bien connues dans la littérature : (Li et Wu, 2014), (Abin *et al.*, 2014), CTM (Yang *et al.*, 2008) et EDISON (Christoudias *et al.*, 2002). Nous avons choisi l'algorithme FMCDRN de détection des communautés pour le framework proposé, car il donne la meilleure segmentation d'image, comme indiqué

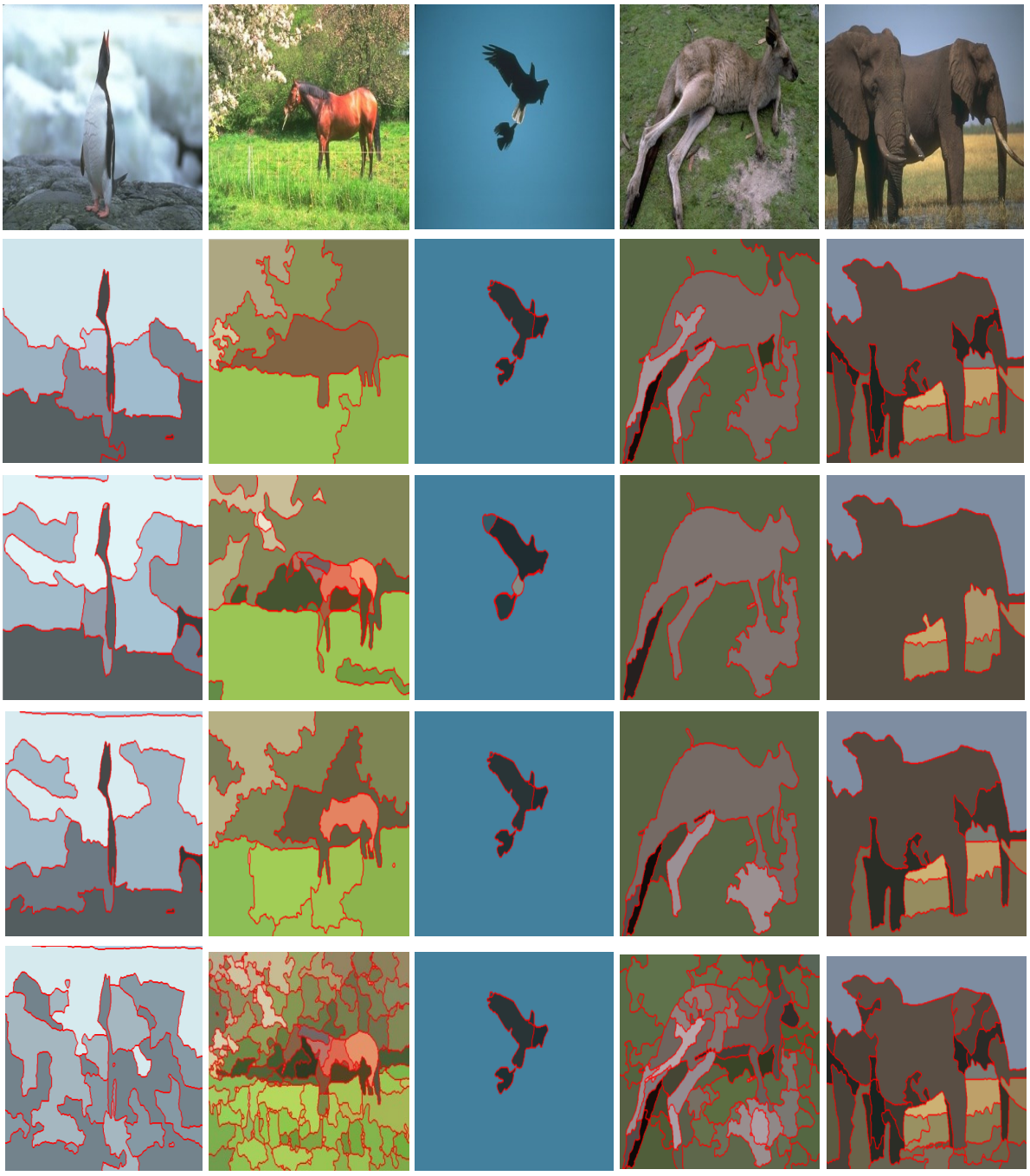

FIGURE 4.5 – Les résultats de segmentation pour le framework proposé avec les méthodes de détection des communautés, pour les images de la catégorie **animaux**, Ligne 1 :Original image ; Ligne 2 : algorithme FMCDRN ; Ligne 3 : algorithme FGMDO ; Ligne 4 : Louvain ; Ligne 5 : Infomap.

FIGURE 4.6 – Les résultats de segmentation pour le framework proposé avec les méthodes de détection des communautés, pour les images de la catégorie **personnes**, Ligne 1 :Original image ; Ligne 2 : algorithme FMCDRN ; Ligne 3 : algorithme FGMDO ; Ligne 4 : Louvain ; Ligne 5 : Infomap.

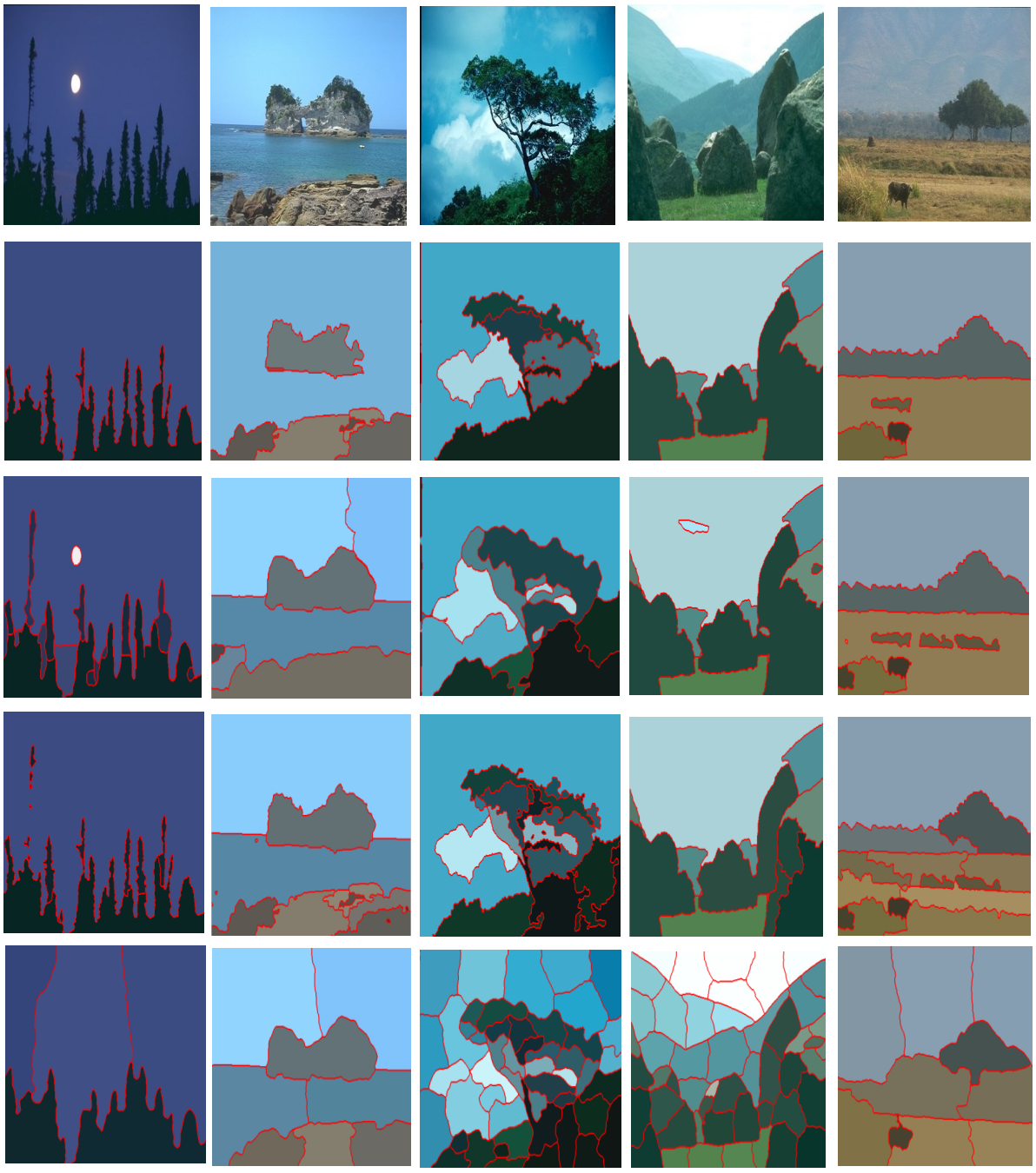

FIGURE 4.7 – Les résultats de segmentation pour le framework proposé avec les méthodes de détection des communautés, pour les images de la catégorie **scènes de la nature**, Ligne 1 :Original image ; Ligne 2 : algorithme FMCDRN ; Ligne 3 : algorithme FGMDO ; Ligne 4 : Louvain ; Ligne 5 : Infomap.

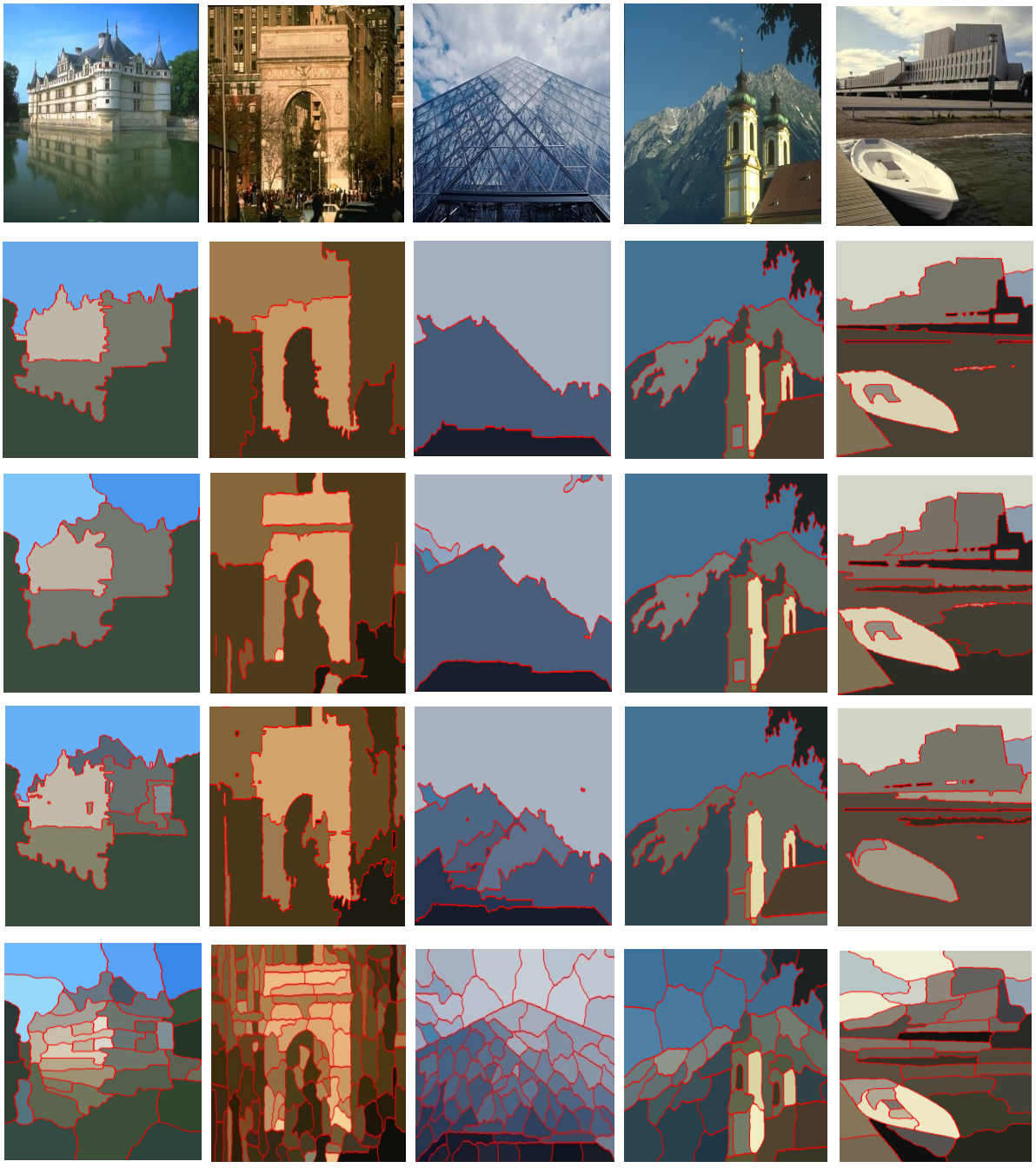

FIGURE 4.8 – Les résultats de segmentation pour le framework proposé avec les méthodes de détection des communautés, pour les images de la catégorie **scènes urbaines**, Ligne 1 :Original image ; Ligne 2 : algorithme FMCDRN ; Ligne 3 : algorithme FGMDO ; Ligne 4 : Louvain ; Ligne 5 : Infomap.

TABLEAU 4.3 – Comparaison quantitative entre les méthodes de détection des communautés utilisées dans le framework proposé pour la catégorie **personnes**.

Algorithmes	PRI	VOI	Précision	Rappel	F-measure	Temps d'exécution (secondes)
framework+FMCDRN	**0.921**	**1.510**	**0.901**	**0.887**	**0.893**	4.525
framework+FGMDO	0.868	1.543	0.878	0.696	0.776	8.842
framework+Louvain	0.849	1.598	0.864	0.684	0.763	5.762
framework+Infomap	0.562	2.454	0.558	0.539	0.548	2.258

TABLEAU 4.4 – Comparaison quantitative entre les méthodes de détection des communautés utilisées dans le framework proposé pour la catégorie **scènes de la nature**.

Algorithmes	PRI	VOI	Précision	Rappel	F-measure	Temps d'exécution (secondes)
framework+FMCDRN	**0.887**	**1.610**	**0.891**	**0.837**	**0.863**	4.897
framework+FGMDO	0.849	1.643	0.862	0.674	0.756	8.984
framework+Louvain	0.828	1.648	0.856	0.671	0.754	5.954
framework+Infomap	0.577	2.421	0.579	0.564	0.571	2.742

TABLEAU 4.5 – Comparaison quantitative entre les méthodes de détection des communautés utilisées dans le framework proposé pour la catégorie **scènes urbaines**.

Algorithmes	PRI	VOI	Précision	Rappel	F-measure	Temps d'exécution (secondes)
framework+FMCDRN	**0.887**	**1.610**	**0.891**	**0.837**	**0.863**	4.624
framework+FGMDO	0.789	1.697	0.782	0.621	0.692	8.725
framework+Louvain	0.783	1.698	0.779	0.612	0.685	5.689
framework+Infomap	0.574	2.419	0.585	0.556	0.570	2.521

précédemment. La Figure 4.9 montre que CTM, EDISON donnent des résultats de segmentation résultant de nombreuses petites régions, ainsi il y a une brise des informations et des régularités dans certaines régions homogènes de l'image, par rapport au framework proposé, qui préserve les informations et les régularités dans l'image segmentée. De plus,

le framework proposé produit des régions homogènes considérables, avec des meilleures performances comme (Li et Wu, 2014), (Abin *et al.*, 2014). Les résultats indiquent la supériorité du framework proposé par rapport aux autres méthodes. Nous calculons ensuite les valeurs moyennes des PRI, VOI, Précision et Rappel pour toutes les images. Comme le montre le tableau 4.6, le framework donne une valeur élevée et de meilleurs résultats par rapport à tous les méthodes de la littérature EDISON, CTM, (Li et Wu, 2014), (Abin *et al.*, 2014) en termes de PRI / VOI, et il a également une performance proche de la perception visuelle humaine avec PRI = 0,828 et VOI = 1,695. De plus, les mesures Précision, Rappel et F-measure du framework proposé avec l'algorithme FMCDRN obtiennent les valeurs les plus élevées avec Précision = 0.788, Rappel = 0.621 et F-measure = 0.694 par rapport aux autres algorithmes, qui indiquent que la plupart de nos segmentations ont des segmentations cohérentes avec la segmentation de la vérité de terrain dans BSDS500. En conclusion, nous pouvons dire que le cadre proposé offre de meilleures performances en termes de précision, de rappel et de mesure F par rapport aux autres algorithmes de l'état de l'art.

TABLEAU 4.6 – Comparaison quantitative entre les différents algorithmes sur toutes les images de la base de données BSDS500.

Algorithmes	PRI	VOI	Précision	Rappel	F-measure
Humain	0.870	1.160	0.910	0.720	0.797
EDISON (Christoudias *et al.*, 2002)	0.786	2.002	0.728	0.524	0.609
CTM (Yang *et al.*, 2008)	0.735	1.978	0.700	0.531	0.603
(Abin *et al.*, 2014)	0.813	1.721	0.764	0.615	0.681
(Li et Wu, 2014)	0.777	1.879	0.733	0.508	0.600
framework+FMCDRN	**0.828**	**1.695**	**0.788**	**0.621**	**0.694**

4.8.3 Temps d'exécution

Nous comparons le temps d'exécution entre le framework proposé avec l'algorithme FMCDRN et les méthodes de la littérature : (Li et Wu, 2014), (Abin *et al.*, 2014). Chaque algorithme est testé sur plus de 100 images de validation de la base de données BSDS500, puis pour chaque étape (segmentation initiale, génération de graphe et détection de communauté), nous calculons la durée moyenne d'exécution. Le tableau 4.7 montre que le

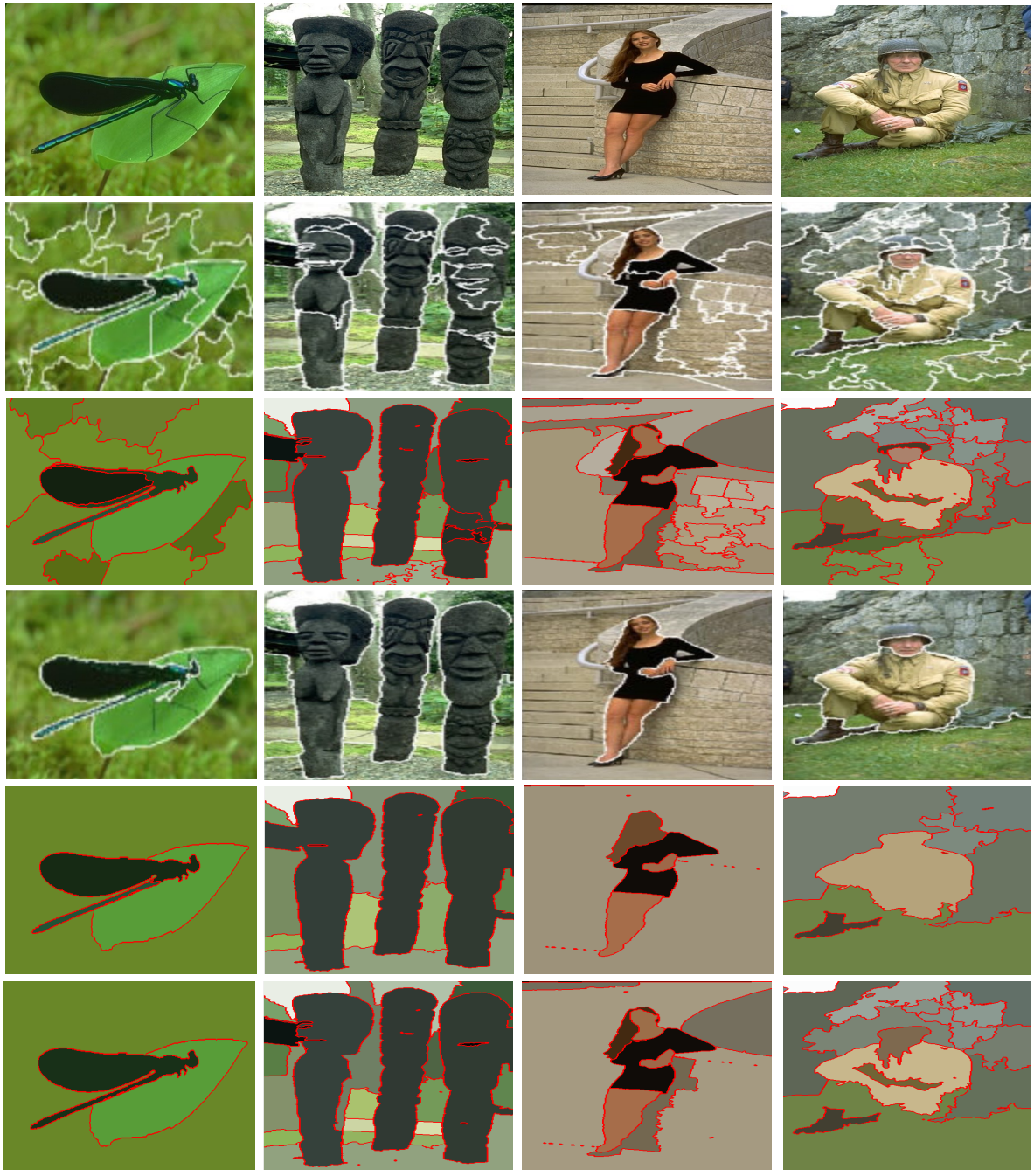

FIGURE 4.9 – Comparaison des resultats de tous les algorithmes, Ligne 1 :Image originale ; Ligne 2 :EDISON ; Ligne 3 : CTM ; Ligne 4 : (Abin *et al.*, 2014) ; Ligne 5 : (Li et Wu, 2014) ; Ligne 6 : le framework proposé + l'algorithme FMCDRN.

framework proposé est toujours plus rapide que (Li et Wu, 2014), (Abin *et al.*, 2014), environ 2,5 fois plus vite que (Abin *et al.*, 2014), et 5 fois plus rapide que (Li et Wu, 2014). En conclusion, on peut dire que le framework proposé donne de meilleurs résultats avec un temps de traitement plus court que d'autres techniques de l'état de l'art.

TABLEAU 4.7 – Temps d'exécution obtenu dans la segmentation pour chaque méthode (en unité de seconde).

Algorithmes	Temps d'exécution				
	Super-pixels	MeanShift	Graphe	Algorithme de détection des communautés	Totale
(Abin *et al.*, 2014)	-	0.239	0.128	11.215	11.482
(Li et Wu, 2014)	3.299	-	0.231	16,870	20.40
framework+FMCDRN	-	0.239	0.178	4.026	**4.443**

4.9 Conclusion

Dans ce chapitre nous avons proposé un framework pour la segmentation des images qui prend en considération les propriétés inhérentes des images et de l'optimisation des mesures : modularité/stabilité. Des algorithmes de détection des communautés ont été utilisés pour optimiser la modularité/stabilité, tels que FMCDRN, FGMDO et Louvain. Tous ces algorithmes peuvent détecter automatiquement le nombre des régions dans l'image. En utilisant à la fois les caractéristiques de la texture (HOG) et de la couleur (LAB), la matrice de similarité est construite de manière adaptative entre les différentes régions de l'image en optimisant la modularité/stabilité et en fusionnant les régions adjacentes de manière itérative. Si aucun changement n'est remarqué dans la structure communautaire lorsque nous appliquons des algorithmes de détection de communauté, la segmentation optimale est atteinte. Les expérimentations ont montré que le framework proposé donne le meilleur résultat de segmentation qualitative, comme le montrent les résultats de la section précédente, et réalise la meilleure performance quantitativement en comparant à toutes les méthodes de la littérature, en termes de PRI, VOI, Précision et Rappel. Puisque, le framework proposé est basé sur trois algorithmes de détection des communautés efficaces, il évite le problème d'avoir nombreuses petites régions dans l'image segmentée et préserve les informations et les régularités dans les objets de l'image. En outre, il fonctionne de manière cohérente et plus rapidement que les algorithmes de l'état de l'art.

Deuxième partie

Analyse des films

Chapitre

5 État de l'art

Sommaire

5.1 Introduction

L'arrivée et la diffusion de la vidéo numérique a orienté les systèmes d'analyse des images vers l'analyse des séquences d'images, qui sont très souvent associées à une bande de son pour former des documents vidéo. Bien entendu, ces informations (image, audio, texte) ne sont pas indépendantes les uns des autres, ce qui signifie qu'un certain nombre de liens sémantiques existent entre elles. Ces liens sont des indices importants qui peuvent aider à comprendre une histoire dans le cas des films, auxquels nous allons nous intéresser plus spécifiquement dans la suite. Un film est une oeuvre produite par un auteur dans un environnement de production, et se distingue donc des autres vidéos, comme les vidéos de surveillance. Pour concrétiser cette idée, l'auteur exploite généralement plusieurs modalités :

- La modalité visuelle : c'est tout ce qui est naturellement ou artificiellement créé et que le spectateur peut constater (ex. images, scènes, *etc*).

- La modalité audio : c'est tout ce que le spectateur peut entendre, comme la parole, la musique, les sons ambiants.

- La modalité textuelle : c'est tout ce que le spectateur peut lire, par exemple le texte qui se superpose aux images, ou bien des textes qui parlent du film (résumé, script, sous titrage, *etc*).

Dans la suite de ce travail de thèse, nous nous focaliserons seulement sur le deux modalité visuelle et textuelle pour analyser l'histoire des films. Différentes approches ont été utilisées pour ce but, dans ce chapitre, nous allons présenter les plus connues dans la littérature, mais surtout on va mettre l'accent sur les approches fondées sur les graphes.

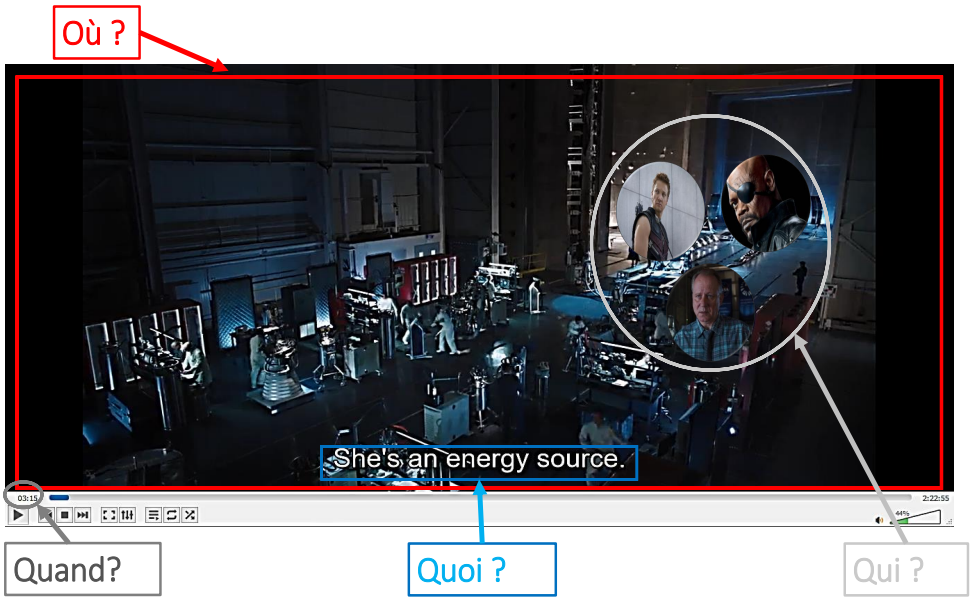

FIGURE 5.1 – L'investigation d'une histoire du film

Parmi les techniques les plus connues pour analyser le contenu d'une vidéo, plus particulièrement un film, c'est la technique abstraction d'une vidéo, c'est une méthode qui permet de résumer le contenu d'un film et de le représenter d'une manière compacte. Généralement, dans la littérature, il existe deux approches pour l'abstraction d'une vidéo : faire le résumé d'une vidéo (en anglais video summarization), ou bien écrémage une vidéo (en anglais video skimming). Récemment une nouvelle approche pour analyser les films a été proposée, et qui est devenue un outil puissant pour capturer les composants qui

construisent une histoire dans un film, c'est l'approche modélisation par graphes. Faire un analyse d'une vidéo, pour le cas d'un résumé, est un processus qui sélectionne un ensemble d'images saillantes appelées images clés (en anglais keyframes) pour représenter le contenu vidéo. Pour le cas d'écrémage d'une vidéo, la vidéo d'origine est représentée sous la forme d'un court clip vidéo. Tandis que la modélisation par graphes met en relation les différentes entités qui constituent une vidéo en interaction pour faire une analyse plus sémantique. Nous allons présenter les principes de chaque approche ci-dessous.

5.2 Approches résumé

Le résumé du film est l'un des techniques clés pour obtenir une représentation courte et précise des données vidéo volumineuses. Un résumé du film est également utile pour le producteur afin de promouvoir le film, et aussi pour le public, pour capturer le thème du film avant de regarder tout son contenu. Le résumé du film a pour objectif de sélectionner les parties du film qui attirent le plus d'attention pour le public. Néanmoins, définir les segments intéressants du film et comment les intégrer efficacement dans un résumé est une tâche subjective, qui reste toujours une question ouverte et mérite une étude plus approfondie. Les travaux les plus récents sur le résumé du film, reposent principalement sur le contenu de la vidéo. Le résumé est généré en extrayant des caractéristiques audiovisuelles de bas ou moyen niveau, soit pour identifier le frame clé - une image dans un interval du temps (le contenu saillant) (Evangelopoulos *et al.*, 2008; Hua *et al.*, 2005), soit pour explorer la structure du film (Ngo *et al.*, 2005). Evangelopoulos *et al.* (2008) présentent une méthode basée sur la saillance pour détecter des segments audiovisuels importants dans le film. Ils se sont concentrés sur les avantages potentiels de la modélisation de l'attention, en se basant sur les fonctionnalités et de l'intégration de signaux multisensoriels. Un algorithme de résumé vidéo est développé sur la base des segments saillants. Hua *et al.* (2005) proposent un framework générique de modèle d'attention de l'utilisateur qui constitue une alternative à l'analyse du contenu de la vidéo, y compris la structuration, l'indexation et le filtrage. En outre, un ensemble de méthodes de modélisation de l'attention visuelle et auditive ont été proposées

Ngo *et al.* (2005) proposent une approche unifiée pour le résumé des vidéos basée sur l'analyse des structures des vidéos et des extraits d'une vidéo. La modélisation par scène et la détection des saillances sont les deux composantes principales de leur approche. La modélisation par scène est obtenue par un algorithme de coupe normalisé et une analyse d'un graphe temporel, tandis que la détection des saillance est réalisée par une modélisation d'attention du mouvement (en anglais motion attention). Dans leur approche proposée, une vidéo est représentée par un graphe complet non orienté et l'algorithme de coupe normalisée est appliqué pour partitionner de manière globale et optimale le graphe en clusters de vidéos. Les clusters résultants forment un graphe temporel dirigé et l'algorithme du plus court chemin est proposé pour détecter efficacement les scènes de la vidéo. Les valeurs d'attention sont ensuite calculées et attachées aux scènes, clusters, shots et sub-shots dans le graphe temporel.

5.3 Approches écrémage

Le système VAbstract (Pfeiffer *et al.*, 1996) est probablement le premier système d'écrémage de film qui identifie les segments caractéristiques d'une vidéo. Ces segments contiennent des acteurs principaux, des dialogues, des évènements et des explosions pour former une bande-annonce de film. Dans ce système, un film a tout d'abord été partitionné en segments de même longueur, puis une scène (telle que celle avec dialogue, mouvement intense ou contraste élevé) a été extraite de chaque segment, sauf la dernière partie du film. Enfin, toutes les scènes sélectionnées ont été organisées dans leur ordre temporel d'origine afin de réduire la possibilité d'une erreur de changement du contexte. Luo et Fan (2004) ont proposé une méthode qui identifie en premier lieu les objets saillants et cartographie les principales prises de vue de la vidéo de certains concepts médicaux. Ensuite, chaque *shot* principale a été pondérée en fonction de sa structure (les éléments du *shot*), du concept médical attribué (par exemple, une opération), des objets saillants contenus et la longueur. Finalement, l'écrémage a été formé en sélectionnant les prises de vue ayant le poids le plus élevé à celui ayant le poids le plus bas jusqu'à ce que leur longueur totale atteigne la longueur prévue. Une façon de déterminer les parties importantes dans une vidéo, consiste à exploiter les modèles d'attention de l'utilisateur, comme indiqué dans

(Ma *et al.*, 2002). Divers modèles d'attention visuelle ont été conçus pour capturer des caractéristiques telles que l'attention aux mouvements, au visage et à la caméra. Par exemple, le modèle d'attention de mouvement a été utilisé pour capturer le mouvement humain, tandis que le modèle d'attention statique a été utilisé pour mesurer l'attention sur une région d'arrière-plan statique. Plusieurs modèles impliquant des mouvements de la caméra ont également été étudiés et utilisés pour construire le modèle d'attention de caméra. De plus, deux modèles d'attention audio (c'est-à-dire, le modèle de saillance audio et le modèle de discours/musique) ont été adoptés. Toutes ces informations d'attention extraites ont été exploitées pour identifier les prises de vue importantes constituant le résumé final de la vidéo. Une autre idée, présentée dans (Sundaram et Chang, 2001), était de segmenter la vidéo en scènes présentant une cohérence de chromaticité, d'éclairage et de son. La complexité de Kolmogorov d'une prise de vidéo qui donne le temps minimum nécessaire à sa compréhension a été mesurée. Enfin, les parties de début et de fin d'une scène ont été sélectionnées pour former l'écrémage, car elles contenaient la plupart des informations essentielles selon le contenu du film.

5.4 Approches modélisation par graphes

La modélisation par graphes, met en relation les différentes entités qui constituent une histoire, ce qui en fait naturellement un outil puissant pour capturer les éléments articulés dans les récits. Cette modélisation par graphes a été appliquée à de nombreux types d'histoires, commençant par les histoires écrites dans les récits, dans les informations provenant des journaux et de la télévision, dans des séries télévisées et finalement dans les films qui sont l'objectif de ce mémoire. Comme il est toujours difficile d'obtenir toutes les informations nécessaires pour analyser un film à partir du contenu de la vidéo lui-même (Demirkesen et Cherifi, 2008; Pastrana-Vidal *et al.*, 2006), nous pouvons nous fier à des sources supplémentaires comme le script du film, les sous-titres, et finalement la vidéo elle-même. Des graphes ont été explorés pour analyser le contenu de la vidéo (Renoust *et al.*, 2014) et se sont avérés efficaces pour l'analyse de sujets et de concepts (Kadushin, 2012). Les graphes d'acteurs en particulier ont été largement analysés dans la littérature (Waumans *et al.*, 2015), dans des vidéos d'actualités télévisées (Renoust *et al.*, 2016a), et

même des sites Web sont dédiés à l'analyse sociale de Game of Thrones (Mish, 2016).

Nombreuses techniques ont été proposées pour analyser les histoires en utilisant la modélisation par graphes. En se basant sur l'analyse de la structure de l'histoire, ces techniques permettent de déduire le sens des interactions entre les objets de l'histoire grâce à une analyse structurelle. Ces approches comportent deux étapes principales : la première est la construction du modèle d'histoire, et la seconde consiste à extraire les informations du modèle. Une fois que le graphe qui relie des entités telles que des personnages, des scènes ou des prises de vue est construit, des informations sémantiques sont extraites à partir de ce réseau.

5.4.1 Méthodes basées sur les graphes de scènes

Ces approches ont proposé des graphes basés sur la segmentation de scènes et des méthodes de détection de scènes pour analyser les histoires des films. Yeung *et al.* (1996) ont proposé une méthode d'analyse utilisant un graphe de transition de scène pour l'exploration et la navigation dans les films (Figure 5.2. Chaque cluster de shots est représenté par un noeud dans le graphe de transition de scène. Un lien est créé entre deux noeuds i et j si le shot représenté par le noeud i précède immédiatement un shot quelconque représenté par le noeud j. Cette approche permet de créer un réseau d'interactions entre les shots, puis de l'analyser afin d'extraire les unités de scénario des scènes.

Corrêa Jr *et al.* (2019) étendent cette approche en construisant une structure narrative pour les documents. Ils connectent un réseau de phrases en fonction de leur similarité sémantique, qui peuvent être utilisées pour caractériser et classer des textes. Jung *et al.* (2004) utilisent un graphe de scènes à structure narrative pour résumer un film. Chaque noeud représente une scène dans le graphe. Les noeuds de scène sont des éléments narratifs avec des interactions de personnages et des connexions entre scènes déterminées par des relations éditoriales. Utiliser uniquement des scènes pour construire un graphe à structure narrative pour l'analyse de films n'est pas suffisant. Les éléments de l'histoire tels que les personnages principaux et leurs interactions ne peuvent pas être récupérés à partir de ces réseaux. Notre travail contraste en utilisant des sources supplémentaires (scripts, sous-titres, *etc.*).

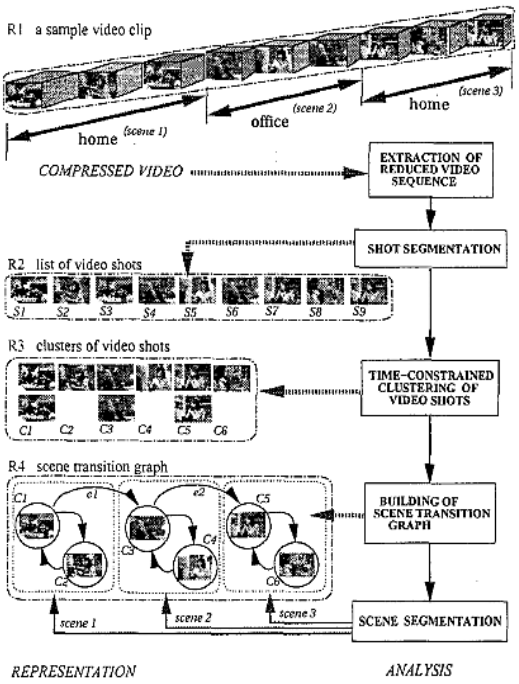

FIGURE 5.2 – Schéma fonctionnel de l'approche proposéé (Yeung *et al.*, 1996).

5.4.2 Méthodes basées sur les graphes de personnages

L'analyse du réseau de personnages est un exercice traditionnel dans l'analyse du réseau social, avec le réseau du roman *Les Misérables*, elle est désormais une classique de la discipline (Knuth, 1993) en inspirant toujours les recherches actuelles. Waumans *et al.* (2015) proposent de créer des réseaux sociaux à partir des dialogues de la série *Harry Potter* (Figure 5.3). Ils ont proposé un nouvel algorithme capable de générer plusieurs types de réseaux (dirigés, non dirigés, pondérés, pondérés à l'aide d'une analyse des sentiments, dynamiques) construits à partir des romans. Le but de leur travail est de définir la signature de l'histoire d'un roman sur la base de l'analyse topologique de son réseau social de personnages.

Chen *et al.* (2019) proposent une approche intégrée pour enquêter sur le réseau social des personnages littéraires en fonction de leurs modèles d'activité dans le roman. Ils utilisent l'algorithme MSC (Minimum Span Clustering) pour identifier la structure communautaire du réseau de personnages, et la visualiser pour finalement calculer les mesures de centralité pour des personnages individuels.

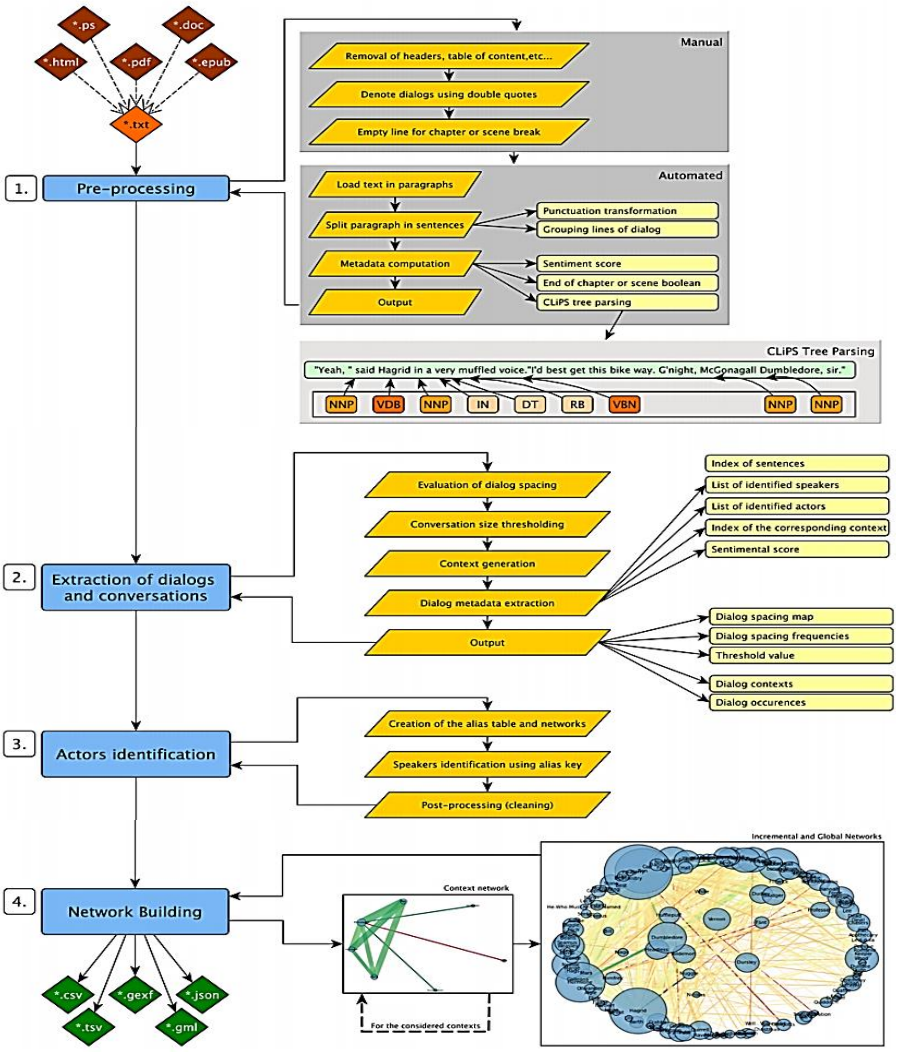

FIGURE 5.3 – Vue schématique de l'ensemble de l'approche proposée (Waumans *et al.*, 2015).

5.4.3 Méthodes basées sur les réseaux sociaux de co-apparition

Les réseaux de co-apparition, qui se connectent lorsque des personnages co-apparaissent à l'écran, ont été un sujet de recherche important, atteignant même les personnages de la populaire série Game of Thrones (Mish, 2016). RoleNet (Weng *et al.*, 2009) est basé sur l'approche SNA (Social Network Analysis) pour analyser les histoires de film. Chaque noeud représente un personnage et les liens représentent des relations de co-apparition dans le réseau social. RoleNet peut identifier automatiquement les rôles principaux et leurs communautés correspondantes en enquêtant sur les interactions sociales entre les personnages.

He *et al.* (2018) étendent la construction des réseaux de co-apparition avec une notion spatio-temporelle. Ils analysent la centralité sociale et la communauté en se basant sur des vérités de terrain créé par l'humain. Tan *et al.* (2014) ont proposé une analyse des réseaux de personnages dans deux séries télévisées de science-fiction. Ces réseaux sont construits sur la base de la co-occurrence de la scène entre les personnages pour indiquer la présence d'une connexion. Les mesures topologiques globales du réseau, telles que la longueur du chemin, la densité du graphe, *etc*, sont calculées et se révèlent similaires entre les deux séries. En outre, divers scores de centralité de noeud sont calculés et utilisés pour interpréter les interactions entre les personnages centraux et le récit général. CoCharNet (Tran et Jung, 2015) utilise un réseau social de co-apparition annoté manuellement sur les six films Star Wars et propose une analyse de centralité. Renoust *et al.* (2016a) proposent une construction automatique de réseaux sociaux politiques à partir de la détection des visages et du suivi des données dans les journaux télévisés (Figure 5.4). La topologie du réseau et l'importance des noeuds (hommes politiques) sont ensuite comparées sur différentes fenêtres temporelles afin de fournir des informations politiques. Notre travail est très inspiré par ces réseaux sociaux de co-apparition, qui fournissent un aperçu intéressant pour les rôles des personnages, mais ils sont encore insuffisants pour placer pleinement les personnages dans une histoire, c'est pourquoi nous nous appuyons sur des indices sémantiques supplémentaires.

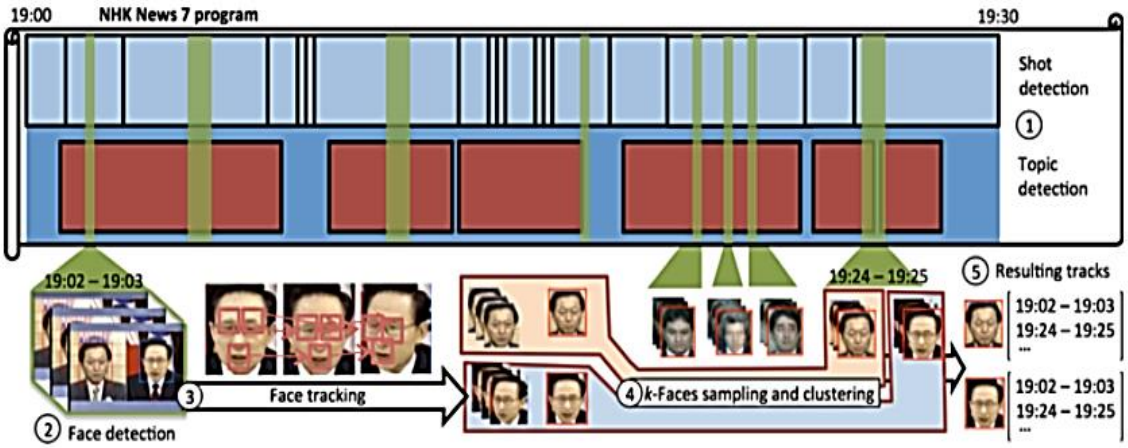

FIGURE 5.4 – Schéma global de l'ensemble des étapes de l'approche proposée (Renoust *et al.*, 2016a).

5.4.4 Méthodes basées sur les réseaux sociaux et sur les dialogues

Les réseaux sociaux dérivés à partir des interactions entre les dialogues dans les scripts de films ont été utilisés pour des différentes fins. Character-net (Park *et al.*, 2012) propose une méthode d'analyse de l'histoire du film via l'analyse de réseau social à l'aide du script de film. Ils construisent un réseau pondéré de personnages à partir des échanges de dialogues afin de classer leur rôle par ordre d'importance. En se basant sur un corpus de scénarios de films Gorinski et Lapata (2018) ont proposé un modèle d'apprentissage end-to-end pour la génération d'un aperçu du film, qui utilise des fonctionnalités basées sur des graphes extraites de réseaux de dialogues et de personnages construits à partir du script de film. Semblables aux réseaux de co-apparition, ces approches utilisent uniquement un réseau social pour l'analyse des films basé sur l'interaction entre les dialogues, ce qui ne peut pas fournir une construction socio-sémantique du contenu de la narration du film. Avec un objectif différent, notre modèle donne un aperçu sémantique de l'histoire du film basé sur des questions.

5.4.5 Méthodes basées sur les réseaux multi-couches

Les approches récentes utilisent des réseaux multiplex pour combiner des indices sémantiques textuels et visuels. StoryRoleNet (Lv *et al.*, 2018) n'est pas proprement une approche multi-couches, mais montre bien l'intérêt de la combinaison multimodale. Il

fournit une construction automatique du réseau d'interaction de personnages et une segmentation de l'histoire en combinant des fonctionnalités visuelles et des sous-titres. Dans les réseaux Visual Clouds (Ren *et al.*, 2018) (Figure 5.5), les réseaux extraits de vidéos d'actualités télévisées sont utilisés comme support de base pour l'affinement de la recherche interactive sur des données hétérogènes. Les couches ne peuvent cependant pas être étudiées individuellement.

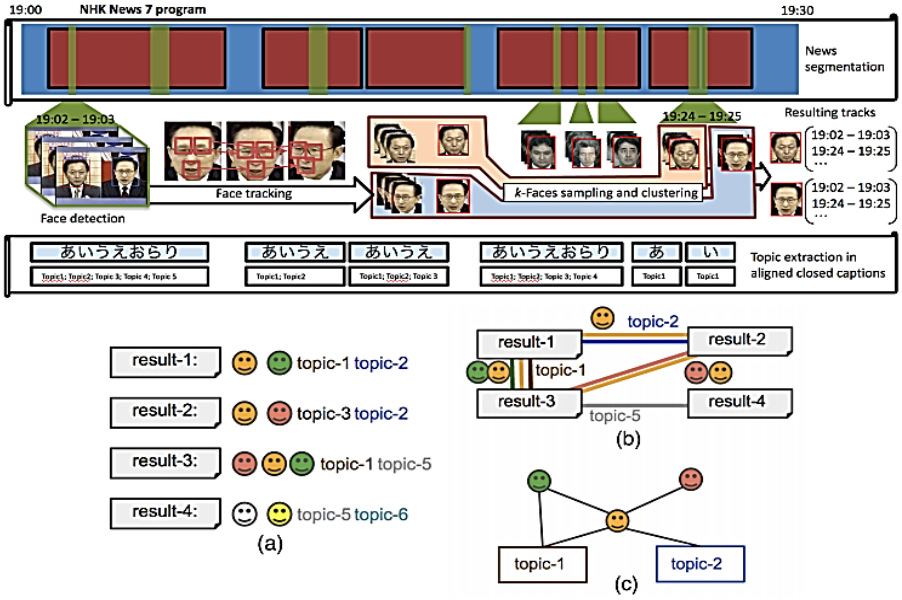

FIGURE 5.5 – Schéma global de l'approche proposée (Ren *et al.*, 2018).

5.5 Conclusion

Dans ce chapitre nous avons présenté diverses méthodes utilisées pour analyser les histoires des films. Plusieurs conclusions peuvent être déduites de cette présentation de différentes méthodes. Les travaux les plus récents sur l'analyse de l'histoire des films reposent principalement sur le contenu vidéo. Le résumé est généré en extrayant des caractéristiques audiovisuelles de bas ou moyen niveau, soit pour identifier le contenu principal, soit pour explorer la structure du film. Toutefois, pour des raisons sémantiques existantes entre les caractéristiques de bas niveau et la compréhension de haut niveau, les caractéristiques audiovisuelles extraites ne peuvent pas caractériser un contenu de film attractif au niveau sémantique et affectif. Pour adresser ces limites, les approches fondées sur les

graphes ont été exploité, pour rassembler les différentes entités qui construisent l'histoire du film, afin de produire une vue globale du contenu du film. Le prochain chapitre, sera consacré pour définir un nouveau modèle basé sur les graphes, pour analyser l'histoire des films.

Chapitre

6 Analyse des films par les réseaux multicouches en utilisant le script

Sommaire

6.1 Introduction

Une propriété de l'oeil humain est de conserver pendant quelques millisecondes l'image projetée d'un objet avant qu'il ne se dissolve. Si une séquence d'images est projetée à plus de 25 images par seconde, les yeux humains ne peuvent pas se rendre compte qu'ils observent une séquence d'images discrètes. Les vidéos et les films utilisent ce principe pour produire la sensation d'images en mouvement. Avec le changement progressif de la consommation audiovisuelle ces dernières années, le nombre de chaînes de télévision, de films et d'autres ressources vidéo augmente rapidement. Parallèlement, avec le développement continu de la technologie d'analyse vidéo, la demande d'accès automatique à des informations telles que le scénario de la vidéo, les personnages, *etc*, augmente également. Récemment, des méthodes et des technologies ont été proposées pour répondre à ces besoins, parmi lesquelles l'analyse du réseau social, qui est une technique efficace pour analyser l'histoire de la vidéo. Les interactions entre les éléments d'une histoire ont souvent été

capturées grâce à la modélisation en réseau (Park *et al.*, 2012; Waumans *et al.*, 2015; Tan *et al.*, 2014; Renoust *et al.*, 2016a,b; Mourchid *et al.*, 2018; Viard et Fournier-S'niehotta, 2018). Il a été utilisé pour soutenir la narration d'un large éventail d'histoires, des livres (Waumans *et al.*, 2015), des séries télévisées (Renoust *et al.*, 2016a), des reportages d'actualité (Tan *et al.*, 2014) et finalement des films (Park *et al.*, 2012; Mourchid *et al.*, 2018), qui sont au centre de nos préoccupations dans cette contribution. Les réseaux sont des objets visuels qui peuvent non seulement être intelligemment visualisés pour expliquer les histoires (Renoust *et al.*, 2016a), mais leur structure peut également être interrogée sous l'angle de la topologie (Waumans *et al.*, 2015; Rital *et al.*, 2005). Ces travaux sont limités. En effet, ils se concentrent principalement sur une seule facette des histoires, principalement les personnages de l'histoire. Quand les journalistes enquêtent sur une histoire, ils articulent souvent les 5 questions : Qui ?, Où ?, Quoi ?, Quand ? et comment/pourquoi ? (Chen *et al.*, 2009; Kipling, 1902). L'analyse des histoires par le réseau tente de répondre à la question Comment/Pourquoi ? en articulant les 4 autres questions dans un réseau. Depuis que le quand ? est fourni avec les données (selon la narration), les travaux précédents ont principalement centrés sur le Qui ? dans un réseau monocouche. Dans ce chapitre, nous souhaitons introduire une méthode automatique qui permet d'exploiter la modalité textuelle pour extraire les réseaux de films à partir des scripts. Nous proposons un nouveau modèle multi-couches basé sur le script du film, permettant d'articuler les personnages, les lieux et les mots-clés. Heureusement, que généralement tous les films sont écrits sous forme de script. Un script est généralement bien structuré et contient tous les composants nécessaires pour analyser automatiquement un film (Jhala, 2008) (scènes, dialogues, personnages, *etc.*). Le réseau multi-couches construit à partir du script capture une structure plus riche d'un film. Il complète l'analyse de réseau mono-couche basée seulement sur les personnages et apporte de nouveaux outils d'analyse topologique (Kivelä *et al.*, 2014).

6.2 Modélisation de l'histoire d'un film par un réseau multi-couches

Dans les disciplines d'investigation, les quatre questions (*Qui ?, Où ?, Quoi ?, Quand ?*) (Kipling, 1902) sont les questions fondamentales qui constituent la formulation pour décrire une histoire complète (Flint, 1917). Déduire comment/pourquoi ? peut être fait en

articulant les autres questions, ce qui en fait des briques essentielles pour analyser une histoire. Dans le contexte de la compréhension du film, nous reformulons les quatre questions comme suit :

- *Qui ?* représente les **personnages** du film ;
- *Où ?* représente les **lieux** où les événements du film se déroulent ;
- *Quoi ?* représente les **mots-clés** qui sont mentionnés dans le film.
- *Quand ?* représente le **temps** guidant la succession d'actions qui se trouvent dans le film.

Sauf le temps, chacune des réponses aux questions (personnages, lieux et mots-clés) forme les entités de base de notre étude.

Notre objectif est d'aider à formuler la compréhension du film en articulant ces quatre éléments. Un modèle de réseau multi-couches est utilisé pour mettre en relation ces éléments afin de former l'histoire du film. Ce graphe est constitué de trois couches de noeuds afin de représenter chaque type d'entité (personnages, lieux et mots-clés) avec de multiples relations entre eux. Nous modélisons deux classes principales de relations : les relations intra-couche entre des noeuds de la même catégorie (par exemple, deux personnes en conversation) ; et les relations inter-couches entre les noeuds de différentes catégories (par exemple, une personne se trouvant à un endroit spécifique). Les multiples familles de noeuds et d'arêtes forment un réseau multi-couches, comme illustré à la Fig.6.1.

Nous définissons maintenant notre graphe multi-couches $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ tel que :

- $V_C \subseteq \mathbb{V}$ représente l'ensemble des personnages $c \in V_C$,
- $V_L \subseteq \mathbb{V}$ représente l'ensemble des lieux $l \in V_L$,
- $V_K \subseteq \mathbb{V}$ représente l'ensemble des mots-clés $k \in V_K$.

Les différentes familles de relations entre les noeuds peuvent être définies comme suit :

- $e \in E_{CC} \subseteq \mathbb{E}$ entre deux personnages tel que $e = (c_i, c_j) \in V_C^2$, quand un personnage $c_i \in V_C$ est en conversation avec un personnage $c_j \in V_C$.

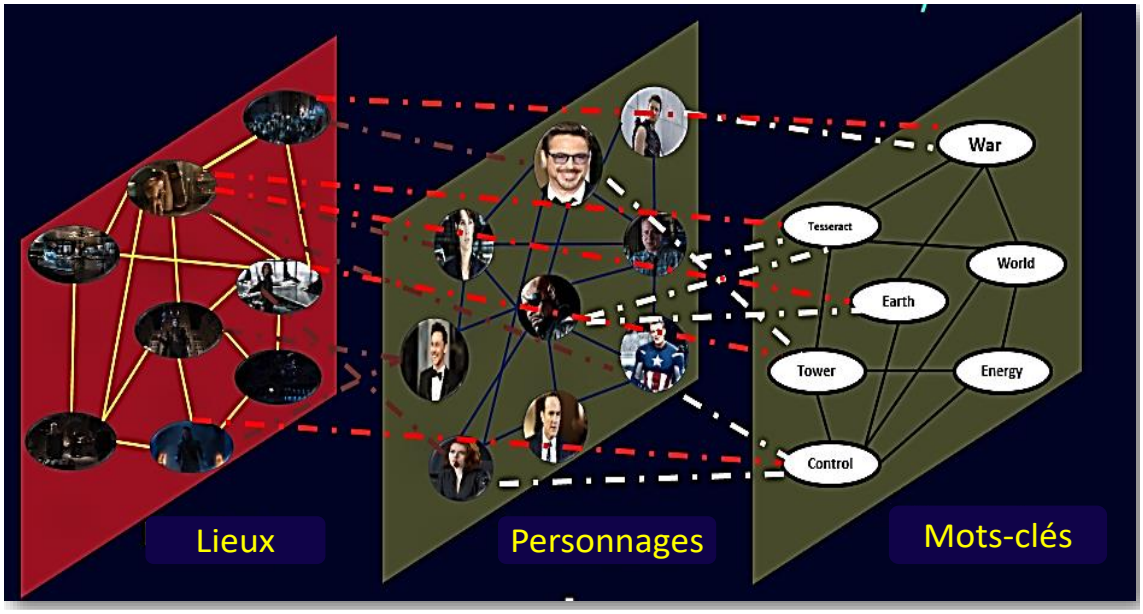

FIGURE 6.1 – Un réseau multi-couches extrait à partir du script du film Avengers, avec les trois couches *Personnage*, *Lieu* et *mots-clés* et leurs interactions

- $e \in E_{LL} \subseteq \mathbb{E}$ entre deux lieux $e = (l_i, l_j) \in V_L^2$, quand il ya une transition temporelle d'un lieu $l_i \in V_L$ vers un autre $l_j \in V_L$.

- $e \in E_{KK} \subseteq \mathbb{E}$ entre les mots-clés $e = (k_i, k_j) \in V_K^2$, quand $k_i \in V_K$ et $k_j \in V_K$ appartiennent à la même phrase.

- $e \in E_{CK} \subseteq \mathbb{E}$ entre un personnage et un mot-clé $e = (c_i, k_j) \in V_C \times V_K$, si le mot-clé $k_j \in V_K$ est prononcé par le personnage $c_i \in V_C$.

- $e \in E_{CL} \subseteq \mathbb{E}$ entre un personnage et un lieu t $e = (c_i, l_j) \in V_C \times V_L$, si le personnage $c_i \in V_C$ est présent dans un lieu $l_j \in V_L$.

- $e \in E_{KL} \subseteq \mathbb{E}$ entre un mot-clé est un lieu $e = (k_i, l_j) \in V_K \times V_K$, quand le mot-clé $k_i \in V_K$ est mentionné dans une conversation qui se déroule dans un lieu $l_j \in V_L$.

Notez que dans cette contribution, nous ne considérons pas la direction et le poids des liens. De plus, comme nous n'avons pas l'intention d'étudier la dynamique du réseau, le temps n'est pas pris en compte. Cependant, le temps supporte tout : l'existence d'un noeud ou d'un lien est définie sur le temps, selon l'ordre des scènes du film.

On peut maintenant se référer aux sous-graphes en ne considérant qu'une couche de liens et son sous-graphe induit :

- $G_{CC} = (V_C, E_{CC}) \subseteq \mathbb{G}$ représente le sous-graphe des interactions entre les personnages ;
- $G_{LL} = (V_L, E_{LL}) \subseteq \mathbb{G}$ représente le sous-graphe des transitions des lieux ;
- $G_{KK} = (V_K, E_{KK}) \subseteq \mathbb{G}$ représente le sous-graphe de co-occurrence des mots-clés ;
- $G_{CK} = (V_C \cup V_K, E_{CK}) \subseteq \mathbb{G}$ représente le sous-graphe des personnages parlant des mots-clés ;
- $G_{CL} = (V_C \cup V_L, E_{CL}) \subseteq \mathbb{G}$ représente le sous-graphe des personnages se trouvant dans des lieux ;
- $G_{KL} = (V_K \cup V_L, E_{KK}) \subseteq \mathbb{G}$ représente le sous-graphe des mots-clés mentionnée dans des lieux.

6.3 Extraction des entités du réseau multi-couches à l'aide de script du film

Nous décrivons maintenant la méthodologie utilisée pour construire le réseau du film. Heureusement, la plupart des longs métrages sont produits à l'aide de scripts très bien structurés (Jhala, 2008). Ils organisent le film en scènes, chacune place des personnages dans des lieux, décrivant leur contexte, leurs actions et, plus important, leurs dialogues. Comme le script met en jeu des nombreuses notions mais qui sont un peu ambiguës, nous donnons d'abord des informations essentielles sur sa structure et quelques définitions importantes. Ensuite, nous présentons la méthodologie utilisée pour extraire les différentes entités avec leurs interactions. Enfin, nous expliquons comment construire le réseau multi-couches en fonction de ces informations.

6.3.1 Structure du script et définitions

Un script détaille tous les éléments d'une histoire : les lieux, les personnages avec leur situation et leurs actions, et surtout les dialogues. Le contenu textuel d'un script suit

souvent un format semi-régulier (Jhala, 2008) tel qu'il est décrite dans la Figure 6.2.

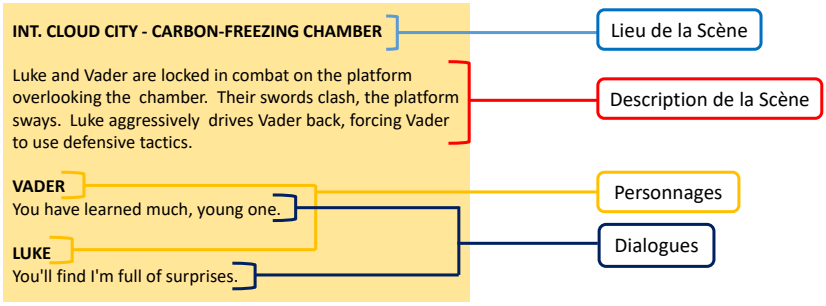

FIGURE 6.2 – Extraits d'un script décrivant une scène du film *The Avengers 2012* (Whedon, 2012), affichant les différents éléments (personnages, dialogues et lieux).

Un script commence généralement par un entête décrivant le lieu et l'heure (Matin, Nuit,...) de la scène. Des mots-clés spécifiques donnent des informations importantes sur la scène (tel que l'intérieur ou l'extérieur de la scène). Les personnages et les objets clés qui sont souvent mis en valeur. Le script suit ensuite une série de dialogues et de descriptions de scènes. Afin de lever toute ambiguïté, nous définissons d'abord le vocabulaire suivant :

- Script : Une source de texte du film qui contient des descriptions sur les scènes, les personnages et les dialogues.
- Scène : partie de script, unité temporelle du film. La collection de toutes les scènes forme le script du film.
- Personnage : Désigne une personne / un animal / une créature qui est présent dans une scène, souvent imité par un acteur.
- Dialogue : Une collection d'énoncés, ce que tous les personnages disent pendant une scène.
- Énoncé : Un bloc de dialogue ininterrompu prononcé par un personnage.
- Conversation : Une série continue d'énoncés entre deux personnages.
- Description : Un bloc de script qui décrit ce qui se passe dans la scène.
- Lieu : Où une scène a-t-elle lieu ou est-elle mentionnée par un personnage.

- mot-clé : La plupart des informations pertinentes tirées d'un énoncé, souvent représentatif de son sujet.

6.3.2 Prétraitement du script

La figure 7.3 illustre le pipeline de traitement du script. Bien que ce processus dépende de la langue, nous limitons notre attention dans ce travail, aux scripts écrits en anglais. Quoi qu'il en soit, le contenu du script suit généralement un format semi-régulier, comme indiqué dans la Fig.6.2.

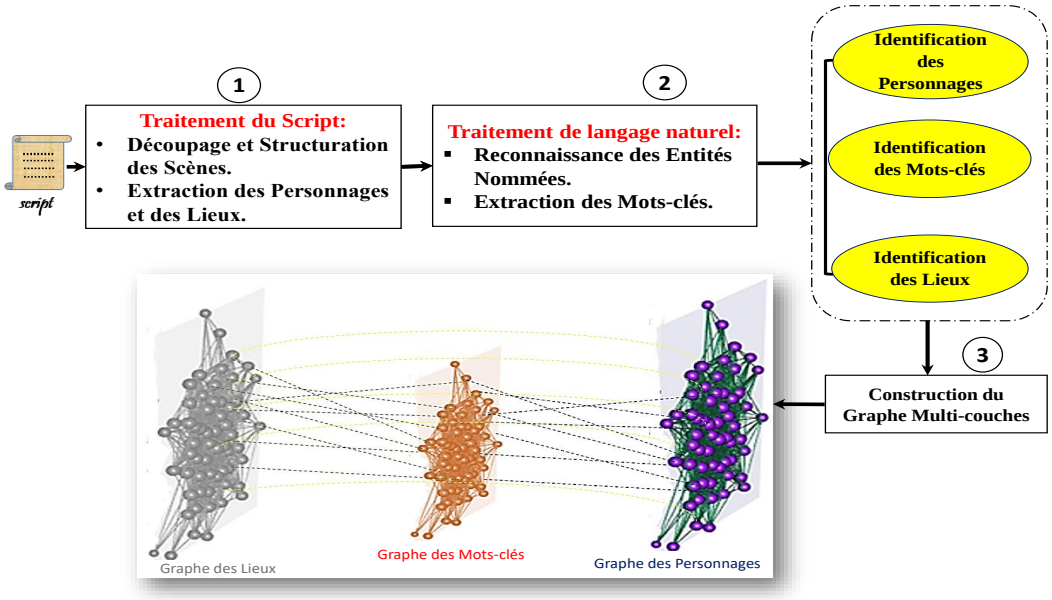

FIGURE 6.3 – Schéma global du processus de construction du modèle proposé.

La première tâche consiste à découper le script en scènes. En effet, ils constituent les principales subdivisions d'un film et, par conséquent, la principale unité d'analyse. Chaque scène est associée à un ensemble (actions, emplacement et personnages). Une scène commence par une ligne de description écrite en majuscules, qui établit le contexte physique de l'action qui suit. Il commence par indiquer si une scène a lieu à l'intérieur ou à l'extérieur (*INT* ou *EXT*), le nom du lieu et peut éventuellement spécifier l'heure (par exemple, *MATIN* ou *NUIT*). Ce sont des marqueurs que nous détectons pour diviser le script en scènes. Les scènes sont attachées à des lieux toujours inclus dans l'entête de la scène, que nous pouvons facilement analyser. Les personnes importantes et les objets

clés sont généralement mis en évidence en lettres majuscules que nous récoltons lors de l'analyse du texte. Le reste de la scène est constitué de dialogues et de descriptions. Les noms des personnages et leurs actions sont toujours représentés avant les lignes de dialogue. Un retrait de ligne aide également à identifier les personnages et les parties de dialogue et aussi la description de la scène.

6.3.3 Prétraitement du Texte

L'étape suivante consiste à traiter le contenu textuel déjà associé à un lieu défini ou à un personnage. Pour ce faire, nous utilisons des outils de traitement du langage naturel pour extraire des entités nommées et des mots-clés.

6.3.3.1 Reconnaissance des entités nommées

La reconnaissance des entités nommées (en anglais Named Entity Recognition NER) désigne les mots importants identifiés dans un contenu textuel (tels que des personnes, des organisations, des villes, *etc.*). Nous utilisons la bibliothèque spaCy (Al Omran et Treude, 2017) pour son efficacité. Nous appliquons le NER aux blocs de description de la scène et en éliminant les entités non pertinentes, telles que les quantités, les ordinaux, l'argent *etc.* en raison que nombreux mots peuvent finir par être mal étiquetés (notamment en raison du contexte ambigu d'un film de science-fiction), nous vérifions manuellement la liste de mots résultante. Trois catégories ont été prises en considération dans ce travail : personnages, lieux et mots-clés. Nous assignons une classe unique pour les noms ambigus faisant référence au même concept (*ex.* $\{TONY, STARK, IRONMAN\} \rightarrow TONYSTARK$). NER nous aide principalement à identifier les personnages présents sur une scène qui sont prononcé dans des énoncés lors des conversations.

6.3.3.2 Extraction des mots-clés

Les mots-clés sont identifiés à partir des dialogues. Nous mesurons la pertinence des mots clés à l'aide de trois méthodes : TF-IDF (Li *et al.*, 2007), LDA (Blei *et al.*, 2003) et Word2Vec (Yuepeng *et al.*, 2015). Les textes de script sont constitués de phrases courtes (même plus courtes après la suppression des mots de liaisons). Word2Vec et TF-IDF pro-

duisent donc peu de mots ou trop de mots sans contenu sémantique. Par conséquent, nous nous appuyons principalement sur LDA, qui apporte le meilleur compromis en supprimant les mots-clés moins sémantiques (tels que *can*, *have*, *etc.*).

6.3.4 Construction du réseau multi-couches

À la suite de deux étapes précédentes, pour chaque scène, on identifié lieu et leur description, les personnages, les énoncés, ainsi que les mots-clés extraits des descriptions et des énoncés. Nous utilisons maintenant cette information pour construire le réseau multi-couches. Reprenons nos questions d'investigation dans le contexte d'une scène : *Qui est présent dans une scène ?* peut être abordé par des personnages, *Où se passe une scène ?* est identifié par les lieux et *Qu'est-ce qui se passe dans une scène ?* est identifié par des mots clés. Ces trois familles d'entités forment naturellement les trois catégories de noeuds, respectivement V_C, V_L et V_K.

Nous souhaitons maintenant mettre les relations que nous avons décrites dans la section 6.2. Deux personnages c_i, c_j peuvent être en relation lorsqu'ils participent à une même conversation, formant ainsi une arête $e_{c_i,c_j} \in E_{CC}$. Nous connectons deux lieux $e_{l_i,l_j} \in E_{LL}$ lorsqu'il existe une transition temporelle entre eux l_i et l_j, *ie* à la suite de la succession de deux scènes. Les mots-clés k_i, k_j qui se trouvent dans une même phrase créent une arête $e_{k_i,k_j} \in E_{KK}$.

En utilisant la structure, extraite du script, nous pouvons ajouter des liens supplémentaires entre les catégories. Un lien $e_{c_i,l_j} \in E_{CL}$ associe un personnage c_i à un emplacement l_j lorsque le personnage c_i apparaît dans une scène se déroulant dans un lieu l_j. Lorsqu'un personnage c_i parle dans une conversation, pour chaque mot-clé k_j détecté dans cet énoncé, nous allons créer une arête $e_{c_i,k_j} \in E_{CK}$. Enfin, nous pouvons associer les mots-clés k_i extraits dans une conversation placée dans un lieu l_j pour former le lien $e_{k_i,l_j} \in E_{KL}$. Un graphe résultant combinant toutes les couches est visualisé dans la Fig.6.1.

6.4 Analyse du réseau multi-couches

6.4.1 Données et Méthodes

Nous souhaitons maintenant appliquer notre modèle sur deux films populaires de science-fiction, *Star Wars : épisode IV-Un nouvel espoir* (Lucas, 1977) (SW) et *The Avengers 2012* (Whedon, 2012) (AV). *Star Wars*, 1977, est une icône dans les films de science-fiction. Il raconte l'histoire d'un jeune homme s'éveillant pour la Force (une religion autant qu'une source de pouvoir) dans une bataille épique entre le côté Lumière (le Jedi) et le côté Obscur (l'Empire) qui domine la galaxie. *The Avengers 2012*, met en jeu une équipe de super-héros (les Avengers) de l'univers Marvel contre Loki, un méchant extra-terrestre cherchant à contrôler la terre à l'aide du Tesseract qui est l'une des six Pierres d'infinité. Les scripts sont collectés à partir de (`www.imsdb.com`), c'est une base de données en ligne mise à disposition par les cinéastes. Une fois les graphes sont extraits pour chaque film, nous examinons d'abord leurs propriétés topologiques globales. Ensuite, nous étudions l'*influence* du noeud en utilisant des mesures classiques de centralité (Degree, Betweenness, Eigenvector) et une mesure récemment introduite combinant les caractéristiques locales et globales du noeud appelée Score d'influence (Bioglio et Pensa, 2017) (en anglais Influence Score).

6.4.2 Propriétés topologiques

Les propriétés topologiques de base des réseaux sont présentées dans le tableau 6.1. Notez que les couches des personnages G_{KK} et des lieux G_{LL} sont constitués d'un seul composant connecté. Les couches des mots clés G_{KK} ont quelques noeuds isolés, nous montrons dans ce cas que les statistiques de la composante "Géante". Les réseaux multi-couches des deux films sont assez petits. Les couches des personnages G_{CC} sont très denses par rapport à les couches des lieux et des mot-clés. En effet, presque tous les personnages ont parlé ensemble dans le film où ils sont présents dans la même scène. De plus, toutes les couches ont un diamètre élevé, notamment la couche des lieux. Cela est dû à quelques transitions temporelles entre des lieux peu fréquents qui introduisent de longs chemins. Dans les deux films, le coefficient de clustering est très élevé pour la

couche des personnages. Les couches des mots-clés sont également bien regroupées. Il n'y a pas de triangle dans les interactions inter-couches (comme dans G_{CL}, G_{CK} et G_{KL}) car ce sont des graphes bipartites reliant deux ensembles d'objets. Nous pouvons observer que les réseaux multi-couches sont assortatifs. En effet, les principales entités du film, en particulier les personnages et les mots clés, ont tendance à se connecter. Par exemple, *tesseract* et *Pont de Hellicarrier*, qui sont des mots clés importants, ont tendance à être liés à *Nick Fury*, un personnage principal du film. Les deux réseaux multicouches ont une valeur petite pour le plus court chemin moyen.

	The Avengers 2012							**Star Wars Épisode IV**						
	$\|V\|$	$\|E\|$	ρ	d	C	τ	l_G	$\|V\|$	$\|E\|$	ρ	d	C	τ	l_G
$\mathbb{G}$	187	1391	0,06	5	0,53	0,5	2,42	269	2586	0,06	6	0,56	0,52	2,61
G_{CC}	38	276	0,46	4	0,78	0,18	2.08	62	650	0,32	4	0,84	0,07	2.02
G_{KK}	81	221	0,07	7	0,49	0,05	2.83	74	701	0,08	7	0,57	$2 * 10^{-2}$	3.46
G_{LL}	68	240	0,07	9	0,29	0,03	3.90	133	566	0,05	10	0,24	0,17	3.81
G_{CL}	81	159	0,04	6	0	-0,01	3.36	143	316	0,04	7	0	0,07	3.24
G_{CK}	96	228	0,07	6	0	-0,02	2.78	85	148	0,06	5	0	-0,1	2.91
G_{KL}	115	267	0,04	6	0	-0,02	3.14	96	205	0,06	10	0	-0,02	3.27

TABLEAU 6.1 – Les propriétés topologiques globales des différents réseaux : nombre de noeuds($|V|$), nombre de liens ($|E|$), Densité (ρ), Diamètre (d), Coefficient de Clustering Moyen (C), Coefficient d'Assortativité (τ), et Le Plus Court Chemin Moyen (l_G).

6.4.3 Influence du noeud

Nous présentons dans le tableau 6.2 les cinq premiers noeuds triés selon leur score de centralité calculé dans les trois couches considérées indépendamment pour SW et AV.

6.4.3.1 Classement des personnages

Tous les top personnages du tableau sont les principaux dans SW avec *Luke Skywalker* dominant toutes les centralités. *Vader*, *Leia* et *Han Solo* accompagnent le personnage principal tout au long du film. *Leia* est la princesse sauvée pour mener les rebelles. *Ben* le capitaine du Millennium Falcon, qui collabore avec *C-3PO* pour sauver la belle princesse. *Dark Vador* est le personnage principal vilain. Dans AV, les personnages principaux jouent un rôle important dans le film et dans l'univers Marvel. Notez que, à l'exception de trois personnages, ils sont des super-héros (ayant également leur propre film (y compris celui qui est le plus connu, *Tony Stark* qui produit les films *Iron Man*). *Nick Fury*, le personnage

le plus influent est essentiellement dans le film, c'est la liaison entre tous les super-héros, qui font équipe avec *Tony Stark*, *Captain America* et *Natasha* contre *Loki* le dernier de notre rang, qui est le principal vilain du film.

6.4.3.2 Classement des mots-clés

Dans *SW*, il existe une distribution extrêmement inégale dans la centralité des mots-clés. Dans le top du classement, nous trouvons toujours des mots-clés tels que *Empire*, *Jedi*, *side*. Le nom de *Luke* le personnage principal le plus présent et répété dans le film. Dans *AV*, le mot clé le plus utilisé est *Tesseract*, l'objet qui motive les méchants à attaquer la terre pour la contrôler. Les mots-clés suivants expliquent pourquoi : *world* et *control* accompagnent cet objet dans le tableau, tout simplement, c'est parce que c'est une source puissante pouvant contrôler le monde. *Phil Coulson* est celui qui a envoyé *Nick fury* pour trouver les héros dont il a besoin pour combattre Loki qui a le pouvoir de *Tesseract*.

6.4.3.3 Classement des lieux

Dans *SW*, le premier lieu est le *Space*, où se déroulent presque toutes les scènes. *Surface of the Death Star* - c'est dans laquelle une autre partie majeure d'actions a lieu - et qui la détruit ensuite. Le *Luke's X-wing Fighter Cockpit* est le véhicule de Luke qui l'aide dans la guerre contre le *Darth Vader's Cockpit*. Dans *AV*, les trois principaux lieux sont les endroits où tous les personnages principaux se regroupent ou s'affrontent et dans lesquels se produisent la plupart des actions (notez que les ponts *Helicarrier Bridge* et *Helicarrier Detention* sont deux endroits de le *Quinjet*. C'est un véhicule qui transporte partout les personnages principaux). *Stark Tower* est la maison de *Tony Stark* qui est un autre endroit clé du film. *Sky* est une scène transitoire. *Manhattan* est la ville où le combat commence entre les super-héros et Loki. Nous pouvons observer que, pour les deux films, le classement du score d'*influence* est généralement bien corrélé à la fraction d'apparence des objets par scène (personnages, mots-clés, emplacements).

6.4.3.4 Classement dans les réseaux multi-couches

Le tableau 6.3 indique les scores de centralité calculés dans les réseaux multi-couches. Les top noeuds sont principalement les personnages principaux du film. Cependant, les

	Personnages					Mots-clés					Lieux				
	D	B	Ei	I.S	O	D	B	Ei	I.S	O	D	B	Ei	I.S	O
SW	Luke	Luke	Luke	Luke	89	Empire	Side	Luke	Empire	7	S	S	Sds	S	31
	C-3PO	C-3PO	Han	C-3PO	41	Luke	Empire	Side	Luke	180	Sds	Mfc	Lxfc	Sds	29
	Han	Vader	C-3PO	Han	42	Jedi	Sandpeople	Jedi	Side	61	Lxfc	A	Dvc	Lxfc	36
	Vader	Han	Leia	Vader	33	Side	Luke	System	Long	38	Sads	Dsh	Sads	Sads	26
	Leia	Red leader	Ben	Leia	24	System	Stay	Back	Jedi	11	Mfc	Td	Rlc	Dvc	21
AV	N.Fury	N.Fury	C.America	N.Fury	36	Tesseract	Tesseract	Tesseract	Tesseract	31	Hb	Hb	Hb	Hb	19
	Tony	Banner	Tony	Tony	48	P.Coulson	P.Coulson	P.Coulson	P.Coulson	17	M	Hds	M	Hds	10
	C.America	C.America	N.Fury	C.America	69	World	Tony	World	World	17	St	St	St	St	11
	Natasha	Loki	Natasha	Natasha	36	Tony	World	Tony	Tony	2	Sk	Sk	Sk	Sk	9
	Banner	Natasha	Banner	Loki	35	Control	Control	Control	Control	17	Hds	M	Hds	M	13

TABLEAU 6.2 – Les 5 top noeuds triés selon le score de centralité : Degree (D), Betweenness (B), Eigencentrality (Ei), Score d'Influence (I.S). Occurrence (O) c'est le nombre de scène où les objets (personnages, mots-clés et lieux) apparaissent. Dans la couche des lieux nous avons mis les abréviations suivantes S : *Space* ; A : *Alderaan* ; Dsh : *Death Star Hallway* ; Td : *Tatooine Desert* ; Dvc : *Darth Vader's Cockpit* ; Sds : *Surface of The Death Star* ; Lxfc : *Luke's X-wing Fighter Cockpit* ; Sds : *Space Around The Death Star* ; Mfc : *Millennium Falcon Cockpit* ; Hb : *Helicarrier Bridge* ; M : *Manhattan* ; St : *Sark Tower* ; Sk : *Sky* ; Hds : *Helicarrier Detention Section.*

principaux lieux et mots clés figurent également dans la liste des meilleurs classements. Nous pouvons observer que si le degré de centralité tend à mettre en avant des personnages, les mesures *betweenness* et *eigenvector* ont tendance à mettre en avant des lieux. Pour résumer, ces résultats montrent que même une analyse topologique très succincte des réseaux multi-couches peut nous donner une bonne idée du contenu et l'histoire du film.

AV	D	N.Fury	Natasha	Tony	Banner	C.America	Loki	Thor	P.Coulson	Barton	Tesseract	Hb	Barton
	B	N.Fury	Banner	Natasha	Hb	Loki	Tony	Thor	C.America	Bg	Bl	Tesseract	Bs
	Ei	C.America	Tony	N.Fury	Natasha	Thor	Banner	Loki	P.Coulson	Bl	Bs	Loki	Hb
	I.S	N.Fury	Natasha	Tony	Banner	C.America	Loki	Thor	P.Coulson	Barton	Tesseract	Hb	Barton
SW	D	Luke	C-3PO	Han	Ben	Leia	Vader	Biggs	Tarkin	R.Leader	S	Sds	Lxfc
	B	Luke	C-3PO	S	Han	A	Leia	Vader	Ben	Sds	Biggs	Mfc	Td
	I.S	Luke	C-3PO	Han	Ben	Leia	Vader	Biggs	Tarkin	R.Leader	S	Sds	Lxfc

TABLEAU 6.3 – Les top 12 noeuds triés par le score de centralité dans les réseaux multi-couches. Dans la couche des lieux nous avons mis les abréviations suivantes, Bg : *Brooklyn Gym* ; Bl : *Banner's Lab* ; Bs : *Bridge Street* ; Mfc : *Millenium Falcon Cockpit* ; Td : *Tatooine Desert* ; Tmes : *Tatooine Mos Eisley Street.*

6.5 Conclusion

Dans ce chapitre, nous avons présenté un modèle multi-couches qui relie les *personnages*, les *lieux* et les *mots-clés* dans les films. Ce modèle est beaucoup plus riche en information que les réseaux mono-couche qui se concentrent généralement soit sur les *personnages* ou les *scènes*. Nous avons proposé également une méthodologie automatique

pour extraire les éléments du réseau à partir du script. Nous avons déployé le modèle sur deux films populaires. Pour le moment, le traitement nécessite un nettoyage des résultats. Pour obtenir une extraction de réseau entièrement automatique, nous devons lever toute ambiguïté en incluant des éléments multimédias, tels que la segmentation de scène, le suivi des visages et les informations de sous-titre. Dans cette contribution, nous avons présenté les résultats d'une analyse des réseaux extraits du film. Cette analyse a confirmé l'efficacité du modèle. Notez qu'une analyse topologique plus approfondie peut permettre d'obtenir beaucoup plus d'informations. Pour être le moins ambigu possible, nous n'avons considéré que les interactions primitives faciles à interpréter, mais nous pourrions facilement projeter des points de vue différents. En plus, le modèle peut être étendu pour dériver un réseau de co-occurrences de personnages dans le même lieu, un réseau orienté de conversations ou la mention de personnages, *etc.* Les travaux futurs impliquent de déployer notre outil sur des collections plus grandes, telles que toute la série Star Wars, ou même une plus grande collection de films. Nous prévoyons d'utiliser le modèle de réseau multi-couches pour caractériser les genres ou les réalisateurs de films, et même établir une corrélation avec les carrières d'acteur issues de bases de données publiques telles qu'IMDB. Dans le prochain chapitre, on va exploiter les deux modalités textuelle et visuelle en même temps, pour enrichir le modèle proposé et produire un nouveau modèle plus riche et capture plus d'éléments pour raconter l'histoire du film.

Chapitre

7 Analyse des films par les réseaux multicouches en utilisant le script, les sous-titres et le contenu multimédia.

Sommaire

7.1 Introduction

Comme il a été discuté dans le chapitre précédent. Le processus de la création du film commence d'abord par écrire le script du film, qui est un texte généralement structuré. Les scripts du film assemblent tous les éléments du film de manière temporelle (scènes, dialogues) et mettent en évidence des informations spécifiques telles que les personnages et les détails du film, afin de faciliter la tâche de l'analyse automatique du film (Jhala, 2008; Mourchid *et al.*, 2018). Ces dernières années, les méthodes d'analyse des films ont considérablement amélioré pour mieux comprendre leur contenu (Guo *et al.*, 2016), malgré que certaines tâches restent difficiles, nous pouvons enrichir toutes les approches textuelles avec la détection et la reconnaissance des visages (Jiang et Learned-Miller, 2017; Cao *et al.*, 2018) ou bien avec une description de la scène (Johnson *et al.*, 2016; Yang *et al.*, 2017) (en anglais dense captionning). Dans notre précédent travail (Mourchid *et al.*, 2018), nous avons introduit une analyse de réseau qui se déploie à travers *Qui ?*, *Quoi ?* et *Où ?* extraits

du contenu textuel du script, articulés autour de *Quand ?* qui organise la succession des évènements au fur et à mesure que le déroulement du script. Nous capturons ceux-ci dans ce chapitre en proposant un modèle de réseau multi-couches qui capture la structure et les éléments d'un film de manière plus riche par rapport aux réseaux classiques en exploitant la modalité textuelle et visuelle. Il complète l'analyse du réseau mono-couche basé seulement sur les personnages et propose de nouveaux outils d'analyse topologique (Domenico *et al.*, 2014). Dans cette contribution, nous étendons cette approche dans plusieurs directions :

- A partir d'un seul film, nous étendons notre modèle sur plusieurs films de la même saga (une histoire plus grande composée de plusieurs histoires).
- Nous ne limitons pas notre modèle seulement sur le script, mais nous exploitons également le contenu multimédia en intégrant l'analyse à partir de l'image (via un sous-titrage dense (dense captionning) et une analyse des visages).
- En plus, nous fondons notre modèle sur le formalisme de réseau multicouche proposé par Kivelä (Kivelä *et al.*, 2014), pour articuler des personnages, des lieux et des mots clés à travers des modalités (texte et image).

7.2 Modélisation de l'histoire du film par un réseau multi-couches

Comme il est mentionné dans le chapitre précédent les questions fondamentales qui constituent la formule décrivant une histoire complète ce sont les (Qui ?, Où ?, Quoi ?, Quand ?). Ces questions sont les éléments essentiels pour analyser l'histoire d'un film. Notre objectif est d'aider à formuler la compréhension du film en articulant ces quatre éléments. Dans cette contribution nous proposons un modèle de réseau multi-couches qui enrichit celui qui est proposé dans le chapitre précédent en exploitant deux couches supplémentaires, des visages et des captions. Dans le contexte de la compréhension du film, nous formulons les 4 questions comme suit :

- *Qui ?* fait référence aux **personnages** et aux **visages** qui sont présents dans un film ;

- *Où ?* fait référence aux **lieux** où les actions d'un film se produisent ;
- *Quoi ?* fait référence aux **mots-clés** dont le film parle et aux **captions** qui décrivent la scène du film.
- *Quand ?* fait référence au **temps** guidant la succession des actions qui se produisent dans le film.

chaque réponse à ces questions – *personnages*, *lieux*, *mots-clés*, *visages*, et *captions* – forme les entités qui constituent l'objet de notre étude. Un réseau multi-couches met ces différentes entités en interaction pour former l'histoire du film. En exploitant d'autres sources supplémentaires telles que les sous-titres et le contenu multimédia du film. Ce réseau est composé de cinq couches de noeuds afin de représenter chaque type d'entité, personnages, mots-clés, lieux, visages et captions, avec les multiples interactions entre elles.

Nous modélisons deux classes principales de relations : les relations intra-couche entre des noeuds d'une même catégorie (*ex.* Deux visages apparaissant dans la même scène) ; et les relations inter-couches entre les noeuds de différentes catégories (*ex.* une caption décrit une scène dans laquelle un personnage est présent) ; Les multiples catégories de noeuds et de liens forment un réseau multi-couches, comme illustré dans la Fig.7.1.

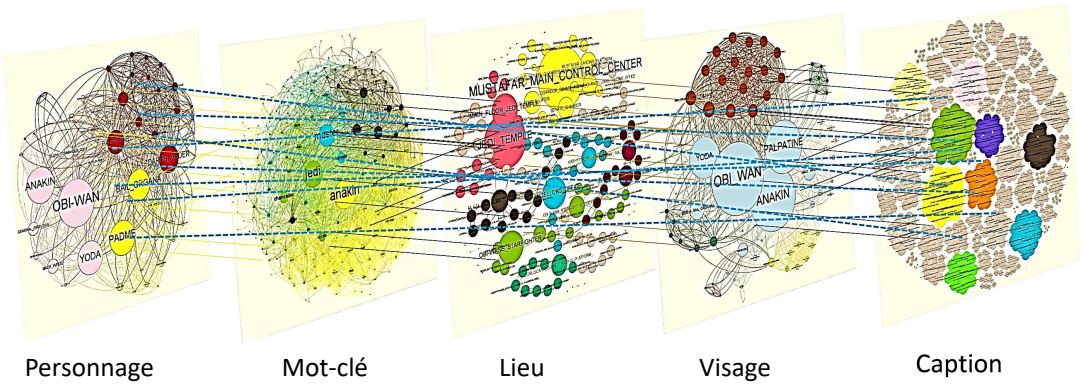

FIGURE 7.1 – Une présentation conceptuelle du modèle réseau multi-couches : cinq catégories de noeuds, *Personnage*, *Mot-clé*, *Lieu*, *Visage* et *Caption* sont en interaction entre et à travers chaque couche.

Nous définissons maintenant notre réseau multi-couches $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ tel que :

- $V_C \subseteq \mathbb{V}$ représente l'ensemble de personnages $c \in V_C$,
- $V_L \subseteq \mathbb{V}$ représente l'ensemble de lieux $l \in V_L$,
- $V_K \subseteq \mathbb{V}$ représente l'ensemble de mots-clés $k \in V_K$.
- $V_F \subseteq \mathbb{V}$ représente l'ensemble de visages $f \in V_F$.
- $V_{Ca} \subseteq \mathbb{V}$ représente l'ensemble de captions $ca \in V_{Ca}$.

Les différentes familles d'interactions peuvent être définies comme suit :

Intra-couche :

- $e \in E_{CC} \subseteq \mathbb{E}$ entre deux personnages tel que $e = (c_i, c_j) \in V_C^2$, quand un personnage $c_i \in V_C$ est en conversation avec un autre $c_j \in V_C$.
- $e \in E_{LL} \subseteq \mathbb{E}$ entre deux lieux tel que $e = (l_i, l_j) \in V_L^2$, quand il y a une transition temporelle d'un lieu $l_i \in V_L$ à un autre $l_j \in V_L$.
- $e \in E_{KK} \subseteq \mathbb{E}$ entre deux mots-clés tel que $e = (k_i, k_j) \in V_K^2$, quand $k_i \in V_K$ et $k_j \in V_K$ appartiennent à la même phrase.
- $e \in E_{FF} \subseteq \mathbb{E}$ entre deux visages $e = (f_i, f_j) \in V_F^2$, quand $f_i \in V_F$ et $f_j \in V_F$ apparaissent dans la même scène.
- $e \in E_{CaCa} \subseteq \mathbb{E}$ entre deux captions tel que $e = (ca_i, ca_j) \in V_{Ca}^2$, quand $ca_i \in V_{Ca}$ et $ca_j \in V_{Ca}$ décrivent la même scène.

Inter-couche :

- $e \in E_{CK} \subseteq \mathbb{E}$ entre un personnage et un mot-clé tel que $e = (c_i, k_j) \in V_C \times V_K$, quand le mot-clé $k_j \in V_K$ est prononcé par le personnage $c_i \in V_C$.
- $e \in E_{CL} \subseteq \mathbb{E}$ entre un personnage et un lieu tel que $e = (c_i, l_j) \in V_C \times V_L$, quand le personnage $c_i \in V_C$ est présent dans un lieu $l_j \in V_L$.
- $e \in E_{CF} \subseteq \mathbb{E}$ entre un personnage et un visage $e = (c_i, f_j) \in V_C \times V_F$, quand le personnage $c_i \in V_C$ apparait dans la même scène que le visage $f_j \in V_F$.

- $e \in E_{CCa} \subseteq \mathbb{E}$ entre un personnage et une caption tel que $e = (c_i, ca_j) \in V_C \times V_{Ca}$, quand un personnage $= c_i \in V_C$ apparait dans la même scène qu'une caption $ca_j \in V_{Ca}$ décrit.

- $e \in E_{KL} \subseteq \mathbb{E}$ entre un mot-clé et un lieu tel que $e = (k_i, l_j) \in V_K \times V_L$, quand le mot-clé $k_i \in V_K$ est mentionné dans une conversation qui se trouve dans un lieu $l_j \in V_L$.

- $e \in E_{KF} \subseteq \mathbb{E}$ entre un mot-clé et un visage tel que $e = (k_i, f_j) \in V_K \times V_F$, quand un mot-clé $k_i \in V_K$ est mentionné dans une scène ou le visage$f_j \in V_F$ apparait.

- $e \in E_{KCa} \subseteq \mathbb{E}$ entre un mot-clé et une caption tel que $e = (k_i, ca_j) \in V_K \times V_{Ca}$, quand un mot-clé $k_i \in V_K$ est mentionné dans une scène qu'une caption $ca_j \in V_{Ca}$ décrit.

- $e \in E_{LF} \subseteq \mathbb{E}$ entre un lieu et un visage tel que $e = (l_i, f_j) \in V_L \times V_F$, quand le visage $f_i \in V_F$ apparait dans la même scène qui se trouve dans un lieu $l_j \in V_L$.

- $e \in E_{LCa} \subseteq \mathbb{E}$ entre un lieu et une caption tel que $e = (l_i, f_j) \in V_L \times V_{Ca}$, quand une caption $ca_i \in V_{Ca}$ décrit une scène qui se trouve dans un lieu $l_j \in V_L$.

- $e \in E_{FCa} \subseteq \mathbb{E}$ entre un visage et une caption tel que $e = (f_i, ca_j) \in V_F \times V_{Ca}$, quand un visage $f_i \in V_F$ apparait dans la même scène qu'une caption $ca_j \in V_{Ca}$ décrit.

Dans cette contribution le poids et la direction des liens ne sont pas considéré. De plus, le temps n'est pas pris en compte. Cependant, le temps supporte tout : l'existence d'un noeud ou d'un lien est définie selon son apparition dans le temps qui guide le déroulement des scènes du film. Nous pouvons maintenant nous référer aux sous-graphes en ne considérant qu'une seule catégorie de liens et son sous-graphe induit.

- $G_{CC} = (V_C, E_{CC}) \subseteq \mathbb{G}$ représente le sous-graphe des interactions entre les personnages ;

- $G_{KK} = (V_K, E_{KK}) \subseteq \mathbb{G}$ représente le sous-graphe des co-occurrence des mots-clés ;

- $G_{LL} = (V_L, E_{LL}) \subseteq \mathbb{G}$ représente le sous-graphe des transitions entre les lieux ;

- $G_{FF} = (V_F, E_{FF}) \subseteq \mathbb{G}$ représente le sous-graphe des interactions entre les visages ;
- $G_{CaCa} = (V_{CaCa}, E_{CaCa}) \subseteq \mathbb{G}$ représente le sous-graphe des co-occurrence des captions ;
- $G_{CK} = (V_C \cup V_K, E_{CK}) \subseteq \mathbb{G}$ représente le sous-graphe des personnages parlant des mots-clés ;
- $G_{CL} = (V_C \cup V_L, E_{CL}) \subseteq \mathbb{G}$ représente le sous-graphe des personnages présents dans des lieux ;
- $G_{CF} = (V_C \cup V_F, E_{CF}) \subseteq \mathbb{G}$ représente le sous-graphe des personnages qui apparaissent avec des visages ;
- $G_{CCa} = (V_C \cup V_{Ca}, E_{CCa}) \subseteq \mathbb{G}$ représente le sous-graphe des personnages qui sont décrits par des captions ;
- $G_{KL} = (V_K \cup V_L, E_{KL}) \subseteq \mathbb{G}$ représente le sous-graphe des mots-clés mentionnés dans des lieux.
- $G_{KF} = (V_K \cup V_F, E_{KK}) \subseteq \mathbb{G}$ représente le sous-graphe des mots-clés prononcé par des visages.
- $G_{KCa} = (V_K \cup V_{Ca}, E_{KCa}) \subseteq \mathbb{G}$ représente le sous-graphe des mots-clés prononcé dans le même scène que les captions décrivent.
- $G_{LF} = (V_L \cup V_F, E_{LF}) \subseteq \mathbb{G}$ représente le sous-graphe des visages apparaissant dans un lieu.
- $G_{LCa} = (V_L \cup V_{Ca}, E_{LCa}) \subseteq \mathbb{G}$ représente le sous-graphe des captions décrivant des lieux.
- $G_{FCa} = (V_F \cup V_{Ca}, E_{FCa}) \subseteq \mathbb{G}$ représente le sous-graphe des captions décrivant des visages.

Après que nous avons défini le modèle du réseau multi-couches, nous devons extraire ses éléments à partir du script, des sous-titres et du contenu multimédia du film. Cela

permet d'analyser les diverses propriétés topologiques du réseau afin de mieux comprendre l'histoire du film.

7.3 Extraction des entités du réseau multi-couches

7.3.1 Description des données

Nous présentons maintenant la méthodologie utilisée pour construire le réseau multi-couches du film. Trois sources de données sont utilisées pour cette tâche : Le script, Les sous-titres et la vidéo.

7.3.1.1 Script

Comme il est présenté dans le chapitre précédent le script est un document textuel très bien structuré (Jhala, 2008). Il est composé de nombreuses scènes, chaque scène contient un lieu, une description de la scène, les personnages et leurs dialogues. Le contenu réel d'un script suit souvent un format semi-régulier (Jhala, 2008) tel que décrit dans la Fig.6.2.

7.3.1.2 Sous-titres

Les sous-titres sont disponibles dans un format SubRip Text (SRT) et se composent de quatre informations de base (Figure 7.2) : (1) un numéro identifiant de l'ordre des sous-titres ; (2) l'heure de début et de fin (heures, minutes, secondes, millisecondes) à laquelle le sous-titre doit apparaître dans le film ; (3) le texte du sous-titre lui-même sur une ou plusieurs lignes et (4) généralement une ligne vide pour indiquer la fin du bloc de sous-titres. Toutefois, les sous-titres n'incluent pas des informations sur les personnages, les scènes, les prises de vue et les actions, tandis que les dialogues dans un script n'incluent pas l'information du temps.

7.3.1.3 Vidéo du film

La vidéo du film est un ensemble des images fixes appelé frames, lorsqu'elles sont affichées sur un écran, créent l'illusion des images en mouvement. Cette illusion d'optique amène le public à percevoir un mouvement continu entre des objets distincts vus en succession rapide. Il est créé en photographiant des scènes réelles avec une caméra

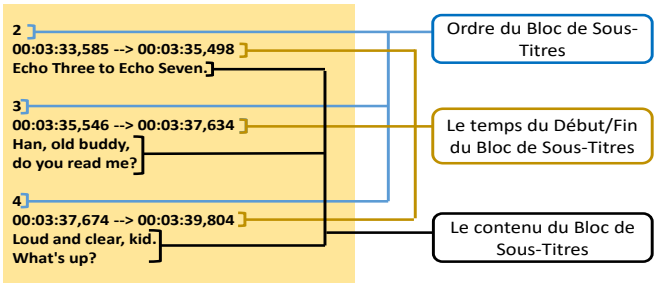

FIGURE 7.2 – Extrait de sous-titres du film *The Empire Strikes Back*.

cinématographique. Ces scènes fournissent des informations visuelles sur les personnages, les lieux, les événements,... .

Nous présentons maintenant la méthodologie utilisée pour extraire les différentes entités avec leurs interactions à partir de chaque source de données (script, sous-titres, vidéo). La figure 7.3 illustre le pipeline du processus de la méthodologie. Enfin, nous expliquons comment construire le réseau en fonction de ces informations.

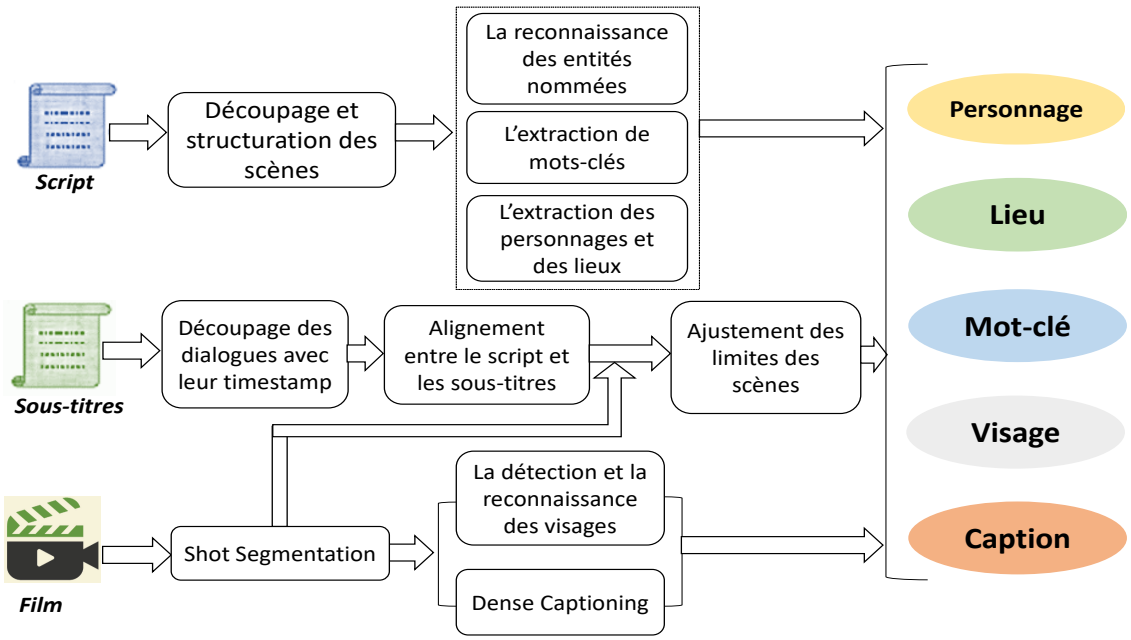

FIGURE 7.3 – Le schéma global du processus de construction du modèle.

Pour enlever toute ambiguïté, nous ajoutons quelques définitions à notre vocabulaire défini dans le chapitre précédent :

- Horodatage : L'information du temps peut être extraite par un alignement entre le

script et les sous-titres.

- Sous-titres : un ensemble de blocs du texte qui contiennent l'information du temps.
- Bloc de sous-titres : un bloc qui contient des énoncés qui ont un temps de début et de fin.
- Visage : Un visage de personnage détecté dans un frame, le visage est associé à un cadre sélectionné dans l'image.
- Caption : un ensemble de descriptions textuelles d'un frame, la caption à un cadre sélectionné dans l'image.

7.3.2 Préraitement du script :Découpage et structuration des scènes

Dans cette étape, on va suivre la même procédure qui est définie dans le chapitre précédent. Nous découpons d'abord le script en scènes. En effet, elles constituent les principales composantes d'un film et par conséquent, la principale unité d'analyse. Chaque scène contient des informations sur les personnages qui parlent, le lieu de la scène et les actions. Premièrement, nous avons une ligne de description technique écrite en majuscules pour chaque scène. Il établit le contexte physique de l'action qui suit. Pour découper le script en scènes, nous utilisons les marqueurs définis dans le chapitre précédent. Les personnes importantes et les objets clés sont généralement mis en lettres majuscules qu'on peut les collecter lors de l'analyse du texte. Le reste de la scène est constitué de descriptions de scène et des dialogues. Les noms des personnages et leurs actions sont toujours représentés avant les lignes du dialogue. Nous pouvons collecter les lieux et les énoncés des personnages, en structurant chaque scène en un ensemble de descriptions et de dialogues. Ensuite, nous identifions les conversations et les personnages présents dans cette scène. Des descriptions spécifiques peuvent ensuite être associées à des lieux et les dialogues sont associés à des personnages. L'étape suivante consiste à traiter le contenu textuel. Pour ce but, les outils de traitement automatique du langage naturel sont utilisés pour extraire les entités nommées et les mots-clés. NER (Named Entity Recognition). Nous utilisons la bibliothèque spaCy (Al Omran et Treude, 2017) comme dans le chapitre précédent. Pour

extraire les mots-clés on utilise les dialogues. Nous nous appuyons principalement sur la méthode LDA, qui apporte le meilleur compromis possible, en supprimant les mots clés moins sémantiques (tels que *can*, *have*, *etc.*).

7.3.3 Traitement de la vidéo

Comme les informations de la vidéo permettent également de répondre à quelques-unes des questions de *Ws*, nous introduisons dans cette contribution, deux techniques basées sur la vision par ordinateur. Pour commencer, on extrait pour chaque film une image clé (frame) par seconde, générant ainsi une moyenne de 8056 frames par film. Chacune de ces frames est ensuite associée à sa scène correspondante.

7.3.3.1 La détection et la reconnaissance des visages

Les visages sont ensuite détectés dans chaque frame. Pour extraire ces visages, nous déployons un outil de détection des visages basé sur l'architecture R-CNN qui est la plus rapide (Jiang et Learned-Miller, 2017) dans la littérature et entrainée avec WIDER. Cet algorithme propose des cadres de sélection pour chaque visage détecté (en moyenne de 5000 visages détectés par film). Nous supprimons ensuite manuellement les détections de faux positifs (un peu de 6.5% en moyenne). Nous devons maintenant identifier à qui appartiennent les visages, pour chacune de ces visages, nous utilisons la technique d'incorporation la plus récente de l'architecture ResNet50 (He *et al.*, 2016) formée sur le jeu de données VGGFace2 (Cao *et al.*, 2018). Cela nous permet d'obtenir un vecteur dimensionnel de 2048 pour chaque visage détecté. Étant que le nombre de visages détectés est limité pour chaque film, nous avons uniquement utilisé des approches automatisées pour faciliter l'annotation manuelle.

Afin de démarrer rapidement la création des classes, nous proposons tout d'abord une classification par DBScan (Ester *et al.*, 1996) pour lequel nous avons peaufiné les paramètres de notre premier jeu de données annoté manuellement, atteignant une précision approximative de 17 %. Sur la base de ces classes détectées et de la distribution du film, nous créons ensuite des modèles de visage sous forme de collections d'images afin d'aider progressivement à extraire de nouvelles images des mêmes personnages. Les résul-

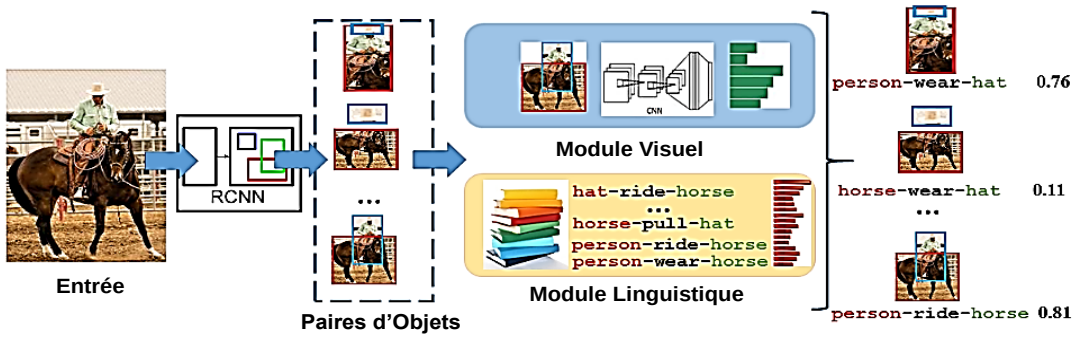

FIGURE 7.4 – Exemple d'extraction des captions à partir d'une image (frame).

tats contenant encore de nombreuses erreurs, nous les avons finalement tous sélectionnés manuellement pour obtenir une reconnaissance propre de chaque personnage.

7.3.3.2 *La détection des captions*

Nous voulons aussi explorer quels sont les objets et leurs relations qui pourraient être déduits des scènes elles-mêmes. La tâche de détection des captions (Johnson *et al.*, 2016) tente d'utiliser des outils de vision par ordinateur et d'apprentissage automatique pour décrire textuellement le contenu d'une image (dans notre étude frame). Nous avons utilisé une approche avec des articulations internes (Yang *et al.*, 2017) entrainée sur le génome visuel (Krishna *et al.*, 2017), qui est accessible au public [1]. Ceci permet d'associer des cadres à une image et attribuer des phrases significatives pour chaque cadre dans l'image (Figure 7.4) avec un indice de confidence $w \in [0, 1]$.

Du fait que notre extraction du frame produit des frames très similaires en fonction de la longueur du film, des captions similaires peuvent être répétées dans différentes frames et peuvent leur attribuer des différents indices de confiance. Pour extraire les captions pertinentes, nous proposons de les classer et de les filtrer.

Nous étendons la définition de *TF-IDF* (Salton *et al.*, 1975) $tfidf = tf * idf$ en intégrant l'indice de confidence de la caption. La notion de document correspond ici à une scène et à la place d'un terme, nous avons une caption. Nous définissons $tf(ca_i, S)$ la fréquence pondérée de la caption ca_i dans une scène s comme suit :

1. https ://github.com/InnerPeace-Wu/densecap-tensorflow

$$tf(ca_i, s) = \frac{\sum_{fr \in s} w_{ca_i,fr}}{\sum_{fr \in s} \sum_{ca \in f} w_{ca,fr}}$$

tel que ca dénote une caption qui a un indice de confidence $w_{ca,fr}$ dans une frame fr de la scène s. Après on définit $idf(ca_i, S)$ la fréquence de la scène inverse tel que :

$$idf(ca_i, S) = log\left(\frac{|S|}{|\{s \in S : ca_i \in s\}|}\right)$$

avec $\{s \in S : ca_i \in s\}$ représente les scènes s qui contiennent la caption ca_i dans le corpus composé par toutes les scènes du film S.

Nous appliquons un seuillage pour les top captions (top 40). Les captions sont des phrases simples, telles que *"a white truck parked on the street"*, et leur processus de génération les rend très ressemblantes (en raison des limitations du vocabulaire d'apprentissage). Pour extraire davantage leur contenu sémantique, nous avons calculé leurs n-grams (Cavnar *et al.*, 1994) ($n = 5$, en gardant un maximum de deux mots vides par n-gram). On applique ensuite un seuillage pour les tops n-grames. Le morceau des phrases obtenu peut ensuite être utilisé comme couche de captions supplémentaire obtenue à partir de la description visuelle de la scène.

7.3.4 Alignement temporel entre le script et les sous-titres

Nous devons maintenant faire l'alignement des informations sémantiques extraites du script à celles extraites de la vidéo. Cela peut naturellement être fait en alignant le script avec le minutage du film. Le film est joué dans le temps, mais le script ne contient aucune information temporelle. Heureusement, les dialogues sont marqués dans le script et doivent correspondre aux personnes qui parlent dans le film. Les sous-titres constituent la forme écrite de ces dialogues et ils sont codés dans le temps en synchronisation avec le film. L'idée est alors de les utiliser comme un intermédiaire pour assigner des codes temporels de dialogues correspondants au script. Par conséquent, nous devrions avoir une approximation exacte du moment où les scènes se produisent via les limites de début/fin des dialogues.

Malheureusement, la correspondance exacte des scripts et des dialogues varie considé-

rablement entre les versions du script et du film. Parfois, une scène peut apparaître dans le script mais pas dans le film, et inversement. D'autres fois, l'ordre et les phrases peuvent être très différents entre les deux.

Pour palier à ces problèmes, nous procédons en plusieurs étapes, comme présenté par Kurzhals *et al.* (2016). les scènes sont décomposées en blocs, pour lesquels chacune est un dialogue de personnage. Nous normalisons ensuite le texte des deux côtés par le biais du stemming. L'idée est alors d'affecter chaque bloc d'énoncé à son équivalent dans les sous-titres. Une première étape est de vérifier l'égalité absolue des sous-titres et du dialogue de script. Une deuxième étape consiste de vérifier l'inclusion du texte entre le script et les sous-titres. Cela ne fonctionne pas pour tous les énoncés mais la partie alignée donne des contraintes de la fenêtre de recherche pour notre prochaine étape. Pour les blocs restants, nous calculons leur vecteurs *TF-IDF* pondéré (Salton *et al.*, 1975) et on aligne avec la similarité minimale du cosinus.

Les mots-clés et les personnages peuvent alors être identifiés avec précision. Mais comme une scène contient une série d'énoncés, nous avons comme résultat une approximation des limites de temps de chaque scène, et aussi pour le lieu. Pour mieux aligner les scènes avec la vidéo, nous avons également appliqué une détection de *shots* approximative à l'aide de l'outil PySceneDetect (Castellano, 2012) et affiné les limites de la scène avec les limites du début et de la fin du *shot*, comme il est montré dans la Figure 7.5.

Cependant, nombreuses scènes ne contiennent aucun dialogue (une scène de bataille qui ne contient qu'une description de ce qui s'y passe) et ne peuvent donc pas être associées à un bloc de sous-titres (ces scènes sont souvent utilisées pour mieux rythmer la narration et peuvent généralement représenterune action de l'extérieur, par exemple un véhicule en mouvement). Dans d'autres cas, les scènes ne peuvent pas être associées à des sous-titres lorsque les blocs des dialogues sont trop petits ou trop modifiés, et que de nombreuses scènes ont en fait été effacées du script au montage du film final. Le tableau 7.1 résume ces statistiques.

L'emplacement de certaines de ces scènes peut encore être déduit de l'appariement des autres scènes. En effet, une scène qui n'a pas été alignée peut être insérée entre ses deux scènes voisines si elles ont été alignées précédemment. Lorsque plusieurs scènes

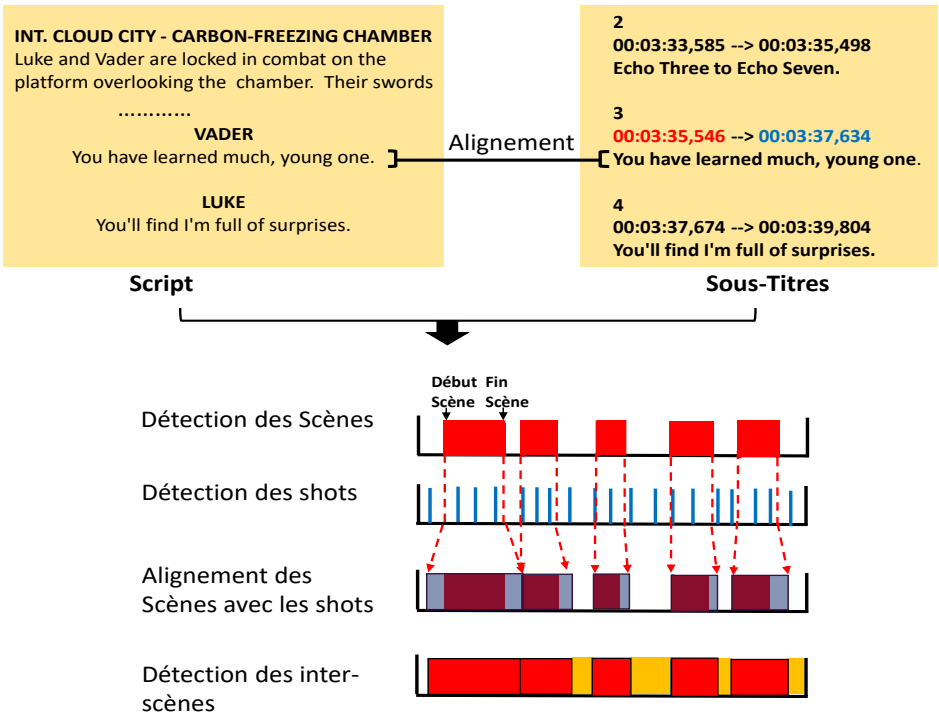

FIGURE 7.5 – Alignement entre le script et le sous-titres : Premièrement, le script est aligné avec le sous-titres. Puis, on raffine les limites de la scène avec le début et la fin du shot.

consécutives ne peuvent pas être associées, nous créons une méta-scène pour les regrouper. Par exemple, si nous avons un intervalle de scènes consécutives entre la scène 1 (00 :02 :00-00 :02 :20) et la scène 5 (00 : 02 : 46-00 :03 :52), nous créons la Meta Scene 2-4 (00 : 02 : 20-00 : 02 : 46) qui commence à la fin de la scène 1 et se termine au début de la scène 5.

7.3.5 Construction du réseau multi-couches

Suite aux étapes précédentes, pour chaque scène, nous avons identifié les lieux et leur description, les personnages, les mots-clés extraits à partir des énoncés, l'apparition des visages et la description des captions. Nous utilisons maintenant ces entités pour construire le graphe multi-couches. Reprenons nos questions d'investigation dans le contexte d'une scène : Comme nous avons présenté dans la chapitre précédent le *Qui ?*, *Où ?*, *Quoi ?* sont représentés respectivement par les personnages, les lieux et les mots-clés. De plus la question *Qui est apparu dans une scène ?* peut être identifiée par les visages et aussi pour

Episode	# Scènes alignées par le script	# Scène trouvées entre 2 scènes (#empty)	# Méta-scènes (#empty)	# Scènes Total (#empty)
SW1	109	23 (14)	51 (36)	183 (50)
SW2	58	22 (17)	70 (46)	150 (63)
SW3	75	16 (14)	97 (66)	188 (80)
SW4	223	66 (52)	193 (172)	479 (224)
SW5	146	52 (51)	77 (70)	275 (121)
SW6	89	27 (19)	22 (23)	138 (42)

TABLEAU 7.1 – Nombre de scènes, alignées, retrouvées, et manquées à parti du script, pour chaque épisode de Star Wars saga qui est un cas d'utilisation dans la Section 7.4.

Que s'est-il passé dans une scène ? peut être abordé par les captions. Plus que les trois familles d'entités (V_C, V_L, V_K) qu'on a défini dans le chapitre précédent pour former les catégories des noeuds, nous ajoutons deux familles V_F et V_{Ca} pour former respectivement les deux catégories visages et captions. Nous souhaitons maintenant construire les relations que nous avons décrites dans la sous-section 6.2. On va garder les même relations qu'on a défini dans le chapitre précédent, de plus on ajoute les relations suivantes : si deux visages f_i et f_j apparaissent dans la même scène, un lien est créé entre eux. Si deux captions ca_i et ca_j décrit une même scène, on crée un lien $e_{ca_i,ca_j} \in E_{CaCa}$.

En utilisant les informations extraites du script, des sous-titres et du contenu multimédia du film, nous pouvons ajouter des liens supplémentaires à ceux qui sont présentés dans le chapitre précédent. Si un personnage c_i est présent dans la même scène que le visage f_j un lien $e_{c_i,f_j} \in E_{CF}$ est créé entre eux. Un lien $e_{c_i,ca_j} \in E_{CCa}$ relie un personnage c_i avec une caption ca_j si la caption décrit une scène où le personnage est présent. On crée un lien $e_{k_i,f_j} \in E_{KF}$ entre un mot-clé k_i et un visage f_j si le mot-clé est mentionné dans une scène où le visage apparait. Quand un mot-clé k_i est mentionné dans une scène que la caption ca_i décrit, on crée un lien $e_{k_i,ca_j} \in E_{KCa}$. Un lien $e_{l_i,f_j} \in E_{LF}$ est créé entre le lieu l_i et le visage f_j quand le lieu est dans la scène où le visage apparait. On associe un lien $e_{l_i,ca_j} \in E_{LCa}$ entre un lieu l_i et une caption ca_j, si le lieu est dans la scène que la caption décrit. Finalement, quand un visage f_i apparait dans une scène que la caption ca_j décrit, un lien $e_{f_i,ca_j} \in E_{FCa}$ est créé. Un graphe résultant combinant toutes les couches est visualisé dans la Fig.7.1.

7.4 Analyse du réseau

Nous souhaitons maintenant effectuer une analyse des réseaux multi-couches de la série des 6-films Star Wars (SW). Tout d'abord, nous commençons par une brève introduction de la saga SW : la saga commence par l'épisode IV-Un nouvel espoir, suivi de deux suites, l'épisode V-L'empire contre-attaque (1980) et l'épisode VI-Le retour du Jedi (1983), cet ensemble est appelé la trilogie originale. Ensuite il y a la trilogie prequel qui vient après, composée de l'épisode I-La menace du fantôme (1999), de l'épisode II-L'attaque des clones (2002) et de l'épisode III-La revanche des Sith (2005). La saga SW raconte l'histoire d'un jeune garçon sauvé (Anakin) de l'esclavage et entraîné par les Jedi, soigné par les Sith. Il avait une liaison amoureuse avec une princesse (Amidala), puis il l'a mis enceinte. La mort de sa mère (Shmi) le pousse à se venger. Il a été forcé de joindre le côté obscur, après une bataille avec un Jedi (Obi-Wan) dans laquelle il risquait de mourir, il a été sauvé par les Sith. Finalement il devient un seigneur. Ses enfants jumeaux sont pris et cachés de lui, ils grandissent indépendamment, l'un devient une princesse (Leia) et l'autre un ouvrier de ferme (Luke), Luke cherche un vieux Jedi qui a des informations sur son père. Le Jedi (Obi-Wan) connait l'histoire du petit garçon, alors, il commence à l'entraîner. Luke tombe sur un message d'une princesse (Leia) en détresse et entreprend de la secourir. Ils embauchent un escroc (Han-Solo) avec son co-équipier (Chewbacca), pour un prix, qui va se charger de la retrouver et de la sauver, il s'avère qu'elle est la soeur jumelle perdue de Luke. Anakin, à présent est un Seigneur de Sith, ressent son danger, donc il l'a capturé. Il a su après que c'est son fils, il tente de retourner Luke vers le côté obscur, mais malheureusement, il est rejeté et le fils tue le père.

Avec autant de personnes à suivre dans cette saga, il peut être difficile de comprendre pleinement la dynamique qui les réuni. Pour démystifier cette saga, nous nous tournons vers la science des réseaux. Notre première tâche consiste à transformer chaque épisode de la saga en un réseau multi-couches. Une fois les graphes sont extraits pour chaque film, nous allons d'abord analyser leurs propriétés topologiques globales. Ensuite, pour étudier l'*influence* du noeud, nous allons la mesurer comme il est proposé par (Bioglio et Pensa, 2017), en faisant la moyenne du rang de trois mesures de centralité. Chaque mesure

de centralité donne des informations complémentaires. Nous choisissons les mesures de centralité *Degree*, *Betweenness* et *Eigenvector*. La centralité *Degree* d'un noeud mesure le nombre de liens avec les autres noeuds. La centralité *Betweenness* mesure la fréquence à laquelle un noeud se trouve sur des chemins courts entre d'autres paires de noeuds. Les noeuds avec une valeur de *Betweenness* élevée connectent différentes zones du réseau multi-couches. La centralité *Eigenvector* classe les noeuds d'un réseau en fonction du nombre de connexions qu'ils ont avec les noeuds de degré élevé. Nous avons également choisi ces mesures car chacune d'elles a été généralisée au cas du réseau multicouche (Domenico *et al.*, 2013).

7.4.1 Propriétés topologiques

Les propriétés topologiques de base des réseaux sont présentées dans la figure 7.6. Remarquons comment le nombre de noeuds des personnages et la complexité des réseaux changent entre les prequels (épisodes 1, 2 et 3) et les films originaux (épisodes 4, 5 et 6). La trilogie originale contient moins de noeuds importants et elle est densément interconnectée. La trilogie prequel a plus de noeuds dans l'ensemble, avec beaucoup plus de liens. Notez que pour tous les films, la couche des personnages G_{CC}, des visages G_{FF} et des lieux G_{LL} sont constitués d'un seul composant connecté. La couche de mots clés G_{KK} a quelques noeuds isolés. La couche des captions G_{CaCa} a un grand nombre de composants isolés. La couche des personnages est très dense par rapport aux autres couches. En effet, presque tous les personnages parlent ensemble dans le film. De plus, toutes les couches ont un diamètre et un chemin plus court moyen élevé, surtout pour la couche des lieux. Cela est dû à quelques transitions temporelles entre des lieux peu fréquents qui introduisent des longs chemins. Dans toute la SW saga, le coefficient de clustering est très élevé pour la couche des personnages, des visages et des captions. Pour le coefficient de clustering de la couche des captions, il est égal à 1. En effet, la couche des captions contient un nombre important de composants isolés par rapport aux autres couches. Nous pouvons expliquer cela par le fait que les captions sont regroupées par scène de film. La couche des mots-clés est également bien regroupée. Il n'y a pas de triangle dans les interactions inter-couches (comme dans G_{CL}, G_{CK}, G_{KL}, *etc.*) Car ce sont des graphes bipartites reliant

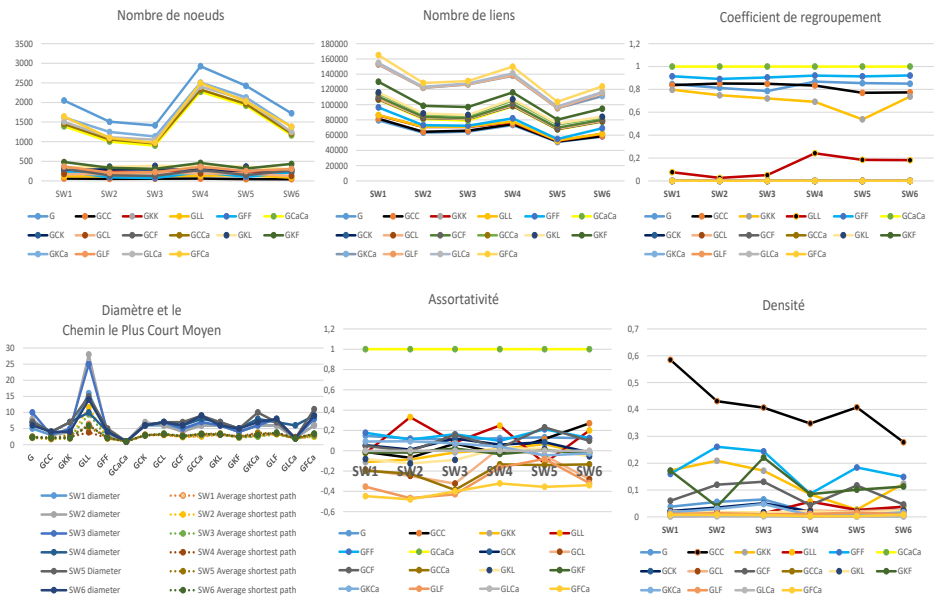

FIGURE 7.6 – Les propriétés topologiques de base par couche pour chaque film de SW saga.

deux ensembles d'objets. Nous pouvons observer que les réseaux multi-couches de chaque épisode de SW sont assortatifs. En effet, les principaux objets du film, en particulier les personnages et les visages, ont tendance à se connecter entre eux.

7.4.2 Analyse des réseaux pour chaque couche

Dans cette étape, nous ne présentons que le résultat d'un épisode de la SW saga, mais notre analyse s'applique sur toute la Saga. Pour voir tous les résultats, merci de voir les annexes (Matériel supplémentaire) pour plus de détails et des éclaircissements. Dans Matériel supplémentaire, nous appliquons le modèle proposé sur les 6 épisodes de SW et nous fournissons d'avantage des résultats qualitatifs pour tous les épisodes. Nous choisissons le premier épisode *"La menace du fantôme"* comme cas d'utilisation dans cette étude.

Voyons d'abord les réseaux dans les films individuels. Nous présentons les 10 premiers noeuds triés par leur score de centralité calculé dans les cinq couches considérées indépendamment pour les différents épisodes de SW. Nous avons toutefois pris quelques décisions discutables : Les personnages *Anakin* et *Darth Vader* sont représentés par deux

noeuds distincts, car cette distinction est importante pour l'histoire de la saga. D'autre part, nous différencions les noeuds qui ont les mêmes noms dans les réseaux multi-couches en ajoutant leurs catégories (pour le cas des visages et des personnages).

7.4.2.1 Classement des personnages

Il semble que *Qui-Gon* soit généralement le personnage le plus connecté du premier épisode de la trilogie prequel, en se basant sur les quatre métriques (Tab 7.2et A.1). *Qui-Gon* est un guerrier Jedi qui se lance dans une incroyable aventure pour sauver la planète Naboo. Après sa mort, son apprenti *Obi-Wan*, la jeune princesse *Amidala* et le jeune garçon *Anakin* prennent le top 3 de la liste du classement. Les trois personnages qui les accompagnaient dans leur aventure, le banni *Jar Jar* Binks et le puissant *capitaine Panaka*, qui voyagent tous vers les planètes de Tatooine et de Coruscant, pour une tentative de sauver leur monde contre *Palpatine*, le leader de la Fédération du commerce.

La question qui se pose, c'est à quoi ressemble les mêmes mesures pour la trilogie originale ?

Nous pouvons observer que tous les personnages qui sont dans le top du classement sont les principaux dans les trois épisodes. Avec *Luke Skywalker* dominant dans toutes les centralités de la trilogie. Cela est compréhensible étant donné que c'est le personnage principal de cette la trilogie originale. Si nous comparons le résultat avec le contenu du film, *Vader*, *Leia* et *Han Solo* accompagnent le personnage principal pendant tous les films. *Leia* est la princesse sauvée qui dirige les rebelles. *Obi-Wan* le capitaine du Millennium Falcon, collabore avec *C-3PO* pour sauver la belle princesse. *C3-PO* qui est à mon avis un noeud crucial qui semble jouer un rôle social important, car ils apparaissent fréquemment dans tous les films. Dark Vador est le vilain principal de la trilogie. Nous pouvons également observer l'apparition d'un nouveau personnage *Lando* dans le top du classement. *Lando* est un vieil ami de *Han Solo*.

7.4.2.2 Classement des mots-clés

Dans la trilogie prequel, il existe une distribution extrêmement inégale dans les centralités de mots-clés. Dans la liste du top du classement (Tableau 7.3 et A.2), nous trouvons

EPISODE	PERSONNAGES			
	D	B	Ei	I.S
SW1	Q.GON	Q.GON	Q.GON	Q.GON
	PADME	J.JAR	O.WAN	O.WAN
	O.WAN	ANAKIN	PADME	PADME
	J.JAR	O.WAN	J.JAR	J.JAR
	ANAKIN	PADME	ANAKIN	ANAKIN
	C.PANAKA	C.PANAKA	C.PANAKA	C.PANAKA
	NUTE	NUTE	NUTE	NUTE
	AMIDALA	AMIDALA	AMIDALA	AMIDALA
	SHMI	BIBBLE	R.OLIE	SHMI
	R.OLIE	SHMI	SHMI	R.OLIE

TABLEAU 7.2 – Les top 10 noeuds triés selon leurs scores de centralité dans la couche des personnages.

le nom du garçon *Anakin*, qui est le personnage principal de la trilogie prequel. Il est notamment le plus fréquent dans le film (Table 7.3). Nous pouvons également observer dans le top du classement que les mots-clés *jedi*, *princess*, *obi-wan* et *kill*. Nous pouvons expliquer leur apparition par le fait que dans la ville Coruscant, les Jedi *Obi-Wan* et *Anakin Skywalker* sauvent le *Chancelier suprême Palpatine* du vaisseau séparatiste de Général Grievous et *Anakin* tue *Count Dooku* avec son sabre laser après un combat. Quand ils atterrissent sur Coruscant, la *princess* Padmée Amidala vient annoncer à *Anakin* qu'elle est enceinte.

Dans la trilogie originale, nous observons la présence du nom de *Luke*, le personnage principal de cette trilogie. *Artoo* est son nouveau allié qui vont rencontrer *Han Solo*, *Chewbacca*, *Obi-wan* et *C-3PO*. Les tous tentent de sauver le chef des rebelles, *princesse Leia*, contre les griffes de l'empire. Dans cette trilogie, *Luke* tente d'obtenir une confirmation sur la prétention de *Dark Vador* d'être son père. Pour cela, on observe la présence du mot clé *father* dans la liste du classement.

7.4.2.3 Classement des lieux

Nous avons tout d'abord fait quelques abréviations pour les lieux. Le tableau des abréviations se trouve dans la section Matériel supplémentaire. Nous observons dans la trilogie précédente (Table 7.4 et A.3), les lieux les plus importants sont *FEDERATION*

EPISODE	Mots-clés			
	D	B	Ei	I.S
SW1	boy	anakin	boy	anakin
	anakin	jedi	anakin	jedi
	jedi	mesa	jedi	boy
	people	race	queen	queen
	queen	queen	people	mesa
	master	master	treaty	people
	mesa	droid	Q.gon	master
	promise	stay	master	ship
	treaty	ship	promise	treaty
	stay	boy	stay	Q.gon

TABLEAU 7.3 – Les top 10 noeuds triés selon leurs scores de centralité dans la couche des mots-clés.

BATTLESHIP BRIDGE (FBB), c'est un énorme vaisseau qui ressemble à des disques aplatis avec une sphère centrale contenant le pont du navire et assemblages de réacteurs. Ce navire contient différents espaces tels que Bridge, Hallway, Conference room où presque toutes les scènes de la série des films se déroulent. *JEDI TEMPLE (JT)* qui est un autre lieu important. Il désignait le quartier général de l'Ordre Jedi, en distinction des multiples académies locales de moindre importance disséminées à travers la galaxie. C'est ce bâtiment qui abritait la majeure partie des lieux d'enseignement et d'entraînement des recrues, les archives millénaires de la Bibliothèque, et surtout le siège du Conseil Jedi. *SPACE (SP)* qui est un autre lieu important où presque toutes les scènes de batailles se sont déroulées.

Dans la trilogie prequel, *SPACE* figure une autre fois dans le top du classement du tableau. *MILLENNIUM FALCON COCKPIT (MFC)* utilisé par Han Solo et Chewbacca pendant la guerre civile galactique - dans laquelle une autre partie majeure des événements a lieu. *REBEL STAR CRUISER (RSC)* est un autre lieu clé où se déroulaient les scènes les plus importantes sur son BRIDGE.

7.4.2.4 Classement des visages

Nous observons que *Anakin*, *Obi-Wan* et *Qui-Gon* jouent un rôle majeur dans la trilogie précédente (Tableau 7.5 et A.4) comme il est observé dans le classement des

EPISODE	LIEUX			
	D	B	Ei	I.S
SW1	FBB	NSQC	FBB	FBB
	AHMR	FBCR	FBHOB	TDNS
	TDNS	FBB	SCU	NSQC
	MEAVP	TDNS	FBH	AHMR
	MER	NSMA	FBMB	NSMA
	NSQC	CSLP	MEAVP	SCU
	NSMA	NPTR	MER	SNS
	SNS	AHMR	FBCR	NPTR
	NSC	NPTRT	NSQC	NSC
	WJS	MEAMH	AHMR	TPTR

Tableau 7.4 – Les top 10 noeuds triés selon leurs scores de centralité dans la couche des lieux.

EPISODE	VISAGES			
	D	B	Ei	I.S
SW1	A.DOPPELGANGER	ANAKIN	Q.GON	ANAKIN
	ANAKIN	O.WAN	A.DOPPELGANGER	A.DOPPELGANGER
	Q.GON	A.DOPPELGANGER	ANAKIN	Q.GON
	O.WAN	Q.GON	O.WAN	O.WAN
	J.JAR	J.JAR	J.JAR	J.JAR
	PADME	PADME	C.PANAKA	PADME
	C.PANAKA	T.R.SPEAKER	PADME	C.PANAKA
	PALPATINE	C.PANAKA	SHMI	PALPATINE
	SHMI	SEBULBA	PALPATINE	SHMI
	NUTE	PALPATINE	RUNE	NUTE

Tableau 7.5 – Les top 10 noeuds triés selon leurs scores de centralité dans la couche des visages.

personnages, qui confirme notre hypothèse. L'apparition de nouveaux noeuds clés est observée dans le top du classement, par exemple l'un des *Amidala Doppelgänger* qui est tenu de protéger la reine. Il semble que ce personnage ne joue pas un rôle important dans le film, mais sa présence dans presque toutes les scènes l'a mis dans le top du classement. Dans la trilogie originale de SW, nous observons la même chose que dans le classement des personnages, à nouveau *Luke Skywalker* dominant toutes les centralités avec *Leia* et *Han Solo* dans les trois épisodes. En outre, un nouveau noeud clé apparaît dans la liste du classement qui est le premier compagnon de *Han Solo* - *Chewbacca*. C'est un personnage passif (il ne parle pas), ce qui rend la tâche de sa détection plus difficile à partir du texte si on a pas le contenu visuel.

EPISODE	CAPTIONS			
	D	B	Ei	I.S
SW1	(metal ; pot ; a ; metal ; pot)	(metal ; pot ; a ; metal ; pot)	(metal ; pot ; a ; metal ; pot)	(metal ; pot ; a ; metal ; pot)
	(person ; wearing ; a ; green. shirt)	(person ; wearing ; a ; green ; shirt)	(person ; wearing ; a ; green ; shirt)	(person ; wearing ; a ; green ; shirt)
	(green ; shirt ; a ; man ; standing)	(green ; shirt ; a ; man ; standing)	(green ; shirt ; a ; man ; standing)	(green ; shirt ; a ; man ; standing)
	(background ; a ; black ; metal ; pot)	(background ; a ; black ; metal ; pot)	(background ; a ; black ; metal ; pot)	(background ; a ; black ; metal ; pot)
	(metal ; door ; a ; black ; bowl)	(metal ; door ; a ; black ; bowl)	(metal ; door ; a ; black ; bowl)	(metal ; door ; a ; black ; bowl)
	(black ; metal ; pot ; a ; metal)	(black ; metal ; pot ; a ; metal)	(black ; metal ; pot ; a ; metal)	(black ; metal ; pot ; a ; metal)
	(gray ; shirt ; a ; black ; metal)	(gray ; shirt ; a ; black ; metal)	(gray ; shirt ; a ; black ; metal)	(gray ; shirt ; a ; black ; metal)
	(jacket ; man ; wearing ; a ; red)	(jacket ; man ; wearing ; a ; red)	(jacket ; man ; wearing ; a ; red)	(jacket ; man ; wearing ; a ; red)
	(brown ; a ; silver ; metal ; handle)	(brown ; a ; silver ; metal ; handle)	(brown ; a ; silver ; metal ; handle)	(brown ; a ; silver ; metal ; handle)
	(room ; a ; red ; metal ; pole)	(room ; a ; red ; metal ; pole)	(room ; a ; red ; metal ; pole)	(room ; a ; red ; metal ; pole)

TABLEAU 7.6 – Les top 10 noeuds triés selon leurs scores de centralité dans la couche des captions.

7.4.2.5 Classement des captions

Dans la liste du classement de la couche des captions (Tableau 7.6 et A.5), on remarque l'apparition de captions telles que *black, dress, two, people, standing, metal, baseball, lamp,* etc. Lorsque nous comparons ces captions avec le contenu de la scène où elles apparaissent dans le film, nous constatons que *(metal, baseball, lamp)* fait référence au light-saber, *(black, dress, two, people, standing)* se réfèrent à Anakin et Palpatine. Dans cette scène, quand Obi-Wan est envoyé à Utapau, pour tuer le général Grievous. Palpatine se révèle à Anakin en tant que Seigneur Sith, Dark Sidious, qui contrôlait la République et le mouvement séparatiste. Anakin part pour le dénoncer au Conseil Jedi. Pour nous, c'est un événement important dans le film qui pousse Anakin à devenir l'apprenti Sith de Palpatine, ce dernier à lui a donné le nom de Dark Vador.

7.4.3 Analyse des réseaux multi-couches

Les tableaux 7.7 et A.6 indiquent les scores de centralité calculés dans les réseaux multi-couches des 6 films. Les top noeuds du classement sont généralement les visages connus du film. Toutefois, les principaux mots-clés et personnages figurent également dans le top de la liste du classement. On peut remarquer que ce classement peut résumer l'histoire de la trilogie prequel. Lorsque la Trade Federation organise un blocus autour de la planète Naboo, le Suprême **Chancellor** Valorum envoie les **Jedi Qui-Gon** et **Obi-Wan** pour négocier la fin du blocus. Cependant, le méchant vice-roi Nute Gunray a été chargé de tuer le **Jedi** et d'envahir Naboo. Cependant, les **Jedi** escape et **Qui-Gon** sauvent la vie du maladroit Gungan **Jar Jar** Binks. Les **Jedi** se rendent dans la capitale pour avertir la reine **Amidala Padme** de l'invasion. Lorsqu'ils s'échappent des droïdes de la Fédération,

ils se rendent sur la planète désertique de Tatooine où ils rencontrent un garçon esclave nommé **Anakin** Skywalker, qui fait manifestement partie de la Force. Dix ans après que la "menace du fantôme" ait menacé la planète Naboo, **Padme Amidala** est maintenant un **Senator** représentant son monde natal. Une faction de séparatistes politiques, dirigée par le Count Dooku, tente de l'assassiner. Le **Chancellor Palpatine** demande donc l'aide de Jango Fett, qui a promis que son armée de clones gérera la situation. A Coruscant, les **Jedi Obi-Wan** et **Anakin** Skywalker délivrent le Suprême **Chancellor Palpatine** du vaisseau spatial du général séparatiste Grievous et après **Anakin** tue le Count Dooku avec son light-saber après une bataille entre eux ; Cependant, **Grievous** s'échappe du **Jedi**. Quand ils atterrissent sur Coruscant, **Padme Amidala** vient annoncer à Anakin qu'elle est enceinte (un bébé **arrive**). Comme observé, dans la liste du classement des réseaux multicouches de la trilogie précédente, les mots gras précédents sont capturés par le réseau multicouche, ce qui prouve l'efficacité du modèle proposé.

Dans la trilogie originale, les àtops noeuds de la liste du classement sont presque les visages et les personnages principaux du film. En outre, nous pouvons observer que les mots-clés principaux et certains des lieux principaux sont présents aussi dans la liste du classement. En analysons le tableau du classement, il peut nous raconter une brève histoire de cette trilogie. Les forces impériales, sous les ordres du cruel Dark Vador, tiennent en otage la princesse **Leia** dans leurs efforts pour réprimer la rébellion contre l'empire galactique. **Luke** Skywalker (qui est né dans le dernier épisode de la trilogie prequel) et **Han Solo**, capitaine du Millennium Falcon, collaborent accompagné les deux **droid**s R2-D2, **C-3PO** et **Chewbacca** l'ami de **Han Solo** pour sauver la belle princesse, et aider l'alliance rebelle pour restaurer la liberté et la justice dans la galaxie. **Luke** Skywalker commence l'entraînement de Jedi avec le légendaire Jedi **Master Yoda**. Seule l'aide du **Master** permettra à **Luke** de survivre lorsque le côté obscur de la Force le convie à se lancer dans le duel ultime avec Dark Vador. L'empereur Palpatine, Dark Vador et l'Empire construisent une nouvelle étoile de la mort. Pendant ce temps, **Han Solo** a été emprisonné et **Luke** Skywalker a envoyé R2-D2 et **C-3PO** pour essayer de le libérer. Ensuite, **Han Solo** et **Princesse Leia** réaffirment leur amour et font équipe avec **Chewbacca**, **Lando** Calrissian, les Ewoks, **droid**s **C-3PO** et R2-D2 pour aider à la perturbation du côté

EPISODE	Réseau Multi-couches			
	D	B	Ei	I.S
SW1	Q.GON	A.DOPPELGANGER	Q.GON	Q.GON
	A.DOPPELGANGER	Q.GON	A.DOPPELGANGER	A.DOPPELGANGER
	ANAKIN	ANAKIN	O.WAN	ANAKIN
	O.WAN	O.WAN	ANAKIN	O.WAN
	J.JAR	J.JAR	J.JAR	J.JAR
	C.PANAKA	Q.GON	C.PANAKA	C.PANAKA
	Q.GON (CHARACTER)	C.PANAKA	jedi	Q.GON (CHARACTER)
	anakin	PADME	anakin	anakin
	master	anakin	master	master
	jedi	master	highness	jedi

TABLEAU 7.7 – Les top 10 noeuds triés selon leurs scores de centralité dans les réseaux multi-couches.

obscur et à la défaite de l'empereur diabolique. Nous pouvons observer que les mots gras précédents sont également capturés par les réseaux multi-couches dans la trilogie originale.

7.4.4 Détection des communautés

Nous sommes également intéressés par les structures communautaires dans les films. Nous voulons diviser les réseaux multi-couches à des communautés cohérentes, ce qui signifie qu'il existe de nombreux liens au sein des communautés et peu entre elles. Nous détectons les communautés du réseau en utilisant une métrique globale appelée modularité $-1 \leq Q \leq 1$. Notre objectif est de partitionner les noeuds en communautés de manière à maximiser Q. Trouver le bon partitionnement est une tâche difficile, nous utilisons donc une méthode d'approximation rapide appelée Louvain (Blondel *et al.*, 2008). Nous présentons dans la Figure 7.7 le nombre des communautés avec la modularité par couche pour chaque film du SW saga. Pour tous les films, nous observons que la couche des captions contient un nombre important de communautés. En effet, presque toutes les captions sont regroupées par scène. Pour la modularité, encore une fois, on remarque que la couche des captions dans de tous les films montre son avantage par rapport aux autres couches.

Pour analyser la structure communautaire d'un film, nous avons choisi dans cette contribution un cas d'utilisation du film : Épisode III - La revanche des Sith (2005). La même méthodologie peut être appliquée aux autres épisodes.

En observant Figure 7.8, on peut dire que la couche des personnages a un total de

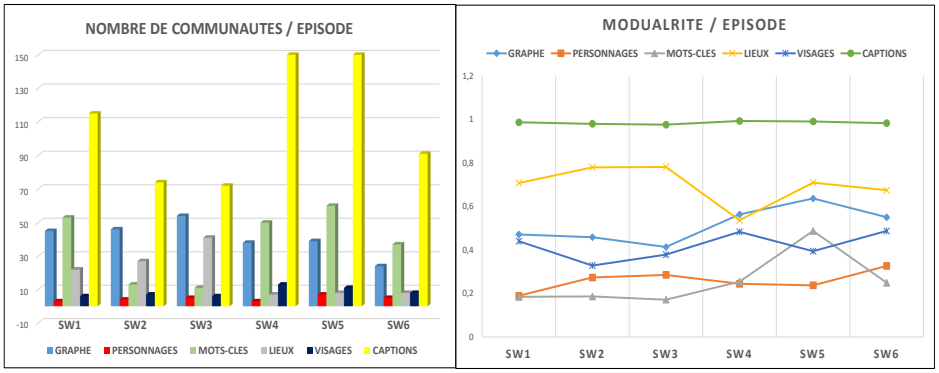

FIGURE 7.7 – La modularité et le nombre de communautés par couche pour SW saga.

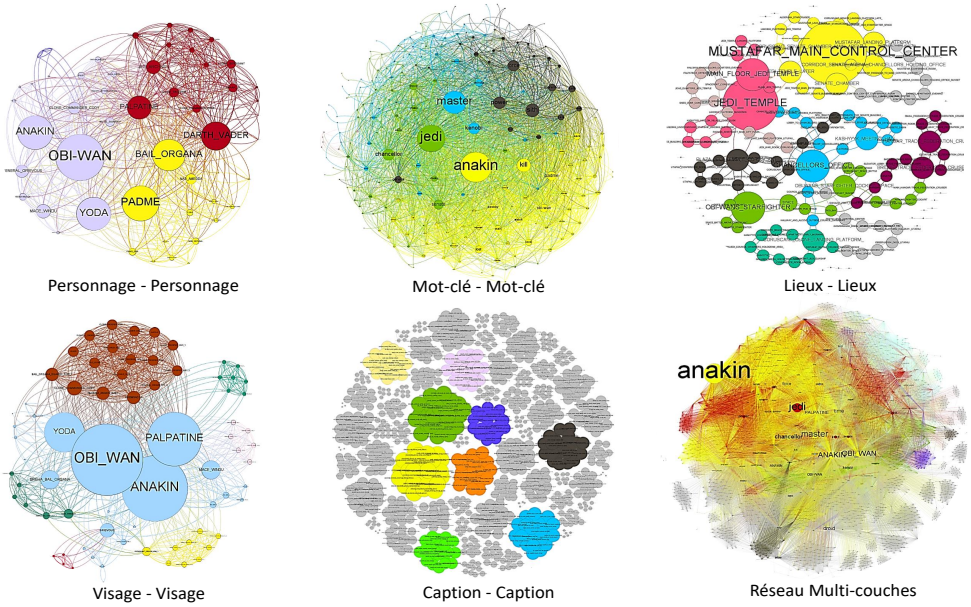

FIGURE 7.8 – Visualisation des communautés dans les différentes couches de l'Épisode III - La revanche des Sith (2005). La taille du noeud correspond à son degré.

4 communautés. Les trois plus grandes communautés sont la verte, violette et la jaune. la communauté verte est composée de l'équipe *Palpatine* avec *Darth Vader* et *Palpatine Followers*, nous pouvons expliquer la présence de *Palpatine* et de *Darth vader* dans la même communauté, car en tant que garde du corps du chancelier, *Anakin* développe une proche amitié avec *Palpatine*. Ce dernier prend *Anakin* comme son apprenti Sith et lui donne le nom de *Darth Vader*. Il ordonne ensuite à *Vader* de tuer tous les Jedi du Jedi Temple, puis d'aller vers le Mustafar system et éliminer les dirigeants séparatistes. Dans

la communauté violette, nous pouvons observer la présence des deux Jedi *Obi-Wan* et de *Anakin*, accompagnés par l'équipe du conseil tel que *Yoda*, *Mace Windu*, *Ki-Adi*, *etc.* Nous pouvons aussi remarquer la présence du *Général Grievous*. Nous expliquons cela par le fait que le chancelier *Palpatine* a été kidnappé par le *Général Grievous*, *Anakin* et *Obi-Wan* mener une mission pour le sauver. Après avoir libéré le chancelier, ils tentent de s'échapper, mais ils sont piégé par le *General Grievous*. *Anakin* et *Obi-Wan* tentent de se libérer, mais *Grievous* échappe et laisse les Jedi et le chancelier à l'intérieur du croiseur gravement endommagé. Dans la couche de mots-clés, nous pouvons observer un total de 10 communautés. La plus importante est la communauté jaune, elle est composée de mots-clés tels que *anakin, kill, padme, obi-wan, vader*. Comme nous pouvons le constater dans la figure 7.8 (mot-clé à mot-clé), *anakin* est le mot clé principal de cette communauté, suivi de mots-clés comme *kill, padme et obi-wan*. Si nous comparons cela avec l'histoire du film, les deux Jedi *obi-wan* et *anakin* ont la mission de protéger *padme* après son échappement de meurtre (*kill*). Nous pouvons également observer la présence du mot clé *vader* dans la même communauté, car *anakin* se retourne contre son mentor obi-wan et devient Dark Vader sous l'ordre de *Palpatine*.

La couche des lieux regroupe 40 communautés, la plus grande est la communauté jaune. Elle contient le lieu *Mustafar* et les différentes endroits qui le compose, tel que *Mustafar Main Control Center*, *Mustafar Landing Platform*. Nous notons également la présence des lieux importants tels que *Senate Chamber, Corridor Senate Arena Chancellors Holding Office*. Si nous comparons avec l'histoire du film quand Obi-Wan rencontre Padme, qui refuse de croire ses affirmations sur la chute d'Anakin au côté obscur. Obi-Wan se cache secrètement dans le navire, lorsqu'elle part vers *Mustafar*, par la suite Obi-Wan Kenobi et Dark Vader (Anakin) se battent à *Mustafar* et Yoda affronte l'empereur (Palpatine), en libérant sa force de l'ancien Jedi. En retour, Yoda jette le Seigneur Sith sur son *Senate office*. Les deux plus puissants praticiens de la Force sont engagés dans un duel féroce à l'intérieur de la *Senate Chamber*.

La couche des visages est composée de 7 communautés, la plus grande est la violette. Il contient des visages tels que *Obi-Wan, Anakin, Palpatine, Yoda, Macé Windu,* etc. Lorsque nous comparons avec les événements du film, presque tous ces personnages

apparaissent dans les mêmes scènes, soit en deux ou bien en trois visages par scène. Par exemple, *Obi-Wan* est presque présent avec *Anakin* dans toutes les scènes du film et également avec *Palpatine* dans certaines scènes. Nous argumentons ces résultats par le fait qu'à Coruscant, les Jedi *Obi-Wan* et *Anakin* délivrent le chancelier *Palpatine* du *General Grievous* (qui fait également partie de la communauté). De plus, nous remarquons la présence de Mace Windu et Yoda dans la même communauté, car dans certaines scènes du film, les deux personnages apparaissent avec les autre personnages, par exemple lorsque *Mace Windu* arrive au bureau du chancelier et vaincre *Palpatine* après un duel au sabre laser, au moment où *Anakin* arrive, *Windu* est sur le point de tuer le *Palpatine*. *Anakin* désarme rapidement *Windu*, ce dernier est face aux pouvoirs de*Palpatine*, qui le forcent à passer par la fenêtre pour enfin mourir.

Nous remarquons dans la couche des captions un grand nombre de communautés, un total de 70 communautés. C'est parce que les captions sont regroupées par scène. Chaque scène comporte un certain nombre de captions décrivant ce qui s'est passé dans cette scène. Nous observons que la plus grande communauté est la communauté jaune. Si nous explorons cette dernière, nous remarquons des descriptions telles que *black, dress, two, people, standing, metal, baseball, lamp,* etc. qui sont présentes dans le top du classement des noeuds de la couche des captions. Ce résultat confirme notre hypothèse sur les captions mentionnées. Les captions sont très corrélés avec les événements importants de l'histoire du film.

Dans le réseau multi-couches du film, nous avons un total de 56 communautés, nous remarquons la présence des noeuds majeurs des différentes couches danse la même communauté (la plus grande "jaune"), tels que le personnage *Anakin*, avec le mot-clé *anakin* et le visage *Anakin*, qui prouve notre analyse précédente que *Anakin* est le personnage principal de la trilogie prequel. Nous observons la même chose pour *Obi-Wan*. Nous remarquons également que certains noeuds clés tels que *Master, Chancellor, Palpatine, kill, Yoda et Mace Windu* sont présents dans la même communauté, ce qui prouve les discussions précédentes sur leur relation. Si nous comparons chaque communauté avec le contenu du film, nous pouvons remarquer une bonne correspondance avec les événements des scènes, ce qui prouve l'efficacité du modèle proposé pour raconter l'histoire du film à

partir d'un réseau multicouches.

7.5 Conclusion

Dans cette contribution, nous avons présenté un modèle multi-couches qui met en interaction les différents éléments d'un film *personnages*, *lieux*, *mots-clés*, *visages* et *captions*. Contrairement aux réseaux mono-couche, qui se concentrent uniquement sur les *personnages* ou les *scènes*, ce modèle est beaucoup plus riche et informatif. Il supporte l'analyse des réseau des personnages en proposant des nouveaux outils d'analyse topologique composé des éléments plus sémantiques, qui peuvent nous donner une image globale de l'histoire du film. Nous proposons également une méthode automatique pour extraire les éléments de réseau multi-couches en utilisant les sous-titres le script et le contenu multimédia du film. Afin d'enrichir notre modèle, des éléments multimédias supplémentaires sont inclus, tels que la reconnaissance faciale, le dense captioning et les informations de sous-titres.

Nous avons déployé ce modèle sur les six films populaires de Star Wars saga. Les résultats d'une analyse des réseaux construits ont confirmé l'efficacité du modèle. Jusqu'ici, nous avons considéré la succession de scènes comme une granularité temporelle. Nous pouvons cependant étendre cette notion et tenter de récupérer du temps tel qu'il est représenté dans le monde du cinéma, ça peut nous aider à étudier la localisation des personnages au long du film qui permettra une meilleure transition entre les lieux. Notez aussi qu'une analyse topologique plus approfondie permet d'obtenir beaucoup plus d'informations, par exemple un réseau de co-occurrences de personnages se trouvant au même endroit, un réseau de conversations orienté, *etc.* Nous aimerions aussi d'étudier cet outil sur une plus grande collection de données, telles que les séries télévisées, ou même une plus grande collection de films. Le modèle de réseau multicouche proposé peut également être utilisé pour caractériser le genre du film, et même pour établir une corrélation avec les carrières d'acteur à partir de bases de données publiques telles qu'IMDB. De plus, en utilisant cet outil, nous pouvons générer automatiquement la bande-annonce d'un film en recherchant les scènes importantes où tous les personnages du film sont présents. Nous travaillons

également sur l'ajout d'une autre couche dans ce réseau multi-couches - *Émotion*, qui est maintenant reconnue comme un aspect important pour caractériser les personnages et le genre du film (Drama, Comédie, *etc*).

Conclusion et perspectives

L'analyse visuelle est un domaine basé sur la combinaison de l'art de l'intuition humaine et la science de la déduction mathématique pour percevoir directement des représentations visuelles et d'en tirer des connaissances et la perspicacité. Elle peut avoir plusieurs représentations (image ou vidéo) qui la rend plus compréhensible et permettent d'en extraire de la connaissance. Différentes techniques ont été proposées pour analyser les images et les vidéos, mais dans cette thèse nous mettons en lumière que les travaux qui sont basés sur les réseaux complexes. C'est pour cette raison nous avons consacré le chapitre 2 aux réseaux complexes pour rappeler les terminologies dont nous avons eu besoin pour la suite de ce mémoire. Le travail de cette thèse a été divisé en deux parties, la première partie s'adresse au problème de la segmentation des images, tandis que la deuxième fut consacrée à l'analyse des histoires des films. En premier temps, dans la première partie, nous avons présenté dans le chapitre 3, un état de l'art sur les approches de segmentation des images. En effet dans la littérature, pour catégoriser les approches de segmentation d'images, plusieurs manières ont été proposées, certains travaux les a classées en quatre classes : - Approche par contours ; - Approche Pixels ; - Approche régions ; - Approche hybride. D'autres en deux classes principales : - Approche frontières ; - Approche régions. Certains aussi en deux classes : - Couleur ; - Texture, et finalement les approches utilisant la théorie des graphes. Pour évaluer les performances des méthodes de segmentation, nous avons présenté quelques bases de données des images dans ce chapitre avec les métriques les plus utilisées dans l'évaluation des résultats de la segmentation. Avec le développement de la théorie des réseaux complexes, la segmentation d'images en se basant sur les graphes a considérablement évoluée. L'identification des régions de

pixels peut être remplie par des méthodes de détection de communautés sur les noeuds d'un graphe. Le chapitre 4 commence par la présentation du schéma global du framework proposé pour la segmentation des images en se basant sur les algorithmes de détection des communautés. Ensuite, la description théorique de chaque étape de ce schéma est présentée. Le framework commence premièrement par une segmentation initiale de l'image afin de construire un graphe de régions adjacentes. Les noeuds représentent les régions initiales et les liens existent entre deux noeuds, si les deux régions qui leurs correspondent sont adjacentes. Puis le lien est pondéré selon la similarité entre les régions en exploitant les caractéristiques de l'image (texture et couleur). Ensuite un processus itératif est appliqué sur le graphe résultant pondéré en utilisant un algorithme de détection des communautés pour partitionner le graphe à un ensemble de communautés. Ces communautés sont utilisées par la suite pour regrouper les régions adjacentes de l'image. En effet, tous les noeuds appartenant à la même communauté sont considérés comme appartenant à la même région et fusionnés en une seule région dans l'image. Le processus s'arrête quand il n'y ait plus de différence entre les structures communautaires de deux itérations successives. Les expérimentations ont montré que le framework proposé donne le meilleur résultat de segmentation qualitative et réalise la meilleure performance quantitativement en comparant à toutes les méthodes de la littérature, en termes de PRI, VOI, Précision et Rappel. La deuxième partie de ce travail est consacré sur l'analyse des histoires des films. Nous avons commencé par le chapitre 5 pour présenter les différentes approches qui ont été utilisées pour ce but, mais surtout nous avons mis l'accent sur les approches fondées sur les graphes.

Pour analyser une histoire, on articule souvent les 5 questions : Qui ?, Où ?, Quoi ?, Quand ? et comment/pourquoi ?. L'analyse des histoires par les graphes tente de répondre à la question Comment/Pourquoi ? en articulant les 4 autres questions dans un graphe. Les travaux précédents ont principalement centrés sur le Qui ? dans un réseau monocouche. Dans le chapitre 6, nous avons introduit une approche plus holistique avec une modélisation réseau multi-couche des histoires des films. Comme il est toujours difficile d'obtenir toutes les informations nécessaires à partir de données vidéo elles-mêmes, nous pouvons se reposer sur la modalité textuelle. Généralement tous les films sont écrits sous

forme de script. Un script est généralement bien structuré et contient tous les composants nécessaires pour analyser automatiquement un film (scènes, dialogues, personnages, *etc.*). Pour extraire le réseau multi-couches (personnages, lieux et mots-clés), nous avons fait recours à l'analyse du texte. Nous avons déployé ce modèle sur deux films populaires. Nous avons ensuite présenté les résultats d'une analyse des réseaux extraits du film. Cette analyse a confirmé l'efficacité du modèle. Ce dernier a permis de capturer une structure plus riche d'un film. Le modèle proposé a complété l'analyse de réseau mono-couche basée seulement sur les caractères en apportant de nouveaux outils d'analyse topologique. Pour but d'enrichir le modèle proposé et avoir une vue plus globalement sur les histoires des films, nous avons proposé dans le chapitre 7 un modèle de réseau multi-couches qui capture la structure et les éléments d'un film de manière plus riche par rapport aux réseaux classiques en exploitant la modalité textuelle et visuelle pour ajouter deux couches supplémentaires (visages et captions) . Nous avons déployé ce modèle sur les six films populaires de Star Wars saga. Les résultats d'une analyse de réseaux construits ont confirmé l'efficacité du modèle. Notant aussi qu'une analyse topologique plus approfondie permet d'obtenir beaucoup plus d'informations, par exemple un réseau de co-occurrences de caractères se trouvant au même endroit, un réseau de conversations orienté, *etc.*. Le modèle de réseau multicouche proposé peut également être utilisé pour caractériser le genre du film, et même pour établir une corrélation avec les carrières d'acteur à partir de bases de données publiques telles qu'IMDB. De plus, en utilisant ce modèle, nous pouvons automatiquement générer les bandes-annonces des films en recherchant les scènes importantes qui contiennent les personnages clés du film. Nous travaillons également sur l'ajout d'une autre couche dans ce réseau multi-couches - *Émotion*, pour caractériser les personnages et le genre de film (drama, comédie, *etc*). Aussi, nous envisageons d'exploiter la notion de similarité entre les graphes pour construire un système de recommandation des films.

Annexes

Annexe

A

Matériels supplémentaires

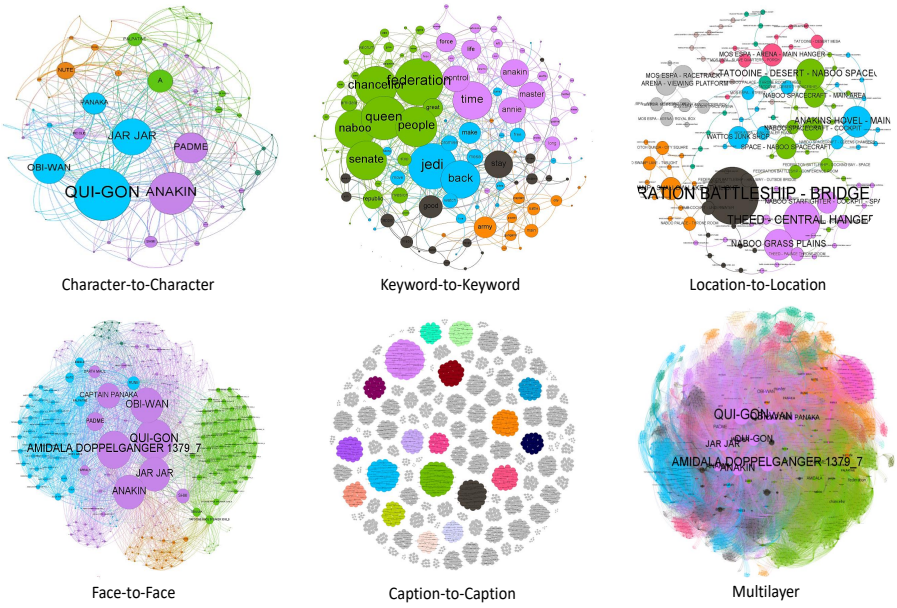

Figure A.1 – Visualisation des communautès dans les différentes couches de l'Episode I -La menace du fantôme (1999)Lucas (1999). La taille du noeud correspond à son degré.

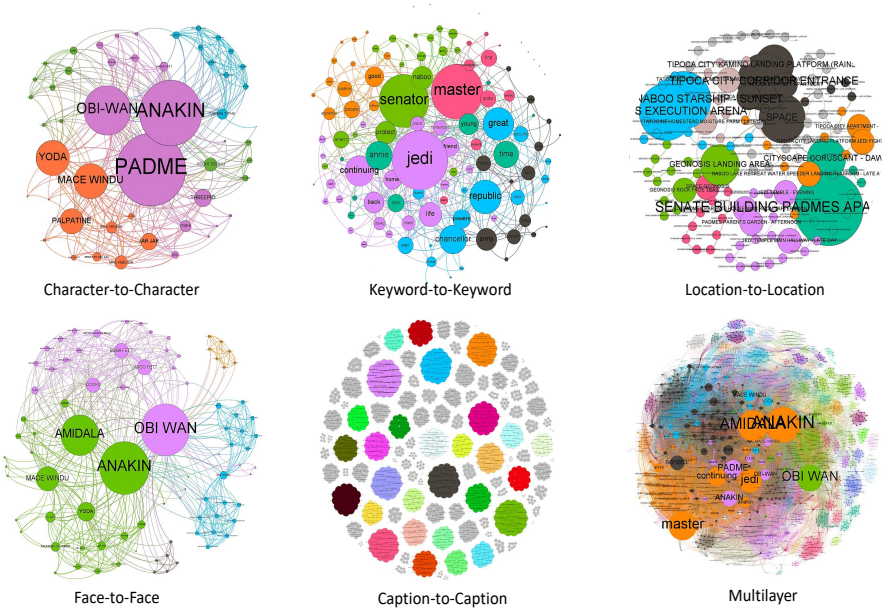

FIGURE A.2 – Visualisation des communautès dans les différentes couches de l'Episode II -L'attaque des clones (2002)Lucas (2002). La taille du noeud correspond à son degré.

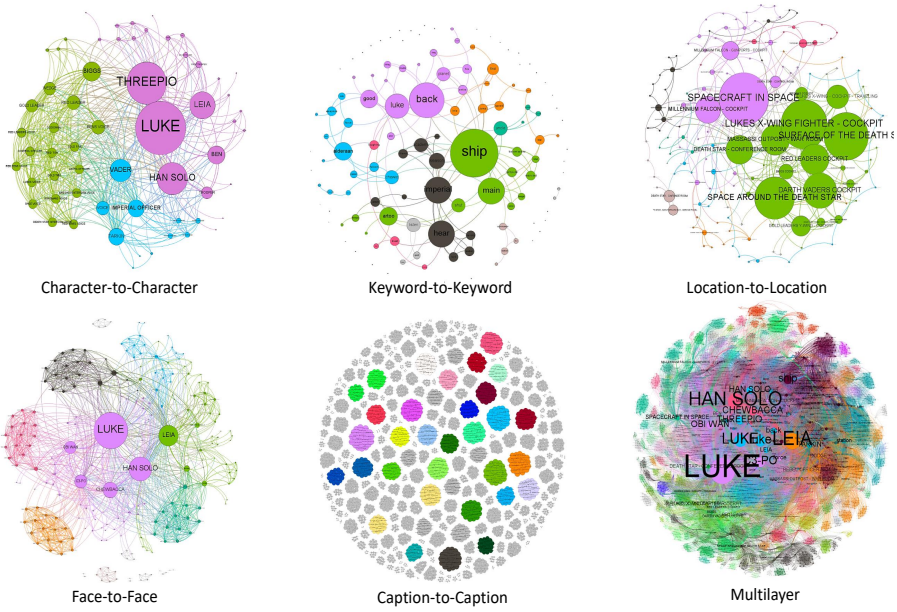

FIGURE A.3 – Visualisation des communautès dans les différentes couches de l'Episode IV -Un nouvel espoir (1977)Lucas (1977). La taille du noeud correspond à son degré.

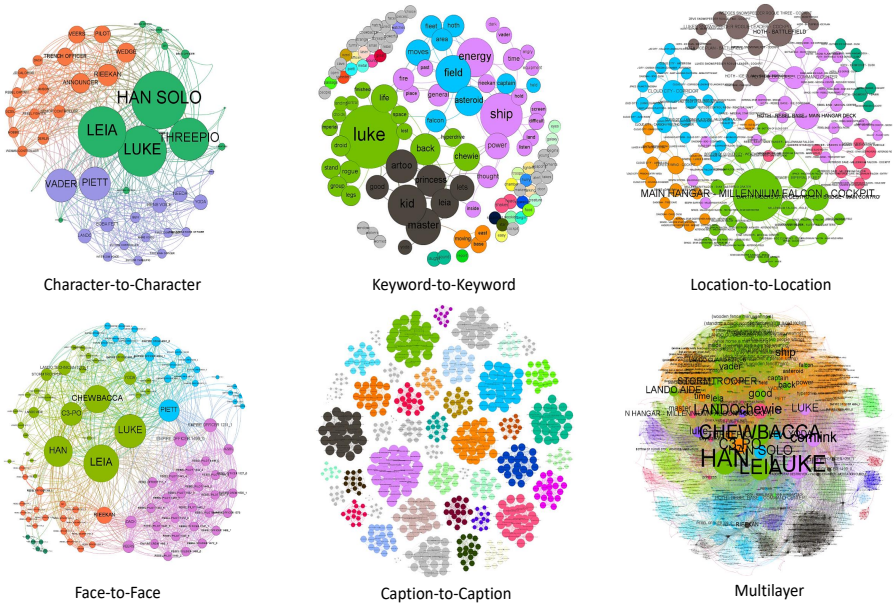

FIGURE A.4 – Visualisation des communautès dans les différentes couches de l'Episode V -L'empire contre-attaque (1980)Lucas (1980). La taille du noeud correspond à son degré.

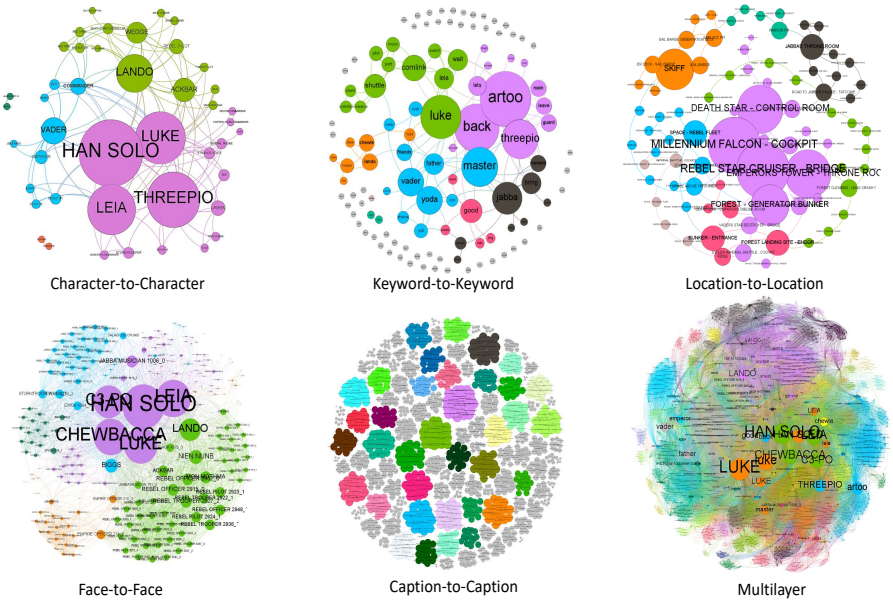

FIGURE A.5 – Visualisation des communautès dans les différentes couches de l'Episode VI -Le retour du Jedi (1983)Lucas (1983). La taille du noeud correspond à son degré.

EPISODE	PERSONNAGES							
	I.S	SCORE	B	SCORE	D	SCORE	Ei	SCORE
SW1	Q.GON	1,00	Q.GON	365,29	Q.GON	188	Q.GON	1,00
	ANAKIN	2,33	J.JAR	262,28	ANAKIN	151	ANAKIN	0,77
	J.JAR	2,67	ANAKIN	256,84	J.JAR	131	J.JAR	0,74
	O.WAN	4,33	O.WAN	145,97	PADME	110	O.WAN	0,70
	PADME	4,67	PADME	124,22	O.WAN	109	PADME	0,63
	PANAKA	6,67	NUTE	109,11	PANAKA	78	PANAKA	0,46
	NUTE	7,00	PALPATINE	105,27	NUTE	61	SHMI	0,31
	PALPATINE	8,33	PANAKA	54,91	PALPATINE	48	NUTE	0,28
	R.OLIE	9,67	DOFINE	10,99	SHMI	42	R.OLIE	0,24
	SHMI	10,67	R.OLIE	9,23	R.OLIE	39	PALPATINE	0,17
SW2	PADME	1,00	PADME	380,80	PADME	154	PADME	1,00
	ANAKIN	2,00	ANAKIN	259,02	ANAKIN	139	ANAKIN	0,95
	O.WAN	3,00	O.WAN	173,52	O.WAN	105	O.WAN	0,75
	M.WINDU	4,33	YODA	42,04	M.WINDU	75	M.WINDU	0,60
	YODA	4,67	M.WINDU	32,98	YODA	75	YODA	0,57
	PALPATINE	6,33	C3-PO	10,92	PALPATINE	54	PALPATINE	0,42
	C3-PO	6,67	PALPATINE	10,58	C3-PO	38	C3-PO	0,33
	J.JAR	8,33	C.TYPHO	10,57	J.JAR	36	J.JAR	0,30
	C.DOOKU	9,33	J.JAR	5,11	C.DOOKU	29	C.DOOKU	0,26
	OWEN	11,00	C.DOOKU	2,91	M.AMEDDA	29	M.AMEDDA	0,22
SW3	O.WAN	1,00	O.WAN	138,18	O.WAN	108	O.WAN	1,00
	PALPATINE	2,67	PADME	129,22	PALPATINE	99	ANAKIN	0,94
	ANAKIN	3,00	PALPATINE	102,65	ANAKIN	98	PALPATINE	0,92
	PADME	3,33	ANAKIN	91,95	PADME	96	PADME	0,83
	YODA	5,00	YODA	77,58	YODA	88	YODA	0,82
	B.ORGANA	6,33	G. GRIEVOUS	60,93	B.ORGANA	64	B.ORGANA	0,50
	D.VADER	7,67	B.ORGANA	57,94	D.VADER	54	D.VADER	0,48
	N.GUNRAY	8,67	N.GUNRAY	39,60	N.GUNRAY	54	M.WINDU	0,46
	G. GRIEVOUS	10,00	D.VADER	16,51	M.AMEDDA	46	C.C CODY	0,44
	M.AMEDDA	10,67	M.MOTHMA	15,24	GUARD	46	N.GUNRAY	0,43
SW4	LUKE	1,00	LUKE	385,33	LUKE	154	LUKE	1,00
	C3-PO	2,00	C3-PO	271,48	C3-PO	121	C3-PO	0,83
	H.SOLO	3,00	H.SOLO	179,11	H.SOLO	92	H.SOLO	0,70
	LEIA	4,00	LEIA	132,40	LEIA	77	LEIA	0,60
	VADER	5,33	VADER	127,91	VADER	64	BEN	0,50
	BIGGS	6,67	BIGGS	79,29	BIGGS	54	VADER	0,32
	I.OFFICER	8,67	CHIEF	58,00	BEN	51	TROOPER	0,28
	BEN	8,67	TARKIN	51,90	I.OFFICER	49	BIGGS	0,26
	TARKIN	9,33	I.OFFICER	42,03	TARKIN	47	I.OFFICER	0,24
	R.LEADER	11,33	R.LEADER	23,65	B.VOICE	39	B.VOICE	0,22
SW5	H.SOLO	1,00	H.SOLO	158,34	H.SOLO	108	H.SOLO	1,00
	LUKE	2,67	LUKE	100,45	LEIA	95	LEIA	0,91
	LEIA	2,67	C3-PO	83,76	LUKE	95	LUKE	0,89
	C3-PO	3,67	LEIA	82,81	C3-PO	79	C3-PO	0,80
	PIETT	5,00	PIETT	44,48	PIETT	62	PIETT	0,55
	VADER	6,00	VADER	44,48	VADER	62	VADER	0,55
	RIEEKAN	7,33	DERLIN	11,92	RIEEKAN	41	RIEEKAN	0,38
	ANNOUNCER	8,33	RIEEKAN	4,84	ANNOUNCER	37	ANNOUNCER	0,33
	WEDGE	9,33	ANNOUNCER	4,56	WEDGE	37	WEDGE	0,33
	VEERS	10,33	WEDGE	4,56	VEERS	33	VEERS	0,26
SW6	H.SOLO	1,33	LANDO	199,35	H.SOLO	84	H.SOLO	1,00
	C3-PO	2,67	H.SOLO	174,41	C3-PO	68	C3-PO	0,86
	LUKE	3,33	LUKE	115,15	LUKE	64	LEIA	0,79
	LANDO	3,67	C3-PO	106,90	LEIA	61	LUKE	0,77
	LEIA	4,00	LEIA	105,88	LANDO	44	LANDO	0,35
	VADER	6,00	VADER	62,55	VADER	28	VADER	0,25
	ACKBAR	8,00	D.S.CONTROLLER	39,97	ACKBAR	24	ACKBAR	0,24
	WEDGE	8,67	COMMANDER	30,52	WEDGE	21	JABBA	0,14
	COMMANDER	9,00	WEDGE	28,81	COMMANDER	14	WEDGE	0,11
	JABBA	10,33	ACKBAR	26,95	JABBA	12	COMMANDER	0,11

TABLEAU A.1 – 10 premiers noeuds triés selon leur score de centralité par épisode pour la couche des personnages.

EPISODE	MOTS-CLÉS							
	I.S	SCORE	B	SCORE	D	SCORE	Ei	SCORE
SW1	federation	1,67	jedi	1003,67	federation	40	queen	1,00
	jedi	3,00	federation	785,34	jedi	39	federation	0,96
	queen	4,00	naboo	627,31	queen	37	senate	0,91
	naboo	6,00	master	623,89	chancellor	34	chancellor	0,90
	senate	6,00	time	588,05	people	34	people	0,87
	people	7,33	anakin	535,60	senate	33	jedi	0,84
	time	7,67	stay	523,58	naboo	33	back	0,82
	chancellor	8,00	queen	496,22	back	32	naboo	0,81
	back	8,33	senate	493,58	time	32	time	0,71
	stay	9,67	back	468,84	stay	23	amidala	0,54
SW2	jedi	1,00	jedi	863,43	jedi	55	jedi	1,00
	master	2,00	master	734,86	master	51	master	0,99
	senator	3,00	senator	700,21	senator	49	senator	0,83
	republic	4,00	republic	435,81	republic	31	republic	0,53
	great	5,33	continuing	411,30	great	30	great	0,51
	continuing	6,33	great	398,98	continuing	28	chancellor	0,49
	chancellor	7,33	good	337,45	chancellor	26	yoda	0,48
	annie	9,33	annie	243,04	annie	26	continuing	0,47
	time	9,67	chancellor	214,06	time	24	protect	0,46
	life	11,67	time	208,90	life	22	time	0,45
SW3	anakin	1,33	anakin	2050,22	anakin	100	jedi	1,00
	jedi	1,67	jedi	1537,64	jedi	97	anakin	0,92
	chancellor	4,33	force	602,74	master	54	council	0,73
	master	4,33	chancellor	578,23	chancellor	52	master	0,64
	force	5,33	time	481,96	council	49	chancellor	0,54
	time	7,33	master	466,42	force	45	windu	0,51
	council	7,67	ship	321,34	windu	41	force	0,47
	windu	8,33	back	281,75	time	38	great	0,44
	republic	10,00	artoo	276,22	republic	36	time	0,41
	yoda	11,67	republic	259,30	great	35	dark	0,35
SW4	ship	1,00	ship	1215,21	ship	22	ship	1,00
	han	2,33	han	714,54	han	16	main	0,67
	hear	4,33	imperial	482,18	main	12	han	0,63
	main	4,33	luke	440,96	hear	12	hear	0,44
	imperial	4,67	hear	398,85	imperial	12	luke	0,39
	luke	5,00	chewie	319,65	luke	11	imperial	0,37
	chewie	9,67	artoo	303,53	good	8	planet	0,33
	good	10,00	main	265,28	artoo	8	computer	0,30
	artoo	11,67	good	240,18	alderaan	8	transmissions	0,30
	shut	14,67	uncle	237,26	season	7	entire	0,27
SW5	luke	1,00	luke	561,06	luke	27	luke	1,00
	master	4,33	ship	448,13	ship	17	kid	0,92
	ship	4,67	chewie	230,04	kid	15	master	0,83
	kid	5,67	energy	216,38	energy	14	artoo	0,78
	artoo	6,00	master	213,43	master	13	princess	0,70
	energy	7,67	time	213,00	field	13	leia	0,63
	han	9,00	artoo	129,52	artoo	11	good	0,57
	chewie	10,00	fire	109,94	princess	10	han	0,56
	field	10,33	moving	108,00	han	10	field	0,50
	princess	11,67	han	91,63	leia	9	ship	0,50
SW6	han	2,00	han	388,80	artoo	22	artoo	1,00
	artoo	2,33	father	279,22	back	19	C3-PO	0,85
	luke	3,67	luke	277,65	luke	17	han	0,81
	jabba	5,33	jabba	220,21	C3-PO	15	master	0,74
	master	5,67	artoo	210,86	master	15	luke	0,71
	C3-PO	8,67	friends	170,27	jabba	12	jabba	0,46
	vader	9,33	shuttle	165,90	vader	10	lets	0,38
	father	9,67	master	146,94	yoda	10	leia	0,37
	yoda	12,33	good	146,48	father	9	bring	0,35
	good	12,67	vader	133,84	good	8	wait	0,34

Tableau A.2 – 10 premiers noeuds triés selon leur score de centralité par épisode pour la couche des mots-clés.

EPISODE	LIEUX							
	I.S	SCORE	B	SCORE	D	SCORE	Ei	SCORE
SW1	FBB	1,00	FBB	3250,47	FBB	14	FBB	1,00
	NGP	4,00	FBCR	2898,56	TCH	10	FBHOB	0,69
	TCH	4,67	NSQC	1656,26	NGP	8	TCH	0,48
	TDNS	6,67	TDNS	1649,20	TDNS	8	NGP	0,38
	FBCR	9,33	NGP	1448,21	AHMR	8	SCU	0,38
	AHMR	11,33	NSMA	1421,79	NSCS	6	FBH	0,34
	NSMA	11,67	NFCS	1416,31	MSAVP	6	FBMB	0,34
	SCU	12,33	NSC	1198,44	MER	6	NSCS	0,30
	NSC	13,00	TCH	1112,97	NSMA	6	FBCR	0,27
	NPTR	14,67	AHMR	1023,31	NSC	6	NFCS	0,25
SW2	CNSS	3,00	SPACE	3320,27	SBPAB	10	TCCE	1,00
	SBPAB	3,67	CCD	2503,83	TCCE	8	CNSS	0,94
	SPACE	4,00	SBPAB	2088,31	CNSS	8	GEA	0,93
	TCKLP	5,33	CNSS	1637,55	GEA	8	TCKLP	0,87
	CCD	6,00	GLA	1519,72	SPACE	7	TCFA	0,68
	TCCE	6,33	TCKLP	1198,69	TCKLP	6	SPACE	0,56
	GLA	8,00	TDHMF	1082,57	CCD	6	SBPAB	0,53
	THMF	10,00	CMSC	1061,00	GLA	6	TC	0,46
	GEA	10,67	THMF	1025,23	TCFA	4	CCD	0,38
	TC	15,33	CJTCC	1005,60	TC	4	THMF	0,27
SW3	PJTC	1,33	PJTC	4411,93	MMCC	14	PJTC	1,00
	MMCC	2,67	ULP	3739,91	PJTC	12	MFJT	0,90
	MCP	11,00	ASH	3128,93	MFJT	6	MMCC	0,81
	CSCMA	11,33	MMCC	2927,78	MCP	6	CSCMA	0,42
	ASH	12,00	CO	2575,75	OBS	6	CCPD	0,39
	OBS	13,00	IDC	2530,00	ASH	6	SC	0,38
	PJTC	13,00	UCRGC	2511,25	LPCIRC	6	SCC	0,33
	ULP	13,33	OBS	2461,83	ULP	6	CSACHO	0,28
	IDC	15,33	BOOC	2431,25	CSCMA	4	MCP	0,27
	LPN	17,00	CJTH	2351,25	CCPD	4	MLP	0,26
SW4	SIS	3,00	SIS	5427,41	SIS	62	LXFC	1,00
	LXFC	3,67	DSCR	2316,40	LXFC	58	SOTDS	0,99
	DSCR	6,33	MFC	2062,13	SOTDS	56	DVC	0,84
	SATDS	7,00	DSH	1134,68	SATDS	50	SATDS	0,84
	MFC	7,67	MFGC	1027,83	DVC	42	RLC	0,68
	MOWR	9,00	TDW	959,43	RLC	36	MOWR	0,68
	SOTDS	9,00	TLH	955,53	MOWR	36	SIS	0,64
	MFGC	10,33	LXFC	820,23	DSCR	34	LXWCT	0,61
	RLC	12,67	DSCOR	696,01	MFC	32	DSCR	0,48
	DSCOR	13,00	SPACE,	685,76	LXWCT	28	GLYWC	0,39
SW5	MHMFC	1,00	MHMFC	4794,51	MHMFC	54	MHMFC	1,00
	HB	4,67	HRBMHD	3148,16	HB	22	DVSDBMCD	0,63
	HRBMHD	5,00	CCC	1063,52	DVSDBMCD	20	SIF	0,56
	HRBCC	8,00	HRBCC	1005,74	HRBMHD	20	HB	0,43
	BOCCWVD	8,00	CCLPMF	958,96	LSRLC	18	BOCCWVD	0,40
	DVSDBMCD	8,33	LXWC	899,89	HRBCC	16	MFH	0,37
	SIF	9,33	BOCCWVD	882,75	CCC	16	WSRTC	0,36
	HIPST	12,00	HB	596,33	SIF	12	LSRLC	0,34
	MFGAC	13,67	RBMC	570,07	WSRTC	12	HRBMHD	0,31
	LSRLC	16,33	HIPST	539,83	HIPST	12	MFSQ	0,28
SW6	MFC	1,00	MFC	1157,72	MFC	14	MFC	1,00
	DSCOR	2,67	DSCOR	885,89	RSCB	14	RSCB	0,95
	RSCB	3,67	DSMDB	809,48	DSCOR	12	DSCOR	0,74
	ETTR	4,67	RTJPT	776,48	ETTR	12	ETTR	0,73
	SKI	7,00	SAT	752,92	SKI	10	SKI	0,68
	SRF	7,67	ETTR	728,46	FGB	10	FGB	0,36
	DSMDB	8,33	RSCB	673,07	SRF	6	SRF	0,33
	FGB	9,00	DSLA	667,95	JTR	6	JTR	0,33
	RTJPT	9,33	SRF	617,25	FLSE	6	FLSE	0,29
	JTR	9,67	DS	615,95	BE	6	BE	0,29

TABLEAU A.3 – 10 premiers noeuds triés selon leur score de centralité par épisode pour la couche des lieux.

EPISODE	VISAGES							
	I.S	SCORE	B	SCORE	D	SCORE	Ei	SCORE
SW1	Q.GON	1,00	Q.GON	4663,97	Q.GON	452	Q.GON	1,00
	A.DOPPELGANGER	2,00	A.DOPPELGANGER	4588,02	A.DOPPELGANGER	415	A.DOPPELGANGER	0,91
	O.WAN	3,67	ANAKIN	4299,55	O.WAN	388	O.WAN	0,83
	ANAKIN	4,00	J.JAR	3846,81	ANAKIN	376	J.JAR	0,80
	J.JAR	4,33	O.WAN	3790,13	J.JAR	373	ANAKIN	0,75
	C.PANAKA	6,00	C.PANAKA	1146,58	C.PANAKA	277	C.PANAKA	0,62
	SHMI	7,67	SHMI	1015,44	PADME	204	PADME	0,44
	PADME	7,67	J.CREW	790,36	SHMI	158	SHMI	0,32
	RUNE	10,00	PADME	727,06	RUNE	144	D.MAUL	0,25
	D.MAUL	10,67	RUNE	711,61	D.MAUL	137	PALPATINE	0,22
SW2	ANAKIN	1,33	OBI WAN	1882,93	ANAKIN	187	ANAKIN	1,00
	OBI WAN	2,00	ANAKIN	882,57	OBI WAN	184	AMIDALA	0,89
	AMIDALA	2,67	AMIDALA	291,28	AMIDALA	143	OBI WAN	0,82
	M.WINDU	4,00	M.WINDU	165,97	M.WINDU	84	M.WINDU	0,46
	YODA	5,00	YODA	112,84	YODA	64	YODA	0,38
	J.FETT	7,00	Z.WESSEL	27,92	J.FETT	53	J.FETT	0,31
	B.FETT	8,33	P.FOLLOWER	25,97	B.FETT	51	DOOKU	0,29
	KI-ADI	8,67	KI-ADI	22,19	DOOKU	50	B.FETT	0,29
	DOOKU	8,67	J.FETT	18,03	KI-ADI	34	KI-ADI	0,20
	P.FOLLOWER	10,33	B.FETT	18,03	P.FOLLOWER	30	DORME	0,17
SW3	ANAKIN	1,33	OBI WAN	543,45	ANAKIN	151	ANAKIN	1,00
	OBI WAN	1,67	ANAKIN	492,93	OBI WAN	133	OBI WAN	0,87
	PALPATINE	3,00	PALPATINE	435,45	PALPATINE	110	PALPATINE	0,77
	YODA	4,33	GRIEVOUS	276,76	YODA	79	YODA	0,42
	M.WINDU	5,33	YODA	172,69	M.WINDU	64	M.WINDU	0,39
	GRIEVOUS	6,00	M.WINDU	147,74	B.B.ORGANA	60	B.B.ORGANA	0,33
	B.B.ORGANA	6,33	B.B.ORGANA	77,49	GRIEVOUS	34	GRIEVOUS	0,22
	CHEWBACCA	8,33	NUTE GUARD	29,72	CHEWBACCA	34	CHEWBACCA	0,15
	C.C CODY	9,33	CHEWBACCA	11,56	C.C CODY	30	C.C CODY	0,14
	NUTE GUARD	11,33	C.C CODY	10,13	JEDI 1	27	C.CROWD 1	0,12
SW4	LUKE	1,00	LUKE	9400,87	LUKE	333	LUKE	1,00
	H.SOLO	2,00	H.SOLO	4404,33	H.SOLO	220	H.SOLO	0,86
	LEIA	3,00	LEIA	3561,68	LEIA	188	LEIA	0,65
	CHEWBACCA	4,67	C3-PO	1368,35	CHEWBACCA	141	CHEWBACCA	0,59
	C3-PO	5,00	TARKIN	1107,43	C3-PO	117	OBI WAN	0,56
	OBI WAN	6,00	CHEWBACCA	984,55	OBI WAN	113	C3-PO	0,40
	TARKIN	8,33	OBI WAN	718,53	DODGE	51	DODGE	0,17
	DODGE	9,33	STORMTROOPER 1	388,00	R.LEADER	44	R.OFFICER	0,13
	BIGGS	11,00	E.OFFICER 5	253,44	TARKIN	44	R.LEADER	0,13
	R.LEADER	11,00	STORMTROOPER 2	243,35	BIGGS	42	BIGGS	0,12
SW5	LEIA	1,00	LEIA	1256,37	LEIA	191	LEIA	1,00
	HAN	2,33	HAN	916,94	LUKE	168	HAN	0,93
	LUKE	2,67	LUKE	758,23	HAN	151	LUKE	0,92
	CHEWBACCA	4,33	PIETT	651,33	CHEWBACCA	137	CHEWBACCA	0,86
	PIETT	5,33	CHEWBACCA	254,99	C3-PO	110	C3-PO	0,67
	C3-PO	6,00	RIEEKAN	244,38	PIETT	103	PIETT	0,46
	RIEEKAN	7,33	E.OFFICER 2	156,41	RIEEKAN	74	L.TECHNICIAN	0,38
	E.OFFICER 1	9,67	C3-PO	151,24	E.OFFICER 1	60	YODA	0,36
	L.TECHNICIAN	10,00	OZZEL	60,81	E.OFFICER 2	56	LANDO AIDE	0,30
SW6	H.SOLO	1,00	H.SOLO	4136,49	H.SOLO	316	H.SOLO	1,00
	LEIA	2,33	LUKE	3950,32	LEIA	265	LEIA	0,89
	LUKE	3,33	LEIA	2744,02	CHEWBACCA	259	CHEWBACCA	0,83
	CHEWBACCA	3,33	CHEWBACCA	2670,80	LUKE	254	LUKE	0,82
	C3-PO	5,00	C3-PO	2219,51	C3-PO	232	C3-PO	0,72
	LANDO	6,33	E.OFFICER 4	1812,93	LANDO	158	LANDO	0,42
	J.MUSICIAN 1	8,67	LANDO	1641,57	N.NUNB	98	J.MUSICIAN 1	0,28
	BIGGS	10,33	E.OFFICER 3	1310,43	BIGGS	96	N.NUNB	0,23
	N.NUNB	11,33	EWOK 1	930,11	J.MUSICIAN 1	84	ACKBAR	0,21
	ACKBAR	12,67	J.MUSICIAN 1	688,06	M.MOTHMA	84	M.MOTHMA	0,20

Tableau A.4 – 10 premiers noeuds triés selon leur score de centralité par épisode pour la couche des visages.

EPISODE	CAPTIONS							
	I.S	SCORE	B	SCORE	D	SCORE	Ei	SCORE
SW1	(a,gray,jacket,man,sitting)	1	(a,gray,jacket,man,sitting)	0	(a,gray,jacket,man,sitting)	33	(a,gray,jacket,man,sitting)	1
	(brown,and,white,suitcase,man)	2	(brown,and,white,suitcase,man)	0	(brown,and,white,suitcase,man)	33	(brown,and,white,suitcase,man)	1
	(holding,a,dog,woman,wearing)	3	(holding,a,dog,woman,wearing)	0	(holding,a,dog,woman,wearing)	33	(holding,a,dog,woman,wearing)	1
	(teddy,bear,a,person,sitting)	4	(teddy,bear,a,person,sitting)	0	(teddy,bear,a,person,sitting)	33	(teddy,bear,a,person,sitting)	1
	(blue,and,white,striped,shirt)	5	(blue,and,white,striped,shirt)	0	(blue,and,white,striped,shirt)	33	(blue,and,white,striped,shirt)	1
	(woman,wearing,a,red,coat)	6	(woman,wearing,a,red,coat)	0	(woman,wearing,a,red,coat)	33	(woman,wearing,a,red,coat)	1
	(black,jacket,woman,wearing,a)	7	(black,jacket,woman,wearing,a)	0	(black,jacket,woman,wearing,a)	33	(black,jacket,woman,wearing,a)	1
	(dog,a,black,metal,railing)	8	(dog,a,black,metal,railing)	0	(dog,a,black,metal,railing)	33	(dog,a,black,metal,railing)	1
	(brown,coat,a,person,standing)	9	(brown,coat,a,person,standing)	0	(brown,coat,a,person,standing)	33	(brown,coat,a,person,standing)	1
	(black,metal,railing,the,man)	10	(black,metal,railing,the,man)	0	(black,metal,railing,the,man)	33	(black,metal,railing,the,man)	1
SW2	(blue,jacket,a,man,wearing)	1	(blue,jacket,a,man,wearing)	0	(blue,jacket,a,man,wearing)	24	(blue,jacket,a,man,wearing)	1
	(chair,a,red,traffic,light)	2	(chair,a,red,traffic,light)	0	(chair,a,red,traffic,light)	24	(chair,a,red,traffic,light)	1
	(blue,jeans,a,silver,metal)	3	(blue,jeans,a,silver,metal)	0	(blue,jeans,a,silver,metal)	24	(blue,jeans,a,silver,metal)	1
	(brown,hair,a,man,wearing)	4	(brown,hair,a,man,wearing)	0	(brown,hair,a,man,wearing)	24	(brown,hair,a,man,wearing)	1
	(black,shirt,a,woman,standing)	5	(black,shirt,a,woman,standing)	0	(black,shirt,a,woman,standing)	24	(black,shirt,a,woman,standing)	1
	(white,plastic,chair,the,chair)	6	(white,plastic,chair,the,chair)	0	(white,plastic,chair,the,chair)	24	(white,plastic,chair,the,chair)	1
	(metal,phone,red,light,on)	7	(metal,phone,red,light,on)	0	(metal,phone,red,light,on)	24	(metal,phone,red,light,on)	1
	(black,jacket,a,man,wearing)	8	(black,jacket,a,man,wearing)	0	(black,jacket,a,man,wearing)	24	(black,jacket,a,man,wearing)	1
	(computer,monitor,a,silver,metal)	9	(computer,monitor,a,silver,metal)	0	(computer,monitor,a,silver,metal)	24	(computer,monitor,a,silver,metal)	1
	(hair,a,man,wearing,glasses)	10	(hair,a,man,wearing,glasses)	0	(hair,a,man,wearing,glasses)	24	(hair,a,man,wearing,glasses)	1
SW3	(gray,jacket,a,woman,wearing)	1	(gray,jacket,a,woman,wearing)	0	(gray,jacket,a,woman,wearing)	31	(gray,jacket,a,woman,wearing)	1
	(red,a,white,plastic,chair)	2	(red,a,white,plastic,chair)	0	(red,a,white,plastic,chair)	31	(red,a,white,plastic,chair)	1
	(computer,mouse,a,boy,standing)	3	(computer,mouse,a,boy,standing)	0	(computer,mouse,a,boy,standing)	31	(computer,mouse,a,boy,standing)	1
	(brown,shirt,a,woman,wearing)	4	(brown,shirt,a,woman,wearing)	0	(brown,shirt,a,woman,wearing)	31	(brown,shirt,a,woman,wearing)	1
	(woman,wearing,a,striped,shirt)	5	(woman,wearing,a,striped,shirt)	0	(woman,wearing,a,striped,shirt)	31	(woman,wearing,a,striped,shirt)	1
	(shirt,red,and,white,fire)	6	(shirt,red,and,white,fire)	0	(shirt,red,and,white,fire)	31	(shirt,red,and,white,fire)	1
	(red,and,white,fire,hydrant)	7	(red,and,white,fire,hydrant)	0	(red,and,white,fire,hydrant)	31	(red,and,white,fire,hydrant)	1
	(man,wearing,a,striped,shirt)	8	(man,wearing,a,striped,shirt)	0	(man,wearing,a,striped,shirt)	31	(man,wearing,a,striped,shirt)	1
	(white,fire,hydrant,a,woman)	9	(white,fire,hydrant,a,woman)	0	(white,fire,hydrant,a,woman)	31	(white,fire,hydrant,a,woman)	1
	(woman,wearing,a,black,hat)	10	(woman,wearing,a,black,hat)	0	(woman,wearing,a,black,hat)	31	(woman,wearing,a,black,hat)	1
SW4	(and,white,motorcycle,red,lights)	1	(and,white,motorcycle,red,lights)	0	(and,white,motorcycle,red,lights)	26	(and,white,motorcycle,red,lights)	1
	(white,and,blue,train,car)	2	(white,and,blue,train,car)	0	(white,and,blue,train,car)	26	(white,and,blue,train,car)	1
	(woman,wearing,a,green,jacket)	3	(woman,wearing,a,green,jacket)	0	(woman,wearing,a,green,jacket)	26	(woman,wearing,a,green,jacket)	1
	(and,green,wall,green,leaves)	4	(and,green,wall,green,leaves)	0	(and,green,wall,green,leaves)	26	(and,green,wall,green,leaves)	1
	(red,plastic,bag,the,flowers)	5	(red,plastic,bag,the,flowers)	0	(red,plastic,bag,the,flowers)	26	(red,plastic,bag,the,flowers)	1
	(green,wall,green,leaves,on)	6	(green,wall,green,leaves,on)	0	(green,wall,green,leaves,on)	26	(green,wall,green,leaves,on)	1
	(wearing,sunglasses,a,black,bag)	7	(wearing,sunglasses,a,black,bag)	0	(wearing,sunglasses,a,black,bag)	26	(wearing,sunglasses,a,black,bag)	1
	(blue,train,car,a,pink)	8	(blue,train,car,a,pink)	0	(blue,train,car,a,pink)	26	(blue,train,car,a,pink)	1
	(motorcycle,red,lights,hanging,from)	9	(motorcycle,red,lights,hanging,from)	0	(motorcycle,red,lights,hanging,from)	26	(motorcycle,red,lights,hanging,from)	1
	(wearing,a,green,jacket,red)	10	(wearing,a,green,jacket,red)	0	(wearing,a,green,jacket,red)	26	(wearing,a,green,jacket,red)	1
SW5	(woman,wearing,a,green,jacket)	1	(woman,wearing,a,green,jacket)	0	(woman,wearing,a,green,jacket)	28	(woman,wearing,a,green,jacket)	1
	(black,bird,teddy,bear,is)	2	(black,bird,teddy,bear,is)	0	(black,bird,teddy,bear,is)	28	(black,bird,teddy,bear,is)	1
	(bird,teddy,bear,is,wearing)	3	(bird,teddy,bear,is,wearing)	0	(bird,teddy,bear,is,wearing)	28	(bird,teddy,bear,is,wearing)	1
	(silver,metal,bottle,the,water)	4	(silver,metal,bottle,the,water)	0	(silver,metal,bottle,the,water)	28	(silver,metal,bottle,the,water)	1
	(large,black,bird,teddy,bear)	5	(large,black,bird,teddy,bear)	0	(large,black,bird,teddy,bear)	28	(large,black,bird,teddy,bear)	1
	(brown,hair,a,man,wearing)	6	(brown,hair,a,man,wearing)	0	(brown,hair,a,man,wearing)	28	(brown,hair,a,man,wearing)	1
	(man,holding,a,frisbee,man)	7	(man,holding,a,frisbee,man)	0	(man,holding,a,frisbee,man)	28	(man,holding,a,frisbee,man)	1
	(snow,a,teddy,bear,sitting)	8	(snow,a,teddy,bear,sitting)	0	(snow,a,teddy,bear,sitting)	28	(snow,a,teddy,bear,sitting)	1
	(ground,a,blue,fire,hydrant)	9	(ground,a,blue,fire,hydrant)	0	(ground,a,blue,fire,hydrant)	28	(ground,a,blue,fire,hydrant)	1
	(fire,hydrant,a,woman,laying)	10	(fire,hydrant,a,woman,laying)	0	(fire,hydrant,a,woman,laying)	28	(fire,hydrant,a,woman,laying)	1
SW6	(black,shirt,a,person,wearing)	1	(black,shirt,a,person,wearing)	0	(black,shirt,a,person,wearing)	24	(black,shirt,a,person,wearing)	1
	(white,logo,on,blue,shirt)	2	(white,logo,on,blue,shirt)	0	(white,logo,on,blue,shirt)	24	(white,logo,on,blue,shirt)	1
	(background,black,and,white,striped)	3	(background,black,and,white,striped)	0	(background,black,and,white,striped)	24	(background,black,and,white,striped)	1
	(bench,person,wearing,a,black)	4	(bench,person,wearing,a,black)	0	(bench,person,wearing,a,black)	24	(bench,person,wearing,a,black)	1
	(man,wearing,a,green,jacket)	5	(man,wearing,a,green,jacket)	0	(man,wearing,a,green,jacket)	24	(man,wearing,a,green,jacket)	1
	(pair,of,glasses,woman,wearing)	6	(pair,of,glasses,woman,wearing)	0	(pair,of,glasses,woman,wearing)	24	(pair,of,glasses,woman,wearing)	1
	(green,jacket,a,person,wearing)	7	(green,jacket,a,person,wearing)	0	(green,jacket,a,person,wearing)	24	(green,jacket,a,person,wearing)	1
	(glasses,woman,wearing,a,black)	8	(glasses,woman,wearing,a,black)	0	(glasses,woman,wearing,a,black)	24	(glasses,woman,wearing,a,black)	1
	(black,jacket,two,men,sitting)	9	(black,jacket,two,men,sitting)	0	(black,jacket,two,men,sitting)	24	(black,jacket,two,men,sitting)	1
	(white,backpack,a,person,sitting)	10	(white,backpack,a,person,sitting)	0	(white,backpack,a,person,sitting)	24	(white,backpack,a,person,sitting)	1

TABLEAU A.5 – 10 premiers noeuds triés selon leur score de centralité par épisode pour la couche des captions.

EPISODE	MULTI-COUCHES							
	I.S	SCORE	B	SCORE	D	SCORE	Ei	SCORE
SW1	Q.GON	1,00	Q.GON	433248,14	Q.GON	2256	Q.GON	1,00
	A.DOPPELGANGER	2,00	A.DOPPELGANGER	401129,59	A.DOPPELGANGER	2039	A.DOPPELGANGER	0,87
	O.WAN	3,33	ANAKIN	273079,32	O.WAN	1729	O.WAN	0,79
	ANAKIN	4,33	O.WAN	211053,57	ANAKIN	1678	Q.GON(CA)	0,74
	Q.GON(CA)	4,67	Q.GON(CA)	192802,55	Q.GON(CA)	1568	J.JAR	0,72
	J.JAR	5,67	J.JAR	159027,64	J.JAR	1515	ANAKIN	0,71
	ANAKIN(CA)	7,33	ANAKIN(CA)	120153,41	ANAKIN(CA)	1143	C.PANAKA	0,53
	C.PANAKA	9,33	NUTE	69086,85	C.PANAKA	1119	ANAKIN(CA)	0,51
	J.JAR(CA)	10,00	time	65969,04	J.JAR(CA)	905	J.JAR(CA)	0,47
	time	10,67	jedi	57592,81	time	850	O.WAN(CA)	0,41
SW2	ANAKIN	1,00	ANAKIN	463839,77	ANAKIN	1701	ANAKIN	1,00
	AMIDALA	2,33	OBI WAN	393626,35	AMIDALA	1479	AMIDALA	0,99
	OBI WAN	3,67	AMIDALA	201725,55	master	1378	master	0,98
	master	5,00	YODA	42102,39	OBI WAN	1260	jedi	0,71
	PADME(CA)	6,33	PADME(CA)	41198,98	jedi	1149	OBI WAN	0,66
	jedi	6,33	CRG	40214,19	continuing	886	continuing	0,60
	ANAKIN(CA)	9,00	CHAC	38585,62	PADME(CA)	844	PADME(CA)	0,48
	continuing	9,67	GEA	36476,57	ANAKIN(CA)	739	ANAKIN(CA)	0,45
	YODA	11,33	master	36421,59	senator	695	senator	0,42
	M.WINDU	14,00	jedi	34697,99	M.WINDU	516	anakin	0,34
SW3	ANAKIN	2,33	ANAKIN	578498,39	anakin	3275	anakin	1,00
	OBI WAN	3,33	OBI WAN	255342,83	jedi	2971	jedi	0,88
	anakin	3,67	PALPATINE	87134,90	ANAKIN	2307	ANAKIN	0,72
	jedi	4,00	OBS	45047,47	OBI WAN	2006	OBI WAN	0,71
	PALPATINE	5,00	GRIEVOUS	43225,19	master	1756	PALPATINE	0,60
	YODA	6,67	YODA	29154,79	chancellor	1534	YODA	0,55
	master	8,33	PJTC	20039,23	PALPATINE	1502	master	0,52
	chancellor	10,33	jedi	19543,17	YODA	1286	chancellor	0,46
	B.B.ORGANA	10,67	anakin	19324,73	council	1208	council	0,40
	council	14,67	B.B.ORGANA	17976,12	time	953	B.B.ORGANA	0,38
SW4	LUKE	1,00	LUKE	2858888,34	LUKE	2390	LUKE	1,00
	LEIA	2,33	LEIA	1225068,04	LEIA	1471	H.SOLO	0,75
	H.SOLO	2,67	H.SOLO	830919,53	H.SOLO	1443	LEIA	0,69
	C3-PO	6,00	C3-PO	528229,34	LUKE(CA)	1125	luke	0,64
	LUKE(CA)	6,00	SIS	425281,08	luke	1001	LUKE(CA)	0,52
	luke	9,33	DVC	368387,68	C3-PO	891	CHEWBACCA	0,48
	C3-PO(CA)	10,33	TARKIN	337032,95	ship	813	OBI WAN	0,46
	OBI WAN	10,33	DSCR	279716,08	CHEWBACCA	740	C3-PO	0,44
	ship	11,33	LUKE(CA)	273334,38	C3-PO(CA)	722	ship	0,42
	CHEWBACCA	12,33	R.LEADER	254298,07	OBI WAN	691	C3-PO(CA)	0,38
SW5	HAN	1,33	LUKE	155127,69	HAN	1033	HAN	1,00
	LUKE	2,33	HAN	141670,46	LUKE	960	CHEWBACCA	0,95
	LEIA	3,00	LEIA	76706,60	LEIA	873	LEIA	0,87
	CHEWBACCA	4,00	PIETT	41321,26	CHEWBACCA	822	LUKE	0,86
	C3-PO	8,00	MHMFC	36654,91	chewie	621	chewie	0,81
	H.SOLO(CA)	9,33	CHEWBACCA	31167,80	C3-PO	620	LANDO	0,76
	comlink	9,67	H.SOLO(CA)	24251,44	comlink	616	comlink	0,70
	chewie	11,33	C3-PO(CA)	18933,41	LANDO	590	C3-PO	0,69
	LANDO	12,00	ZEF	18853,37	H.SOLO(CA)	544	vader	0,69
	LUKE(CA)	15,67	C3-PO	18550,29	ship	488	good	0,67
SW6	LUKE	1,00	LUKE	405309,67	LUKE	1495	LUKE	1,00
	H.SOLO	2,00	H.SOLO	229704,31	H.SOLO	1389	H.SOLO	0,99
	LEIA	3,67	LANDO	157953,49	luke	1157	LEIA	0,88
	luke	5,33	LEIA	145318,53	LEIA	1143	CHEWBACCA	0,81
	C3-PO	6,00	H.SOLO(CA)	111635,44	CHEWBACCA	1008	luke	0,79
	CHEWBACCA	6,00	C3-PO	94854,11	C3-PO	971	C3-PO	0,73
	H.SOLO(CA)	6,33	C3-PO(CA)	81765,04	H.SOLO(CA)	918	H.SOLO(CA)	0,56
	C3-PO(CA)	8,00	luke	67187,73	artoo	854	C3-PO(CA)	0,56
	LANDO	9,00	CHEWBACCA	67145,94	C3-PO(CA)	812	artoo	0,46
	artoo	11,67	shield	66515,42	LANDO	739	good	0,44

Tableau A.6 – 10 premiers noeuds triés selon leur score de centralité par épisode pour le réseau multi-couches.

NAME	ABBREVIATION
QUI-GON	Q.GON
JAR JAR	J.JAR
OBI-WAN	O.WAN
CAPT PANAKA	C.PANAKA
MACE WINDU	M.WINDU
PALPATINE	PALPATINE
COUNT DOOKU	C.DOOKU
HAN SOLO	H.SOLO
RED LEADER	R.LEADER
DEATH STAR CONTROLLER	D.S.CONTROLLER
BREHA BAIL ORGANA	B,B.ORGANA
BAIL ORGANA	B.ORGANA
GENERAL GRIEVOUS	G.GRIEVOUS
CLONE COMMANDER CODY	C.C.CODY
RIC OLIE	R.OLIE
JANGO FETT	J.FETT
BOBA FETT	B.FETT
CAPTAIN TYPHO	C.TYPHO
AMIDALA DOPPELGANGER 1	A.DOPPELGANGER
EMPIRE OFFICER 3	E.OFFICER 3
MAS AMEDDA	M.AMEDDA
REBEL OFFICER	R.OFFICER
LANDO TECHNICIAN	L.TECHNICIAN
EMPIRE OFFICER 1	E.OFFICER 1
EMPIRE OFFICER 2	E.OFFICER 2
JABBA MUSICIAN 1	J.MUSICIAN 1
REBEL OFFICER 2	R.OFFICER 2
NIEN NUNB	N.NUNB
DARTH VADER	D.VADER
NUTE GUNRAY	N.GUNRAY
MON MOTHMA	M.MOTHMA
IMPERIAL OFFICER	I.OFFICER
BENS VOICE	B,VOICE
JABBA CREW	J.CREW
PALPATINE FOLLOWER	P.FOLLOWER
CORUSCANT CROWD 1	C.CROWD 1
EMPIRE OFFICER 4	E.OFFICER 4
LANDO TECHNICIAN	L.TECHNICIAN
ZAM WESSEL	Z.WESSEL
DARTH MAUL	D.MAUL
EMPIRE OFFICER 5	E.OFFICER 5

TABLEAU A.7 – Table d'abréviation pour les noms des personnages et des visages

NAME	ABB
ALDERAAN STARCRUISER - HALLWAY	ASH
ANAKINS HOVEL - MAIN ROOM	AHMR
BAIL ORGANAS OFFICE - CORUSCANT	BOOC
BOTTOM OF CLOUD CITY - WEATHER VANE - DUSK	BOCCWVD
BOTTOM OF CLOUD CITY - WEATHER VANE - DUSK	BOCCWVD
BUNKER - ENTRANCE	BE
CHANCELLORS OFFICE	CO
CITYSCAPE - CORUSCANT - PRE-DAWN	CCPD
CITYSCAPE CORUSCANT - DAWN	CCD
CLOUD CITY - CORRIDOR	CCC
CLOUD CITY - LANDING PLATFORM - MILLENNIUM FALCON	CCLPMF
COCKPIT NABOO STARSHIP - SUNSET	CNSS
CORRIDOR - SENATE ARENA - CHANCELLORS HOLDING OFFICE	CSACHO
CORUSCANT - JEDI TEMPLE - HALLWAY	CJTH
CORUSCANT - SENATE CHAMBER - MAIN ARENA	CSCMA
CORUSCANT JEDI TEMPLE COUNCIL CHAMBER	CJTCC
CORUSCANT MAIN SENATE CHAMBER	CMSC
DARTH VADERS STAR DESTROYER - BRIDGE - MAIN CONTROL DECK	DVSDBMCD
DEATH STAR - CONFERENCE ROOM	DSCR
DEATH STAR - CONTROL ROOM	DSCOR
DEATH STAR - CONTROL ROOM	DSCR
DEATH STAR - HALLWAY	DSH
DEATH STAR - MAIN DOCKING BAY	DSMDB
DUNE SEA - LANDING AREA	DSLA
EMPERORS TOWER - THRONE ROOM	ETTR
FEDERATION BATTLESHIP - BRIDGE	FBB
FEDERATION BATTLESHIP - CONFERENCE ROOM	FBCR
FEDERATION BATTLESHIP - HALLWAY	FBH
FEDERATION BATTLESHIP - HALLWAY - OUTSIDE BRIDGE-	FBHOB
FEDERATION BATTLESHIP - MAIN BAY	FBMB
FOREST - GENERATOR BUNKER	FGB
FOREST LANDING SITE - ENDOR	FLSE
GEONOSIS EXECUTION ARENA	GEA
GEONOSIS LANDING AREA	GLA
GOLD LEADERS Y-WING - COCKPIT	GLYWC
HOTH - BATTLEFIELD	HB
HOTH - ICE PLAIN - SNOW TRENCH	HIPST
HOTH - REBEL BASE - COMMAND CENTER	HRBCC
HOTH - REBEL BASE - MAIN HANGAR DECK	HRBMHD
INDUSTRIAL DISTRICT - CORUSCANT	IDC
JABBAS THRONE ROOM	JTR
LANDING PLATFORM - CORUSCANT - IMPERIAL REHAB CENTER	LPCIRC
LANDING PLATFORM - NABOO SKIFF	LPN
LUKES SNOWSPEEDER ROGUE LEADER - COCKPIT	LSRLC
LUKES X-WING - COCKPIT	LXWC
LUKES X-WING - COCKPIT - TRAVELING	LXWCT
LUKES X-WING FIGHTER - COCKPIT	LXFC
MAIN FLOOR - JEDI TEMPLE	MFJT
MAIN HANGAR - MILLENNIUM FALCON - COCKPIT	MHMFC
MAIN HANGAR - MILLENNIUM FALCON - COCKPIT	MHMFC
MASSASSI OUTPOST - WAR ROOM	MOWR
MILLENNIUM FALCON - COCKPIT	MFC
MILLENNIUM FALCON - GIANT ASTEROID CRATER	MFGAC
MILLENNIUM FALCON - GUNPORTS - COCKPIT	MFGC
MILLENNIUM FALCON - HOLD	MFH
MOS ESPA - ARENA - VIEWING PLATFORM	MSAVP
MOS ESPA - RACETRACK	MER
MUSTAFAR - COLLECTION PANELS	MCP
MUSTAFAR - LANDING PLATFORM	MLP
MUSTAFAR - MAIN CONTROL CENTER	MMCC
NABOO FIGHTER - COCKPIT - SPACE	NFCS
NABOO GRASS PLAINS	NGP
NABOO PALACE - THRONE ROOM	NPTR
NABOO SPACECRAFT - COCKPIT	NSC
NABOO SPACECRAFT - MAIN AREA	NSMA
NABOO SPACECRAFT - QUEENS CHAMBERS	NSQC
NABOO STARFIGHTER - COCKPIT - SPACE	NSCS
OBI-WAN'S STARFIGHTER	OBS
PLAZA - JEDI TEMPLE - CORUSCANT	PJTC
PLAZA - JEDI TEMPLE - CORUSCANT-ROOM	PJTCR
REBEL BASE - MEDICAL CENTER	RBMC
REBEL STAR CRUISER - BRIDGE	RSCB
RED LEADERS COCKPIT	RLC
ROAD TO JABBAS PALACE - TATOOINE	RTJPT
SENATE BUILDING PADMES APARTMENT BEDROOM	SBPAB
SENATE CHAMBER	SC
SENATE CHAMBER - CORUSCANT	SCC
SKIFF	SKI
SPACE	SP
SPACE - IMPERIAL FLEET	SIF
SPACE - REBEL FLEET	SRF
SPACE ABOVE TATOOINEA	SAT
SPACE AROUND THE DEATH STAR	SATDS
SPACECRAFT IN SPACE	SIS
SUB COCKPIT - UNDERWATER	SCU
SURFACE OF THE DEATH STAR	SOTDS
TATOOINE - DESERT - NABOO SPACECRAFT	TDNS
TATOOINE - DESERT WASTELAND	TDW
TATOOINE - LARS HOMESTEAD	TLH
TATOOINE DESERT HOMESTEAD MOISTURE FARM	TDHMF
TATOOINE HOMESTEAD MOISTURE FARM	THMF
THEED - CENTRAL HANGER	TCH
TIPOCA CITY (RAINSTORM)	TC
TIPOCA CITY CORRIDOR ENTRANCE	TCCE
TIPOCA CITY FETT APARTMENT	TCFA
TIPOCA CITY KAMINO LANDING PLATFORM (RAINSTORM)	TCKLP
UTAPAU - CONFERENCE ROOM - GRAND CHAMBER	UCRGC
UTAPAU - LANDING PLATFORM	ULP
WEDGES SNOWSPEEDER ROGUE THREE - COCKPIT	WSRTC
MILLENNIUM FALCON - SLEEPING QUARTERS	MFSQ
SPACE - AIR BATTLE	SAB
BUNKER - CONTROL ROOM	BCR
VADERS STAR DESTROYER - BRIDGE	VSDB
CONFERENCE ROOM (GEONOSIS)	CRG
GEONOSIS HIGH AUDIENCE CHAMBER	CHAC

Tableau A.8 – Table d'abréviation pour les noms des lieux

PRODUCTION SCIENTIFIQUE

Les travaux réalisés et présentés dans ce mémoire ont été valorisés dans les publications indiquées ci-dessous :

Revues internationales (2)

1. **Y. Mourchid,** M. El Hassouni, H. Cherifi, "A general framework for complex network-based image segmentation." Multimedia Tools and Applications (2019) : 1-26.

2. **Y. Mourchid,** M. El Hassouni, H. Cherifi. "Image segmentation based on community detection approach". The International Journal of Computer Information Systems and Industrial Management Applications. ISSN, 2150-7988.

Papier soumis (1)

1. **Y. Mourchid**, B. Renoust, O. Roupin, L. Van, H. Cherifi, M. El Hassouni, "Movienet : A Movie Multilayer Network Model using Visual and Textual Semantic Cues", Soumis à Applied Network Science Journal

Conférences nationales et internationales (5)

1. M. Lafhel, **Y. Mourchid**, H. Cherifi, B. Renoust, M. EL Hassouni, "Ranking movies using multilayer networks", The 10th Conference on Network Modeling and Analysis November 06 - 08, 2019 Dijon, France, MARAMI 2019 (Papier Accepté).

2. **Y. Mourchid,** B. Renoust, M. El Hassouni, H. Cherifi (2018, December), "Multilayer Network Model of Movie Script", In International Conference on Complex Networks and their Applications (pp. 782-796). Springer, Cham.

3. **Y. Mourchid,** M. El Hassouni, H. Cherifi (2017, May), "Image Segmentation by Deep Community Detection Approach", In International Symposium on Ubiquitous Networking (pp. 607-618). Springer, Cham.

4. **Y. Mourchid,** M. El Hassouni, H. Cherifi, (2016, November). "An image segmentation algorithm based on community detection", In International Workshop on Complex Networks and their Applications (pp. 821-830). Springer, Cham.

5. **Y. Mourchid,** M. El Hassouni, H. Cherifi, "A new image segmentation approach using community detection algorithms", In 2015 15th International Conference on Intelligent Systems Design and Applications (ISDA) (pp. 648-653). IEEE.

6. **Y. Mourchid,** M. El Hassouni, H. Cherifi, "La Segmentation des Images par Les Réseaux Complexes", Journée URAC-Maroc, Rabat, 28 Novembre 2015.

BIBLIOGRAPHIE

ABIN, A. A., MAHDISOLTANI, F. et BEIGY, H. (2014). Wisecode : wise image segmentation based on community detection. *The Imaging Science Journal*, 62(6):327–336.

AL OMRAN, F. N. A. et TREUDE, C. (2017). Choosing an nlp library for analyzing software documentation : a systematic literature review and a series of experiments. pages 187–197.

ALBERT, R., JEONG, H. et BARABÁSI, A.-L. (1999). Internet : Diameter of the world-wide web. *nature*, 401(6749):130.

ARBELAEZ, P., MAIRE, M., FOWLKES, C. et MALIK, J. (2010). Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916.

ARENAS, A., FERNANDEZ, A. et GOMEZ, S. (2008). Analysis of the structure of complex networks at different resolution levels. *New journal of physics*, 10(5):053039.

BAO, P., ZHANG, L. et WU, X. (2005). Canny edge detection enhancement by scale multiplication. *IEEE transactions on pattern analysis and machine intelligence*, 27(9): 1485–1490.

BERGE, C. (2001). *The theory of graphs*. Courier Corporation.

BERNSTEIN, E. J. et AMIT, Y. (2005). Part-based statistical models for object classification and detection. 2:734–740.

BIOGLIO, L. et PENSA, R. G. (2017). Is this movie a milestone ? identification of the most influential movies in the history of cinema. pages 921–934.

BLEI, D. M., NG, A. Y. et JORDAN, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

BLONDEL, V. D., GUILLAUME, J.-L., LAMBIOTTE, R. et LEFEBVRE, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics : theory and experiment*, 2008(10):P10008.

BOLON, P., CHASSERY, J.-M., COCQUEREZ, J.-P., DEMIGNY, D., GRAFFIGNE, C., MONTANVERT, A., PHILIPP, S., ZÉBOUDJ, R., ZERUBIA, J. et MAÎTRE, H. (1995). *Analyse d'images : filtrage et segmentation*.

BRANDES, U. (2001). A faster algorithm for betweenness centrality. *Journal of mathematical sociology*, 25(2):163–177.

BRANDES, U., DELLING, D., GAERTLER, M., GORKE, R., HOEFER, M., NIKOLOSKI, Z. et WAGNER, D. (2007). On modularity clustering. *IEEE transactions on knowledge and data engineering*, 20(2):172–188.

CAI, Y., SHI, C., DONG, Y., KE, Q. et WU, B. (2011). A novel genetic algorithm for overlapping community detection. pages 97–108.

CANNY, J. (1987). A computational approach to edge detection. *In Readings in computer vision*, pages 184–203. Elsevier.

CAO, Q., SHEN, L., XIE, W., PARKHI, O. M. et ZISSERMAN, A. (2018). Vggface2 : A dataset for recognising faces across pose and age. pages 67–74.

CASTELLANO, B. (2012). PySceneDetect. Last accessed : 2019-06-20.

CAVNAR, W. B., TRENKLE, J. M. *et al.* (1994). N-gram-based text categorization. 161175.

CELEUX, G. (1985). The sem algorithm : a probabilistic teacher algorithm derived from the em algorithm for the mixture problem. *Computational statistics quarterly*, 2:73–82.

CELEUX, G. et GOVAERT, G. (1992). A classification em algorithm for clustering and two stochastic versions. *Computational statistics & Data analysis*, 14(3):315–332.

CHEN, B.-W., WANG, J.-C. et WANG, J.-F. (2009). A novel video summarization based on mining the story-structure and semantic relations among concept entities. *IEEE Transactions on Multimedia*, 11(2):295–312.

CHEN, D., LÜ, L., SHANG, M.-S., ZHANG, Y.-C. et ZHOU, T. (2012). Identifying influential nodes in complex networks. *Physica a : Statistical mechanics and its applications*, 391(4):1777–1787.

CHEN, R.-G., CHEN, C.-C. et CHEN, C.-M. (2019). Unsupervised cluster analyses of character networks in fiction : Community structure and centrality. *Knowledge-Based Systems*, 163:800–810.

CHEN, X., ZHENG, C., YAO, H. et WANG, B. (2017). Image segmentation using a unified markov random field model. *IET Image Processing*, 11(10):860–869.

CHRISTOUDIAS, C. M., GEORGESCU, B. et MEER, P. (2002). Synergism in low level vision. page 40150.

ÇIĞLA, C. et ALATAN, A. A. (2010). Efficient graph-based image segmentation via speeded-up turbo pixels. pages 3013–3016.

CLAUSET, A., NEWMAN, M. E. et MOORE, C. (2004). Finding community structure in very large networks. *Physical review E*, 70(6):066111.

COMANICIU, D. et MEER, P. (2002). Mean shift : A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (5):603–619.

CORRÊA JR, E. A., MARINHO, V. Q. et AMANCIO, D. R. (2019). Semantic flow in language networks. *arXiv preprint arXiv :1905.07595*.

COSCIA, M., GIANNOTTI, F. et PEDRESCHI, D. (2011). A classification for community discovery methods in complex networks. *Statistical Analysis and Data Mining : The ASA Data Science Journal*, 4(5):512–546.

DELVENNE, J.-C., YALIRAKI, S. N. et BARAHONA, M. (2010). Stability of graph communities across time scales. *Proceedings of the national academy of sciences*, 107(29):12755–12760.

DEMIRKESEN, C. et CHERIFI, H. (2008). A comparison of multiclass svm methods for real world natural scenes. pages 752–763.

DERICHE, R. (1987). Using canny's criteria to derive a recursively implemented optimal edge detector. *International journal of computer vision*, 1(2):167–187.

DING, C. H., HE, X., ZHA, H., GU, M. et SIMON, H. D. (2001). A min-max cut algorithm for graph partitioning and data clustering. pages 107–114.

DOMENICO, M., PORTER, M. et ARENAS, A. (2014). Multilayer analysis and visualization of networks. *J. Complex Netw*, 10.

DOMENICO, M., SOL-RIBALTA, A., OMODEI, E., GMEZ, S. et ARENAS, A. (2013). Centrality in interconnected multilayer networks.

DUCH, J. et ARENAS, A. (2005). Community detection in complex networks using extremal optimization. *Physical review E*, 72(2):027104.

ESTER, M., KRIEGEL, H.-P., SANDER, J. et XU, X. (1996). Density-based spatial clustering of applications with noise. 240.

EULER, L. (1741). Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, pages 128–140.

EVANGELOPOULOS, G., RAPANTZIKOS, K., POTAMIANOS, A., MARAGOS, P., ZLATINTSI, A. et AVRITHIS, Y. (2008). Movie summarization based on audiovisual saliency detection. *In 2008 15th IEEE International Conference on Image Processing*, pages 2528–2531. IEEE.

FALOUTSOS, M., FALOUTSOS, P. et FALOUTSOS, C. (1999). On power-law relationships of the internet topology. 29(4):251–262.

FELZENSZWALB, P. F. et HUTTENLOCHER, D. P. (2004). Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181.

FLINT, L. N. (1917). Newspaper writing in high schools : Containing an outline for the use of teachers.

FORSYTH, D. A. et PONCE, J. (2003). A modern approach. *Computer vision : a modern approach*, 17:21–48.

FORTUNATO, S. (2010). Community detection in graphs. *Physics reports*, 486(3-5):75–174.

FORTUNATO, S. et BARTHELEMY, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41.

GATH, I. et GEVA, A. B. (1989). Unsupervised optimal fuzzy clustering. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7):773–780.

GEMAN, S. et GEMAN, D. (1987). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *In Readings in computer vision*, pages 564–584. Elsevier.

GEORGESCU, B., SHIMSHONI, I. et MEER, P. (2003). Mean shift based clustering in high dimensions : A texture classification example. 3:456.

GIRVAN, M. et NEWMAN, M. E. (2002). Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826.

GONDRAN, M. et MINOUX, M. (1984). *Graphs and algorithms*. Wiley.

GONZALEZ, R. C. et WOODS, R. E. (1992). Digital image processing.

GORINSKI, P. J. et LAPATA, M. (2018). What's this movie about ? a joint neural network architecture for movie content analysis. pages 1770–1781.

GUIMERA, R., SALES-PARDO, M. et AMARAL, L. A. N. (2004). Modularity from fluctuations in random graphs and complex networks. *Physical Review E*, 70(2):025101.

GUO, Y., LIU, Y., OERLEMANS, A., LAO, S., WU, S. et LEW, M. S. (2016). Deep learning for visual understanding : A review. *Neurocomputing*, 187:27–48.

HARALICK, R. M. (1987). Digital step edges from zero crossing of second directional derivatives. *In Readings in Computer Vision*, pages 216–226. Elsevier.

HE, J., XIE, Y., LUAN, X., ZHANG, L. et ZHANG, X. (2018). Srn : The movie character relationship analysis via social network. pages 289–301.

HE, K., ZHANG, X., REN, S. et SUN, J. (2016). Deep residual learning for image recognition. pages 770–778.

HEIMO, T., KUMPULA, J. M., KASKI, K. et SARAMÄKI, J. (2008). Detecting modules in dense weighted networks with the potts method. *Journal of Statistical Mechanics : Theory and Experiment*, 2008(08):P08007.

HERLIN, I., BÉRÉZIAT, D., GIRAUDON, G., NGUYEN, C. et GRAFFIGNE, C. (1994). Segmentation of echocardiographic images with markov random fields. pages 200–206.

HUA, X.-S., LU, L., ZHANG, H.-j. et DISTRICT, H. (2005). A generic framework of user attention model and its application in video summarization. *IEEE Transaction on multimedia*, 7(5):907–919.

HUISMAN, H. et THIJSSEN, J. (1998). Adaptive texture feature extraction with application to ultrasonic image analysis. *Ultrasonic Imaging*, 20(2):132–148.

JHALA, A. (2008). Exploiting structure and conventions of movie scripts for information retrieval and text mining. pages 210–213.

JIANG, H. et LEARNED-MILLER, E. (2017). Face detection with the faster r-cnn. pages 650–657.

JOHNSON, J., KARPATHY, A. et FEI-FEI, L. (2016). Densecap : Fully convolutional localization networks for dense captioning. pages 4565–4574.

JOYCE, J. M. (2011). Kullback-leibler divergence. *International encyclopedia of statistical science*, pages 720–722.

JUNG, B., KWAK, T., SONG, J. et LEE, Y. (2004). Narrative abstraction model for story-oriented video. pages 828–835.

KADUSHIN, C. (2012). *Understanding social networks : Theories, concepts, and findings*. OUP USA.

KIPLING, R. (1902). Just so stories.

KIPLING, R. (1998). *Just so stories for little children*. OUP Oxford.

KIVELÄ, M., ARENAS, A., BARTHELEMY, M., GLEESON, J. P., MORENO, Y. et PORTER, M. A. (2014). Multilayer networks. *Journal of complex networks*, 2(3):203–271.

KNUTH, D. E. (1993). *The Stanford GraphBase : a platform for combinatorial computing*. AcM Press New York.

KOSCHÜTZKI, D., LEHMANN, K. A., PEETERS, L., RICHTER, S., TENFELDE-PODEHL, D. et ZLOTOWSKI, O. (2005). Centrality indices. *In Network analysis*, pages 16–61. Springer.

KRISHNA, R., ZHU, Y., GROTH, O., JOHNSON, J., HATA, K., KRAVITZ, J., CHEN, S., KALANTIDIS, Y., LI, L.-J., SHAMMA, D. A., BERNSTEIN, M. S. et FEI-FEI, L. (2017). Visual genome : Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

KURZHALS, K., JOHN, M., HEIMERL, F., KUZNECOV, P. et WEISKOPF, D. (2016). Visual movie analytics. *IEEE Transactions on Multimedia*, 18(11):2149–2160.

LANCICHINETTI, A. et FORTUNATO, S. (2009). Community detection algorithms : a comparative analysis. *Physical review E*, 80(5):056117.

LAPTEV, I. et PÉREZ, P. (2007). Retrieving actions in movies. pages 1–8.

LE MARTELOT, E. et HANKIN, C. (2011). Multi-scale community detection using stability as optimisation criterion in a greedy algorithm. pages 216–225.

LE MARTELOT, E. et HANKIN, C. (2013). Fast multi-scale detection of relevant communities in large-scale networks. *The Computer Journal*, 56(9):1136–1150.

LECHLEK, L., GHOUGAL, M. et BOUTAOUCHE, F. (2012). La segmentation d'image médicale par la méthode d'ensembles de niveaux level sets.

LI, H., CAI, J., NGUYEN, T. N. A. et ZHENG, J. (2013). A benchmark for semantic image segmentation. pages 1–6.

LI, J. et SONG, Y. (2013). Community detection in complex networks using extended compact genetic algorithm. *Soft computing*, 17(6):925–937.

LI, J., ZHANG, K. *et al.* (2007). Keyword extraction based on tf/idf for chinese news document. *Wuhan University Journal of Natural Sciences*, 12(5):917–921.

LI, S. et WU, D. O. (2014). Modularity-based image segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(4):570–581.

LINARES, O. A., BOTELHO, G. M., RODRIGUES, F. A. et NETO, J. B. (2017). Segmentation of large images based on super-pixels and community detection in graphs. *IET Image Processing*, 11(12):1219–1228.

LIU, Y., NIE, L., LIU, L. et ROSENBLUM, D. S. (2016). From action to activity : sensor-based activity recognition. *Neurocomputing*, 181:108–115.

LUCAS, G. (1977). Star wars : Episode iv- a new hope. *Twentieth Century Fox Film Corporation.*

LUCAS, G. (1980). Star wars : Episode v- the empire strikes back. *Twentieth Century Fox Film Corporation.*

LUCAS, G. (1983). Star wars : Episode vi- return of the jedi. *Twentieth Century Fox Film Corporation.*

LUCAS, G. (1999). Star wars : Episode i- the phantom menace. *Twentieth Century Fox Film Corporation.*

LUCAS, G. (2002). Star wars : Episode ii- attack of the clones. *Twentieth Century Fox Film Corporation.*

LUO, H. et FAN, J. (2004). Concept-oriented video skimming and adaptation via semantic classification. pages 213–220.

LV, J., WU, B., ZHOU, L. et WANG, H. (2018). Storyrolenet : Social network construction of role relationship in video. *IEEE Access*, 6:25958–25969.

MA, Y.-F., LU, L., ZHANG, H.-J. et LI, M. (2002). A user attention model for video summarization. pages 533–542.

MAHESWARI, S. et KORAH, R. (2016). Review on image segmentation based on color space and its hybrid. pages 639–641.

MANCORIDIS, S., MITCHELL, B. S., RORRES, C., CHEN, Y. et GANSNER, E. R. (1998). Using automatic clustering to produce high-level system organizations of source code. pages 45–52.

MARR, D. et HILDRETH, E. (1980). Theory of edge detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 207(1167):187–217.

MATHIEU, C. (1996). Segmentation d'images par pyramides souples : Application a l'imagerie medicale multidimensionnelle.

MCINERNEY, T. et TERZOPOULOS, D. (1996). Deformable models in medical image analysis : a survey. *Medical image analysis*, 1(2):91–108.

MEILA, M. (2005). Comparing clusterings : an axiomatic view. pages 577–584.

MISH, B. (2016). Game of nodes : A social network analysis of game of thrones.

MORI, G. (2005). Guiding model search using segmentation. 2:1417–1423.

MOURCHID, Y., RENOUST, B., CHERIFI, H. et HASSOUNI, M. E. (2018). Multilayer network model of movie script. pages 782–796.

MUERLE, J. L. (1968). Experimental evaluation of techniques for automatic segmentation of objects in a complex scene. *Pictorial pattern recognition*, pages 3–13.

NEWMAN, M. E. (2006). Modularity and community structure in networks. *Proceedings of the national academy of sciences*, 103(23):8577–8582.

NEWMAN, M. E. et GIRVAN, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113.

NGO, C.-W., MA, Y.-F. et ZHANG, H.-J. (2005). Video summarization and scene detection by graph modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(2):296–305.

NOBLE, A. et BOUKERROUI, D. (2006). Ultrasound image segmentation : a survey. *IEEE Transactions on medical imaging*, 25(8):987–1010.

OKAMOTO, K., CHEN, W. et LI, X.-Y. (2008). Ranking of closeness centrality for large-scale social networks. pages 186–195.

ORMAN, G. K., LABATUT, V. et CHERIFI, H. (2011a). On accuracy of community structure discovery algorithms. *arXiv preprint arXiv :1112.4134*.

ORMAN, G. K., LABATUT, V. et CHERIFI, H. (2011b). Qualitative comparison of community detection algorithms. pages 265–279.

OTSU, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66.

PARK, S.-B., OH, K.-J. et JO, G.-S. (2012). Social network analysis in a movie using character-net. *Multimedia Tools and Applications*, 59(2):601–627.

PASTRANA-VIDAL, R. R., GICQUEL, J. C., BLIN, J. L. et CHERIFI, H. (2006). Predicting subjective video quality from separated spatial and temporal assessment. 6057:60570S.

PENG, B., ZHANG, L. et ZHANG, D. (2013). A survey of graph theoretical approaches to image segmentation. *Pattern Recognition*, 46(3):1020–1038.

PFEIFFER, S., LIENHART, R., FISCHER, S. et EFFELSBERG, W. (1996). Abstracting digital movies automatically. *Journal of Visual Communication and Image Representation*, 7(4):345–353.

PICHLER, O., TEUNER, A. et HOSTICKA, B. J. (1998). An unsupervised texture segmentation algorithm with feature space reduction and knowledge feedback. *IEEE Transactions on Image Processing*, 7(1):53–61.

PIZZUTI, C. (2012). Boosting the detection of modular community structure with genetic algorithms and local search. pages 226–231.

PLANTIÉ, M. et CRAMPES, M. (2013). Survey on social community detection. *In Social media retrieval*, pages 65–85. Springer.

PUZICHA, J., BUHMANN, J. M., RUBNER, Y. et TOMASI, C. (1999). Empirical evaluation of dissimilarity measures for color and texture. 2:1165–1172.

RAGHAVAN, U. N., ALBERT, R. et KUMARA, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106.

REICHARDT, J. et BORNHOLDT, S. (2006). Statistical mechanics of community detection. *Physical Review E*, 74(1):016110.

REN, H., RENOUST, B., VIAUD, M.-L., MELANÇON, G. et SATOH, S. (2018). Generating visual clouds from multiplex networks for tv news archive query visualization. pages 1–6.

RENOUST, B., KOBAYASHI, T., NGO, T. D., LE, D.-D. et SATOH, S. (2016a). When face-tracking meets social networks : a story of politics in news videos. *Applied Network Science*, 1(1):4.

RENOUST, B., LE, D.-D. et SATOH, S. (2016b). Visual analytics of political networks from face-tracking of news video. *IEEE Transactions on Multimedia*, 18(11):2184–2195.

RENOUST, B., MELANÇON, G. et VIAUD, M.-L. (2014). Entanglement in multiplex networks : understanding group cohesion in homophily networks. *In Social Network Analysis-Community Detection and Evolution*, pages 89–117. Springer.

RITAL, S., CHERIFI, H. et MIGUET, S. (2005). Weighted adaptive neighborhood hypergraph partitioning for image segmentation. pages 522–531.

RONHOVDE, P. et NUSSINOV, Z. (2010). Local resolution-limit-free potts model for community detection. *Physical Review E*, 81(4):046114.

ROSVALL, M. et BERGSTROM, C. T. (2008). Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4): 1118–1123.

ROWLEY, H. A. (1999). Neural network-based face detection. Rapport technique, CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE.

SALTON, G., WONG, A. et YANG, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

SATULURI, V. et PARTHASARATHY, S. (2009). Scalable graph clustering using stochastic flows : applications to community discovery. pages 737–746.

SHI, J. et MALIK, J. (2000). Normalized cuts and image segmentation. *Departmental Papers (CIS)*, page 107.

SIMON, H. A. (1991). The architecture of complexity. *In Facets of systems science*, pages 457–476. Springer.

SIMON, P. et UMA, V. (2018). Review of texture descriptors for texture classification. *In Data Engineering and Intelligent Computing*, pages 159–176. Springer.

SIVIC, J. et ZISSERMAN, A. (2003). Video google : A text retrieval approach to object matching in videos. page 1470.

STRASTERS, K. C. et GERBRANDS, J. J. (1991). Three-dimensional image segmentation using a split, merge and group approach. *Pattern Recognition Letters*, 12(5):307–325.

SUMENGEN, B., BERTELLI, L. et MANJUNATH, B. (2006). Fast and adaptive pairwise similarities for graph cuts-based image segmentation. pages 179–179.

SUNDARAM, H. et CHANG, S.-F. (2001). Condensing computable scenes using visual complexity and film syntax analysis.

TAN, M., UJUM, E. et RATNAVELU, K. (2014). A character network study of two sci-fi tv series. 1588(1):246–251.

TRAAG, V. A. et BRUGGEMAN, J. (2009). Community detection in networks with positive and negative links. *Physical Review E*, 80(3):036115.

TRAN, Q. D. et JUNG, J. E. (2015). Cocharnet : Extracting social networks using character co-occurrence in movies. *J. UCS*, 21(6):796–815.

TRÉMEAU, A. et COLANTONI, P. (2000). Regions adjacency graph applied to color image segmentation. *IEEE Transactions on image processing*, 9(4):735–744.

TRUONG, B. T. et VENKATESH, S. (2007). Video abstraction : A systematic review and classification. *ACM transactions on multimedia computing, communications, and applications (TOMM)*, 3(1):3.

UNNIKRISHNAN, R., PANTOFARU, C. et HEBERT, M. (2007). Toward objective evaluation of image segmentation algorithms. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6):929–944.

VAN DONGEN, S. M. (2000). *Graph clustering by flow simulation*. Thèse de doctorat.

VAPNIK, V. (2013). *The nature of statistical learning theory*. Springer science & business media.

VIARD, T. et FOURNIER-S'NIEHOTTA, R. (2018). Movie rating prediction using content-based and link stream features. *arXiv preprint arXiv :1805.02893*.

VIOLA, P. et JONES, M. J. (2004). Robust real-time face detection. *International journal of computer vision*, 57(2):137–154.

VON LUXBURG, U. (2007). A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416.

WANI, M. A. et BATCHELOR, B. G. (1994). Edge-region-based segmentation of range images. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (3):314–319.

WAUMANS, M. C., NICODÈME, T. et BERSINI, H. (2015). Topology analysis of social networks extracted from literature. *PloS one*, 10(6):e0126470.

WENG, C.-Y., CHU, W.-T. et WU, J.-L. (2009). Rolenet : Movie analysis from the perspective of social networks. *IEEE Transactions on Multimedia*, 11(2):256–271.

WHEDON, J. (2012). Marvel's the avengers. *Marvel Studios*.

WU, Z. et LEAHY, R. (1993). An optimal graph theoretic approach to data clustering : Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (11):1101–1113.

WYSZECKI, G. et STILES, W. S. (1982). *Color science*, volume 8. Wiley New York.

YANG, A. Y., WRIGHT, J., MA, Y. et SASTRY, S. S. (2008). Unsupervised segmentation of natural images via lossy data compression. *Computer Vision and Image Understanding*, 110(2):212–225.

YANG, L., TANG, K., YANG, J. et LI, L.-J. (2017). Dense captioning with joint inference and visual context. 2.

YEUNG, M., YEO, B.-L. et LIU, B. (1996). Extracting story units from long programs for video browsing and navigation. pages 296–305.

YUEPENG, L., CUI, J. et JUNCHUAN, J. (2015). A keyword extraction algorithm based on word2vec. *e-Science Technology & Application*, 4:54–59.

ZHANG, H., FRITTS, J. E. et GOLDMAN, S. A. (2008). Image segmentation evaluation : A survey of unsupervised methods. *computer vision and image understanding*, 110(2):260–280.

ZOU, Q. et BAI, J. (2018). Interest points detection in image based on topology features of multi-level complex networks. *Wireless Personal Communications*, 103(1):715–725.

Résume

L'analyse des données visuelles est toujours une problématique d'intérêt dans différents domaines d'application (surveillance, automobile, sécurité, environnement, médecine,). Dans le cadre du travail réalisé, nous nous sommes intéressés au développement des méthodologies pour analyser les données visuelles, notamment, les images et les vidéos (films). Dans ce travail de thèse, on s'adresse à deux problématiques de la littérature, la segmentation pour les images et l'analyse des histoires pour les vidéos, plus précisément les films. Différentes techniques ont été développées dans la littérature pour l'analyse du contenu visuel, mais qui souffrent de certaines limites selon la nature des données traitées, la précision, la robustesse, la sémantique, etc. En dehors de ces méthodes, il s'est avéré que les approches fondées sur les réseaux complexes sont un très bon outil pour résoudre les deux problématiques mentionnées. Dans ce sens, le travail de cette thèse est divisé en deux parties : La première partie vise à proposer un framework qui est basé sur les approches graphes, notamment les algorithmes de détection des communautés, pour résoudre le problème de la segmentation des images. Quant à la seconde partie, elle propose un modèle multi-couches basé sur les graphes pour analyser les histoires des films. Ce modèle exploite le script du film pour construire les liens entre les différentes entités qui composent l'histoire du film. Une extension de ce modèle a été proposée en utilisant les informations textuelles (script et sous-titres) et les informations visuelles (contenu multimédia, reconnaissance faciale, détection des captions) du film. Ces informations permettent d'extraire une structure plus riche du film. Les propriétés des graphes construits ont été exploitées par la suite pour analyser l'histoire du film (ex. propriétés topologiques, centralités, la structure communautaire).

Mots-clefs (9) : analyse visuelle, réseaux complexes, détection des communautés, multi-couches, segmentation des images, analyse des vidéos, films, analyse des histoires, traitement automatique du texte (TAL), reconnaissance faciale.

Abstract

Visual data analysis is always a problem of interest in different fields of application (surveillance, automobile, security, environment, medicine, etc.). In this work, we focus on the development of methodologies for analyzing visual data, including images and videos (movies). We were interested in developing tools dedicated to the analysis of visual content from images and videos (movies). In this thesis, we address two issues, image segmentation and analysis of video stories, especially movie stories. Several techniques were developed in the literature for visual content analysis, which almost suffer from certain limitations depending on the processed data, precision, robustness, semantics, etc. Beyond these proposed methods, it has been shown that complex network approaches are a very good tool to address the two mentioned issues. In this context, this work is divided into two parts: The first part aims to propose a framework based on graph approaches, in particular, community detection algorithms to address the problem of image segmentation. As for the second part, it proposes a multi-layer model based on graphs to analyze movie stories, this model exploits the script of the movie to build interactions between the different entities that constitute the movie story. An extension of this model was proposed next, using the textual information (script and subtitles) and the visual information (multimedia content, facial recognition, dense captioning) of the movie. These informations can extract a richer structure of the movie. Graph properties were exploited next, to analyze the movie story (eg. topological properties, centralities, community structure).

Key Words (9) : visual analytics, complex networks, community detection, multilayer, image segmentation, video analysis, movies, story analysis, natural language processing (NLP), face recognition.